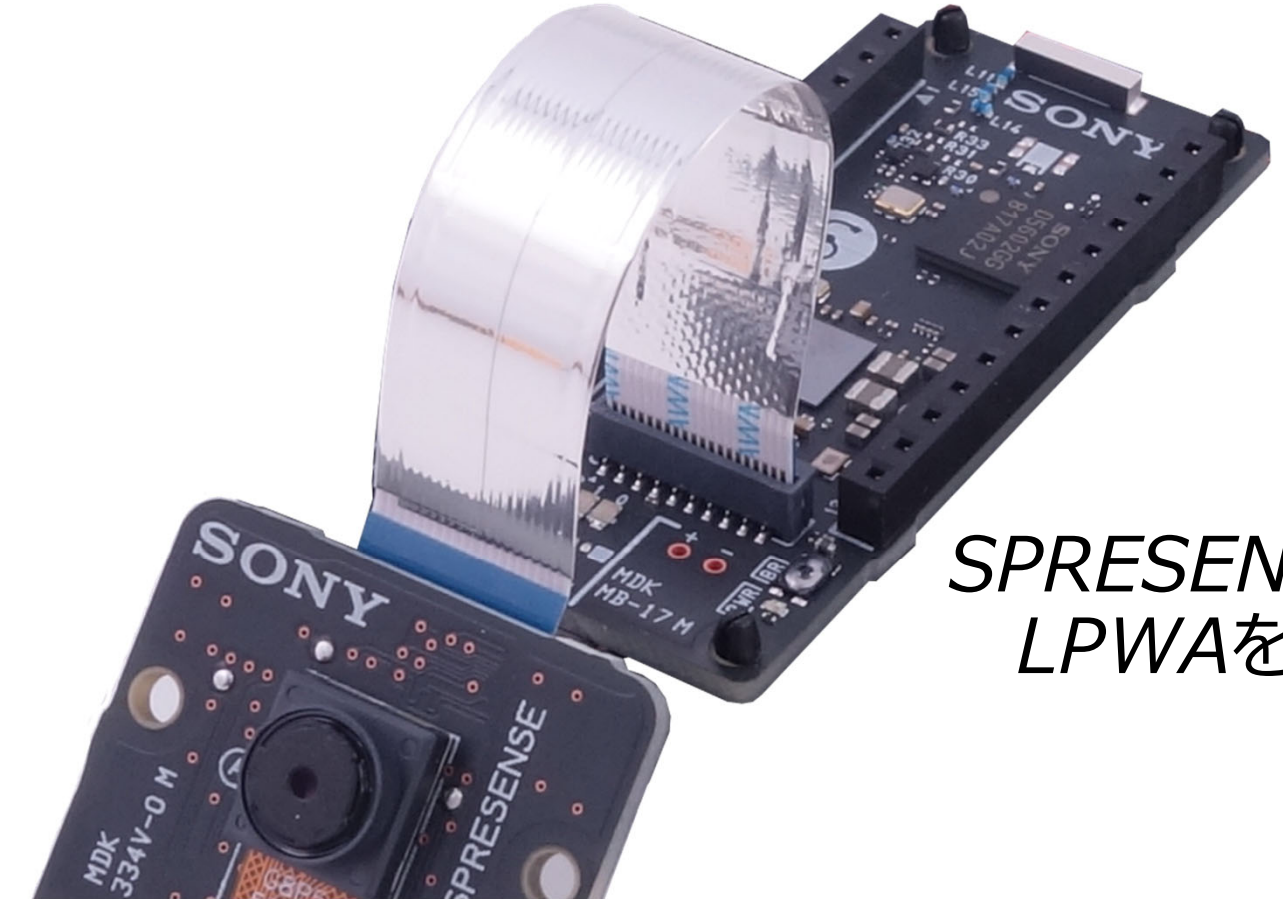


SPRESENSE™ でいろんな *LPWA*を使ってみよう！

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
- SpresenseでいろんなLPWAを使ってみよう！

LPWAとLTE（Cat-M1）について

ライセンスバンド			アンライセンスバンド			
基地局						
スター型					マルチホップ	
移動に強い	移動に弱い		移動に強い	移動に弱い		
LTE-M	NB-IoT	Sigfox	ELTRES	LoRa/ LoRaWAN	Zeta	WiSUN
1Mbps	150kbps	800kbps	数bps	250kbps	50kbps	150kbps
携帯エリア	20km	50km	100km以上	1～10km	2～10km	500m
既存基地局利用。	既存基地局利用。	1日の伝送量制限あり。	移動に強く長距離を飛ぶ。			ホームネットワークなど。

前回のハンズオンは、**LTE cat-M1**でを使ったものでした！

今回は、Spresenseにつながる様々な**LPWA**の通信の**Add-on**ボードを使ってみましょう！

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
- SpresenseでいろんなLPWAを使ってみよう！
  - Wi-SUN Add-onを使ってみよう！
  - Private LoRa Add-onを使ってみよう！
  - Zeta Add-onを使ってみよう！

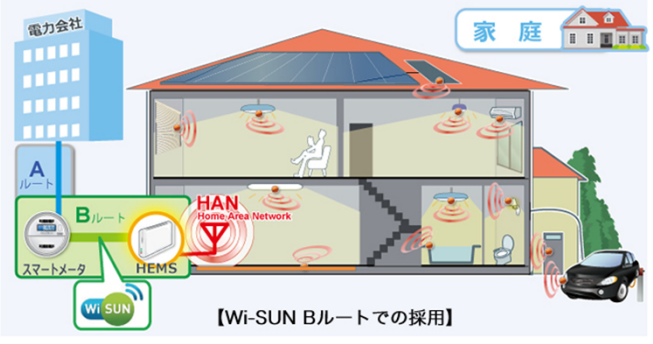
Wi-SUNとは？

Wi-SUNとは？

Wi-SUNは Wireless Smart Utility Network の略で、日本では特定小電力無線と呼ばれる920MHz帯で使用され、2.4GHzや5GHz帯を使用するWi-Fiと比べると、通信速度は遅いものの、通信距離は長く、障害物にも強くてつながりやすく、しかも低消費電力という利点があります。

Wi-SUNの採用

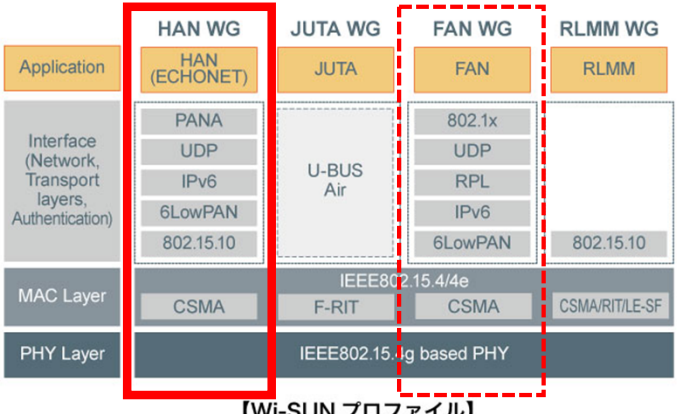
Wi-SUNは日本を代表する電力会社のスマートメータ（次世代電力量計）と家庭内ＨＥＭＳ（Home　Energy Management System）間のBルート通信に標準採用されています。



【Wi-SUN Bルートでの採用】

Wi-SUNプロファイル

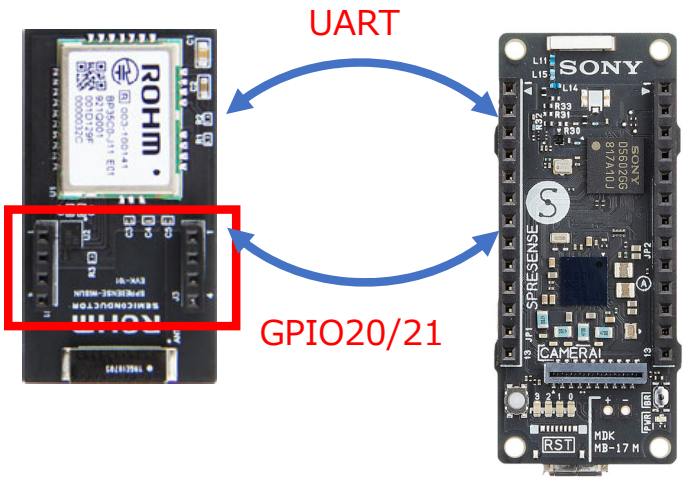
Wi-SUNはIEEE 802.15.4g規格に基づく通信で、IEEE 802.15.4gを最下層のプロトコルベースとしており、その上位層は、アプリケーションに応じてにプロファイルが決められています。
このWi-SUNプロファイルの作成や相互接続性を確保するための認証、およびその普及活動はWi-SUNアライアンスが行っています。



【Wi-SUN プロファイル】

Wi-SUN Add-on ボードの紹介

ローム社提供のAdd-onボードです。UARTによる通信で制御されます。



詳しくは、ローム社のサイトをご確認ください。

https://www.rohm.co.jp/support/spresense-add-on-board

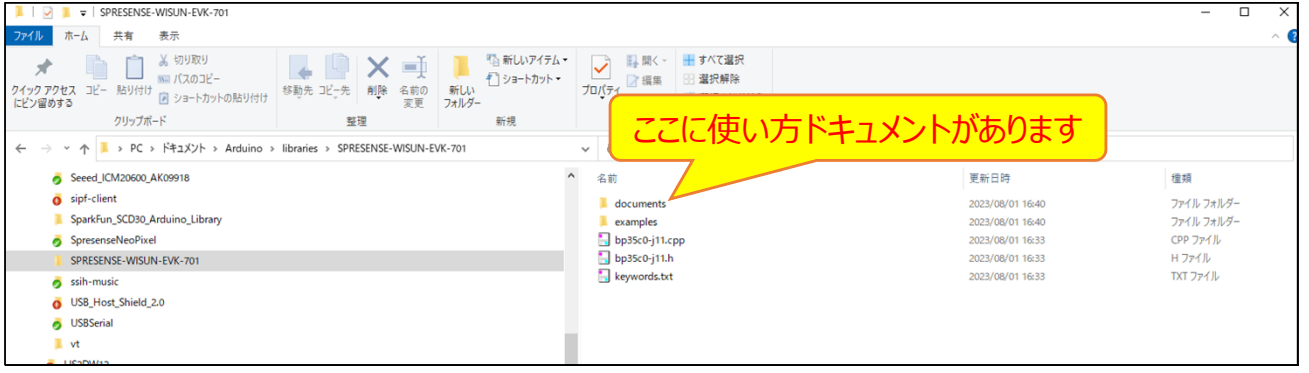
回路図などもあります。

Wi-SUNボード動作確認の流れ

Wi-SUNボードの使い方等の情報は、ロームさんのGithub上にあります。

https://github.com/RohmSemiconductor/Arduino

こちらを取り出した後、“SPRESENSE-WISUN-EVK-701“のディレクトリを、”Documents¥Arduino¥libraries”の下にコピーしてください。



ライブラリプロパティーがないため、ツールからのインストールができません。

Wi-SUNボード動作確認の流れ

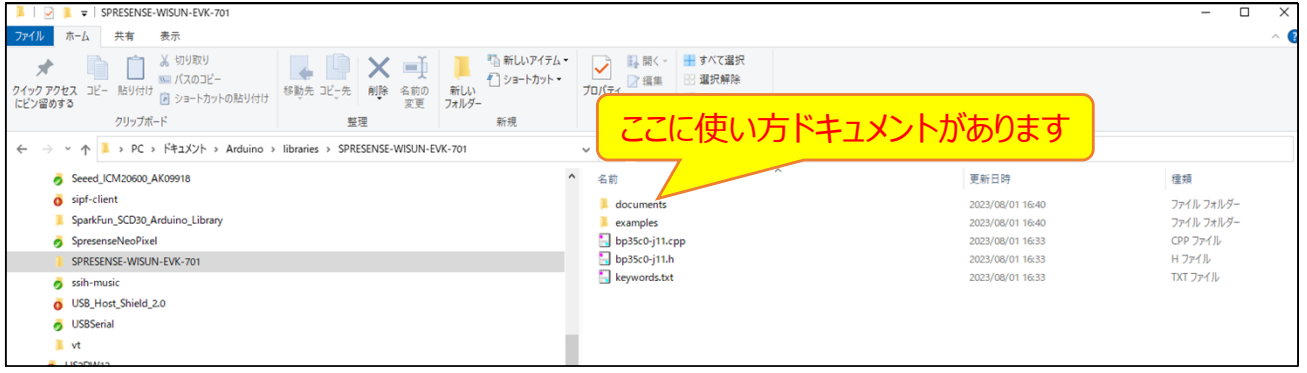
Wi-SUNボードの使い方等の情報は、ロームさんのGithub上にあります。

https://github.com/RohmSemiconductor/Arduino

が、こちらはあまり十分ではなかったので、自分の方で更新しました。こちらを使ってもらう方が良いかと思います。

https://github.com/TomonobuHayakawa/Arduino-Rohm/

こちらを取り出した後、“SPRESENSE-WISUN-EVK-701“のディレクトリを、”Documents¥Arduino¥libraries”の下にコピーか、ライブラリの ”**ライブラリのインクルード->Zip形式でのライブラリのインクルード**” してください。



Wi-SUNボード動作確認の流れ

想定する使い方次第ですが、ここでは、Spresense をエンドデバイスとして、PCにWi-SUNドングル経由でデータをuploadするサンプルになっています。

実際には、Spresense を親機にすることもできます！

PCドングルの後継機はこちらになります。

https://www.ratocsystems.com/products/wisun/usb-wisun/rs-wsuha/

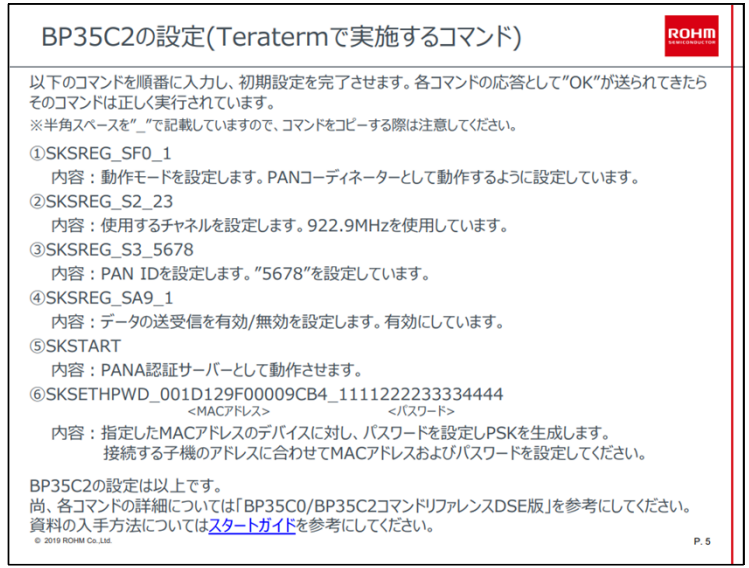
Wi-SUNボード動作確認の流れ

PCドングルを使うのにTeratermが必要です。
こちら（https://ttssh2.osdn.jp/）からダウンロード&インストールしてください。

PCドングルのドライバは、Windowsが自動的にインストールしてくれると思いますが、もし、しばらくしてもポートが見えない場合は、こちら（ Drivers - FTDI (ftdichip.com) ）からダウンロード&インストールしてください。

USBドングルの使い方は、こちらを参照してください。

RohmSemiconductor/Arduino/SPRESENSE-WISUN-EVK-701/documents/SPRESENSE-WiSUN-EVK-701_サンプルソフトウェア説明書_ROHC.pdf



BP35C2の設定(Teratermで実施するコマンド)

以下のコマンドを順番に入力し、初期設定を完了させます。各コマンドの応答として"OK"が送られてきたらそのコマンドは正しく実行されています。
※半角スペースを"_"で記載していますので、コマンドをコピーする際は注意してください。

①SKSREG_SF0_1
内容：動作モードを設定します。PANコーディネーターとして動作するように設定しています。
②SKSREG_S2_23
内容：使用するチャネルを設定します。922.9MHzを使用しています。
③SKSREG_S3_5678
内容：PAN IDを設定します。"5678"を設定しています。
④SKSREG_SA9_1
内容：データの送受信を有効/無効を設定します。有効にしています。
⑤SKSTART
内容：PANA認証サーバーとして動作させます。
⑥SKSETHPWD_001D129F00009CB4_1111222233334444
<MACアドレス> <パスワード>
内容：指定したMACアドレスのデバイスに対し、パスワードを設定しPSKを生成します。
接続する子機のアドレスに合わせてMACアドレスおよびパスワードを設定してください。

BP35C2の設定は以上です。
尚、各コマンドの詳細については「BP35C0/BP35C2コマンドリファレンスDSE版」を参考にしてください。
資料の入手方法についてはスタートガイドを参考にしてください。

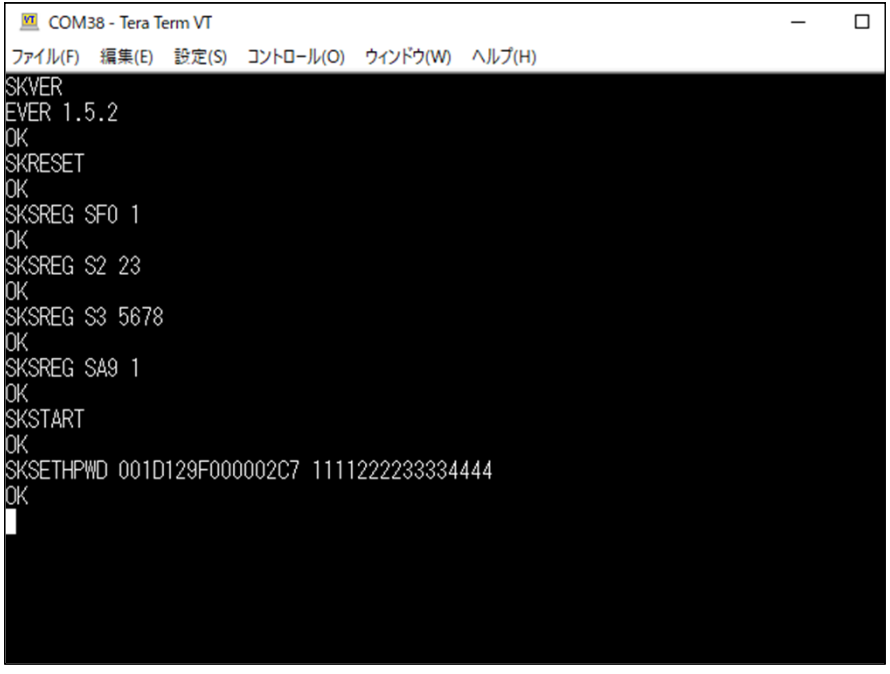
ドングルの起動

PCドングルを親機として起動します。TeraTerm上から以下のコマンドを順に入力します。

SKVER
SKRESET
SKSREG SF0 1
SKSREG S2 23
SKSREG S3 5678
SKSREG SA9 1
SKSTART
SKSETHPWD 001D129F0000XXXX 1111222233334444

Add-onボードの上面に表記

現在持っているWi-SUN
Add-onボードのMACアドレス

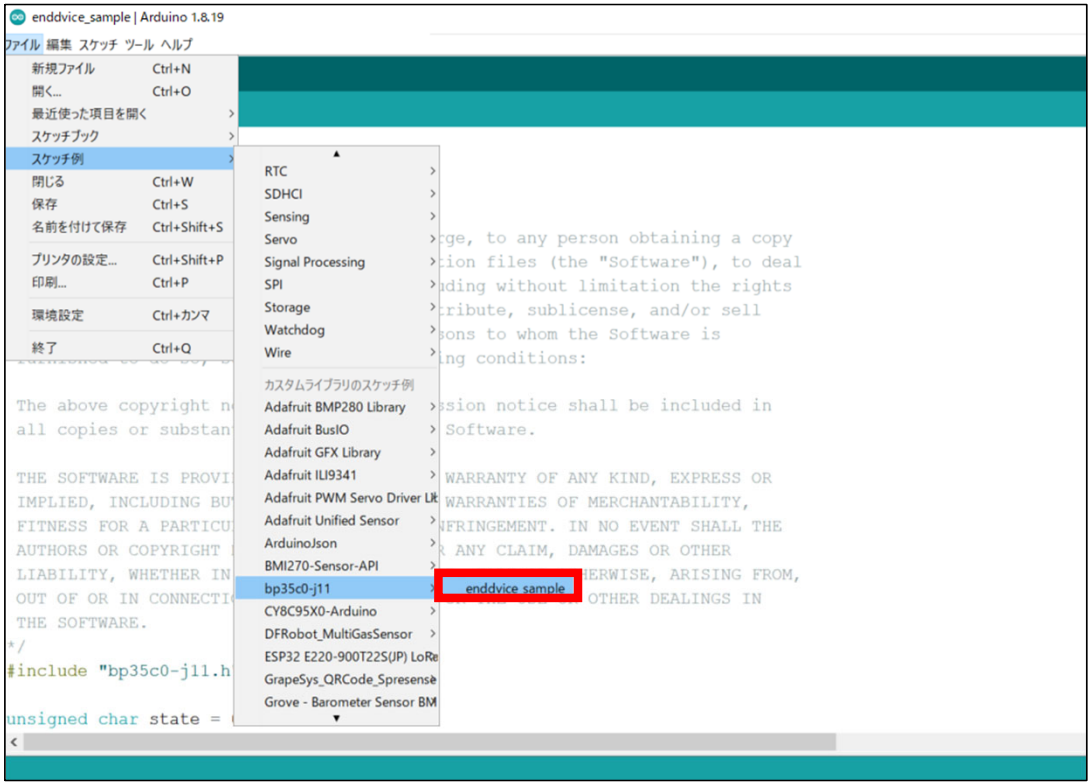


これを実行すると子機からの通信待ち状態になります。

こちらのコマンドは、ここに用意しました。

https://github.com/TomonobuHayakawa/Arduino-Rohm/blob/master/SPRESENSE-WISUN-EVK-701/examples/enddvice_sample/SKSTACK-IP/PAN_Coordinator.txt

enddvice_sample を動かしてみよう！



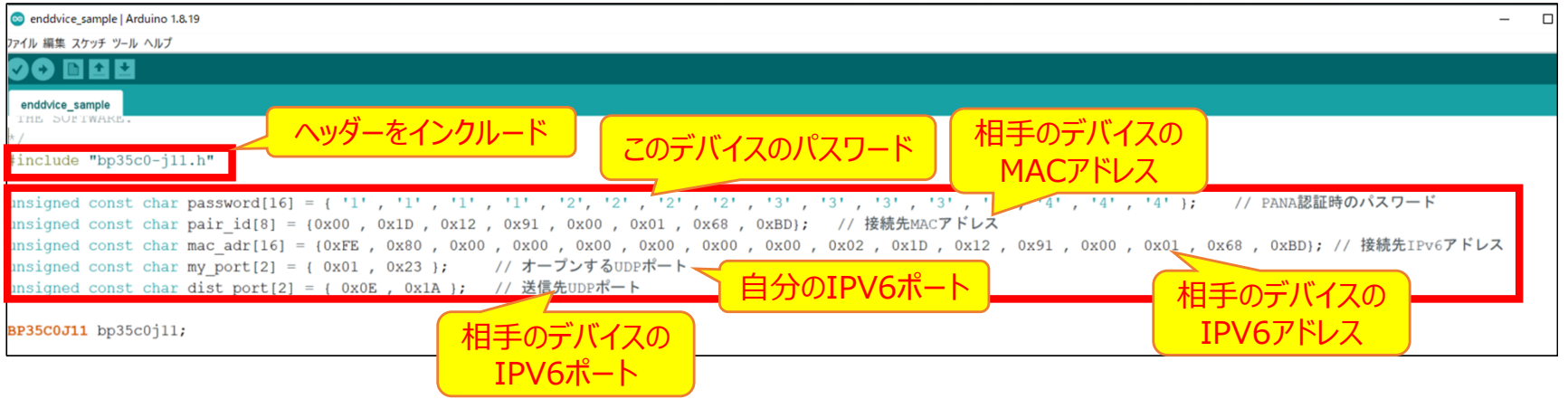
WiSUNライブラリは、ライブラリ名は、デバイス名（bp35c0-j11）にしています。

このWiSUNライブラリの中に **enddevice_sample** というサンプルを作成しました！

こちらを動かしてみましょう！

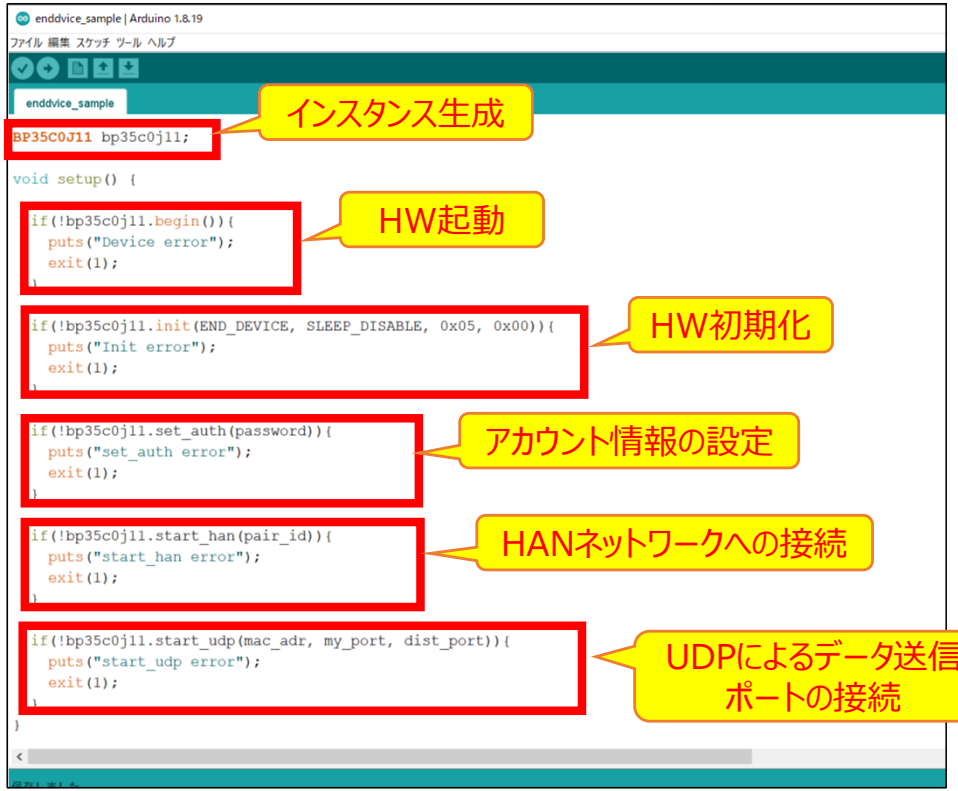
このサンプルは、エンドデバイスとしての動作を行うサンプルになります。

enddvice_sample のコード解説

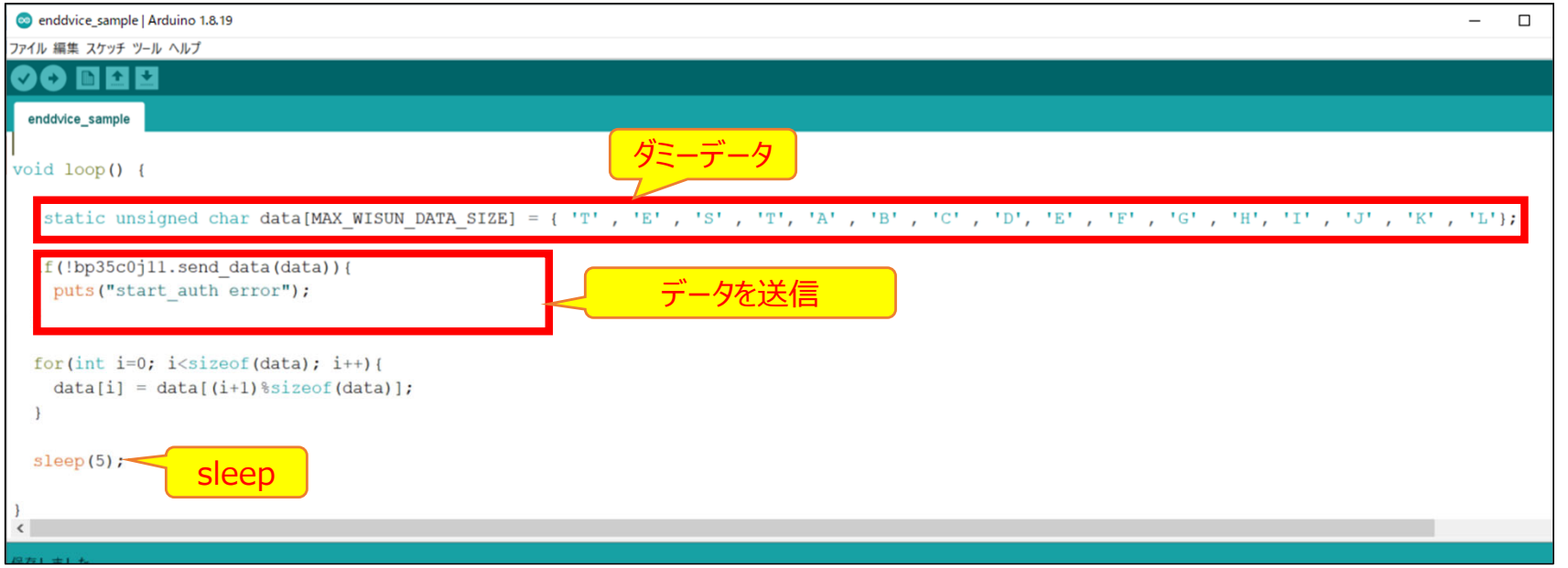


相手デバイス（Wi-SUNドングル）のMACアドレス、IPV6アドレス、ポートは、以下で調べられます。

enddvice_sample のコード解説

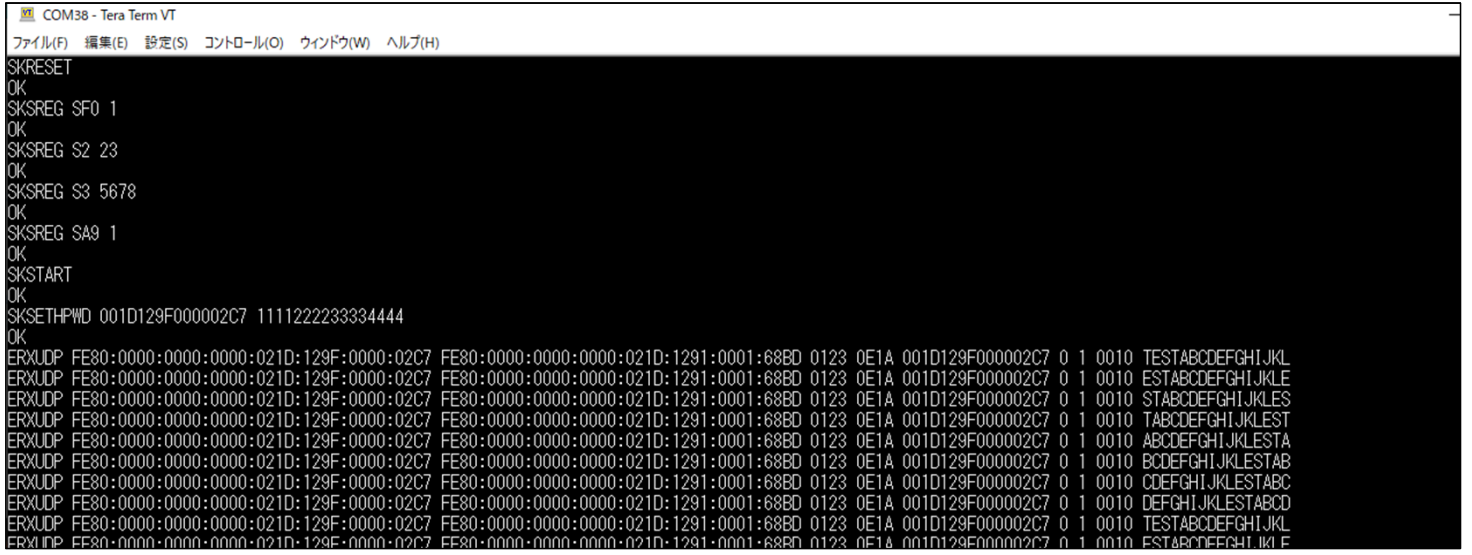


enddvice_sample のコード解説



enddvice_sample の結果

このようにSpresenseからドングルへデータの送信を行えました！
皆さんも、是非、お試しください！



Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
- SpresenseでいろんなLPWAを使ってみよう！
  - Wi-SUN Add-onを使ってみよう！
  - Private LoRa Add-onを使ってみよう！
  - Zeta Add-onを使ってみよう！

Private LoRaとは？

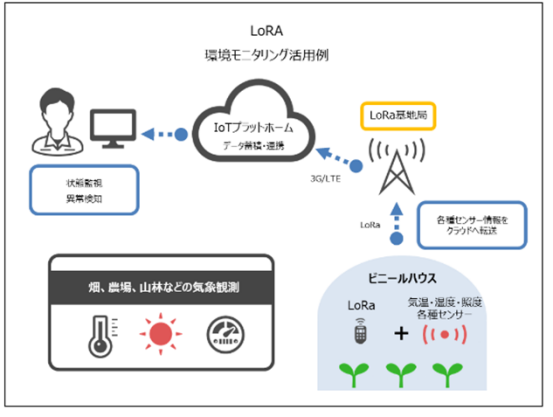
LoRa変調の特徴

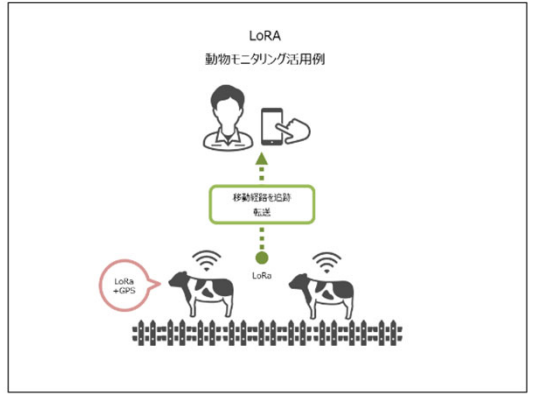
LoRa通信は、920MHz帯での独自のスペクトラム拡散方式で、遠く離れた距離でも、減衰した信号から送信信号を取得でき、ノイズなどにも非常に強い耐性を持つ低消費電力通信です。

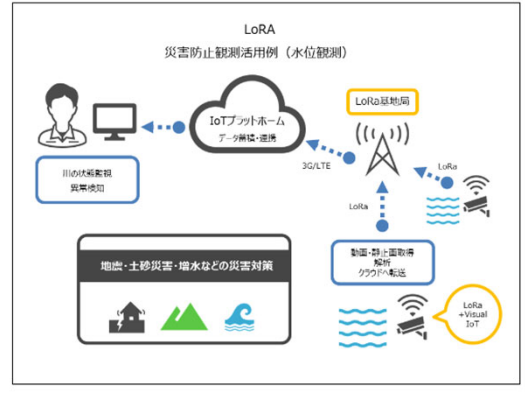
Private LoRaとは

Private LoRaは、デバイス間の通信プロトコルや制御を機器ごとに独自の定めた方式で、通信料金がかからないのも魅力の1つです。

Private LoRaの活用例

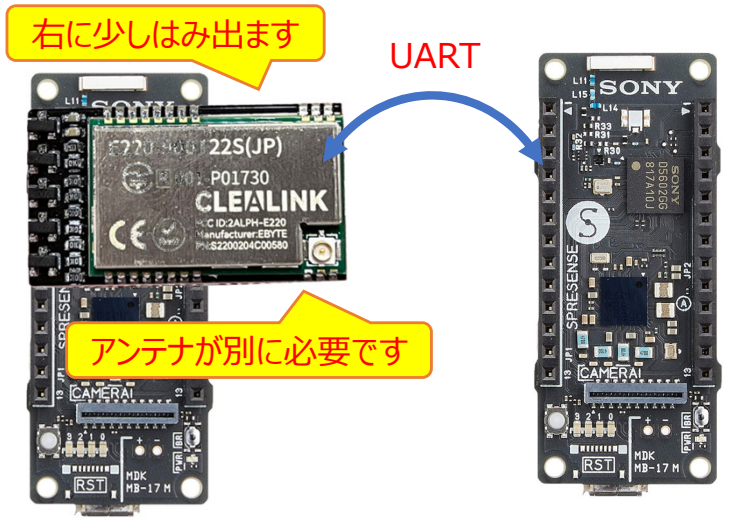






LoRa Add-on ボードの紹介

クレアリンク社提供のAdd-onボードです。UARTによる通信で制御されます。



詳しくは、クレアリンク社のサイトをご確認ください。

https://dragon-torch.tech/cat-components/extension-boards/dth-sslr/

回路図などもあります。

この基板のモジュールでは、見通し距離で最大30km強、郊外市街地で4km前後、直径6～8kmの円域(当社実測値)を1つのアンテナ基地局でカバーできる特徴を持ちます。

LoRaボードドライバについて

LoRa Add-onクレアリンク社からこちらからドライバが提供されています。

https://dragon-torch.tech/cat-components/extension-boards/dth-sslr/

ただし、一部修正したほうがいい部分などもあり、**こちらのライブラリ**を使ってもらえればと思います。

https://github.com/TomonobuHayakawa/Spresense-Playground/tree/master/libraries/spresense_e220900t22s_jp_lib

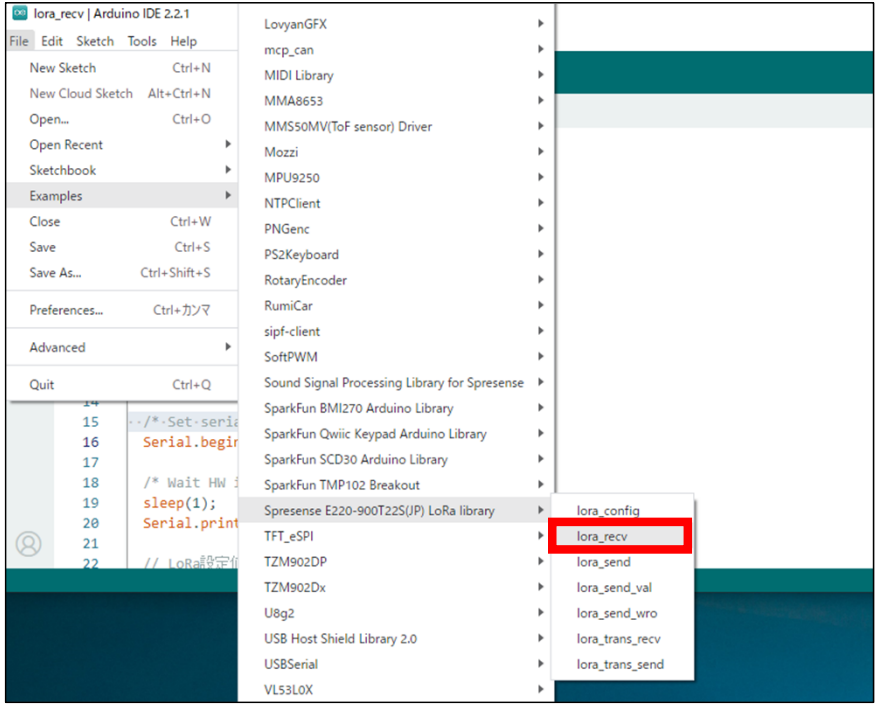
こちらのドライバをArduinoの

ライブラリのインクルード -> Zip形式のライブラリのインクルード

でインクルードしてください。

固定サイズモードでLoRa受信機として動かそう！

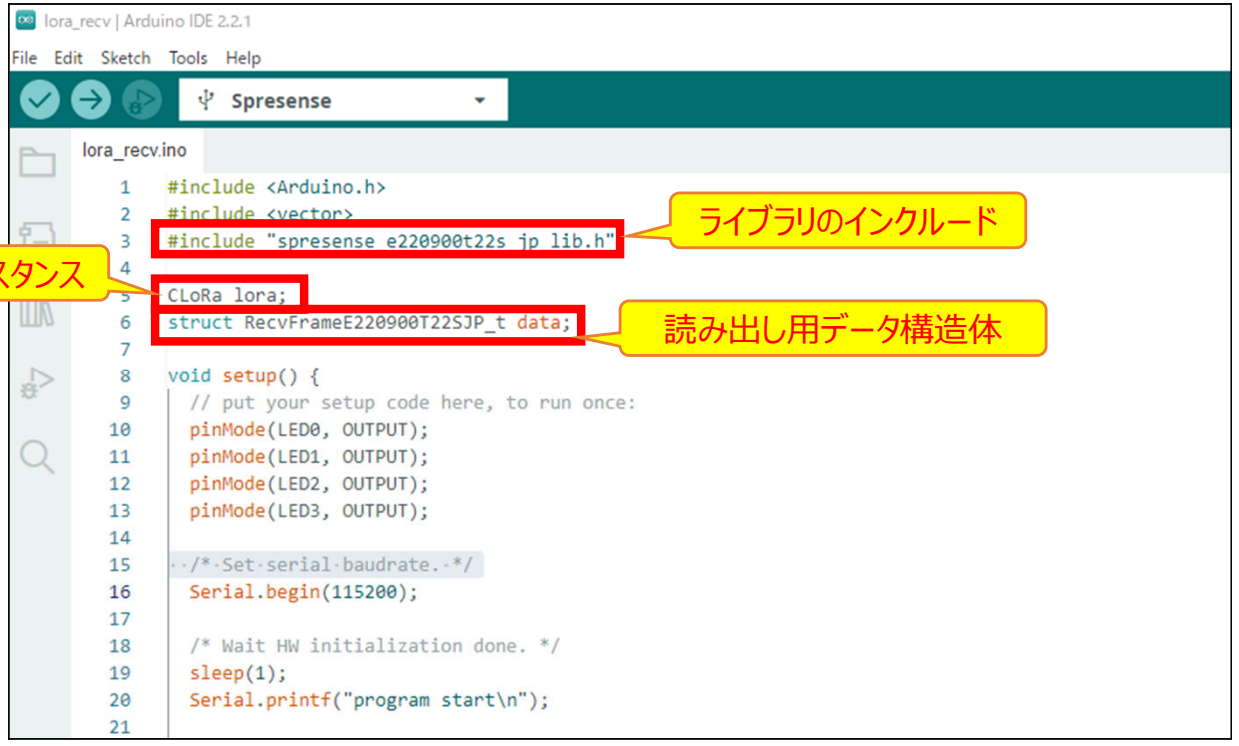
SpresenseをLoRa受信機（固定サイズモード）として動作させるサンプルはこちらになります。
（固定サイズの方が早く確実に送受信が可能ですが、自由度が少なくなります。）



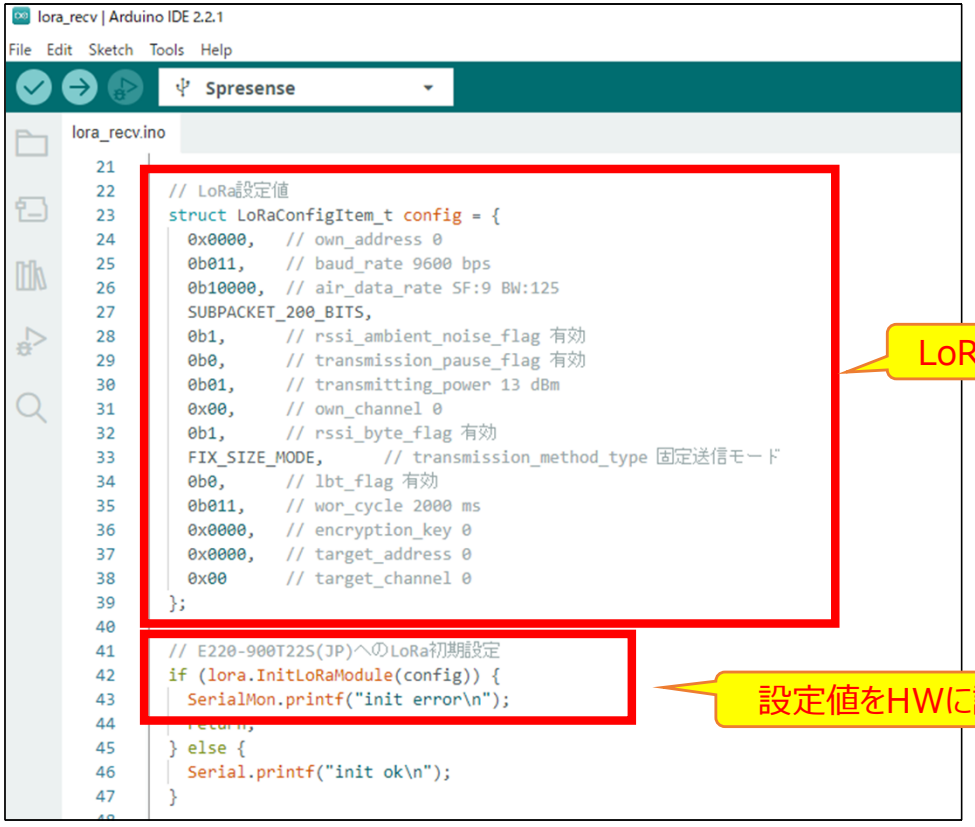
Spresenseで、LoRaを動作させる場合、受信機として動作させるSpresenseを１台と、

送信機として動作させるSpresense１台を１組として動作をさせます。

lora_recv のコード解説



lora_recv のコード解説



lora_recv のコード解説

設定モードから通常モードへ変更

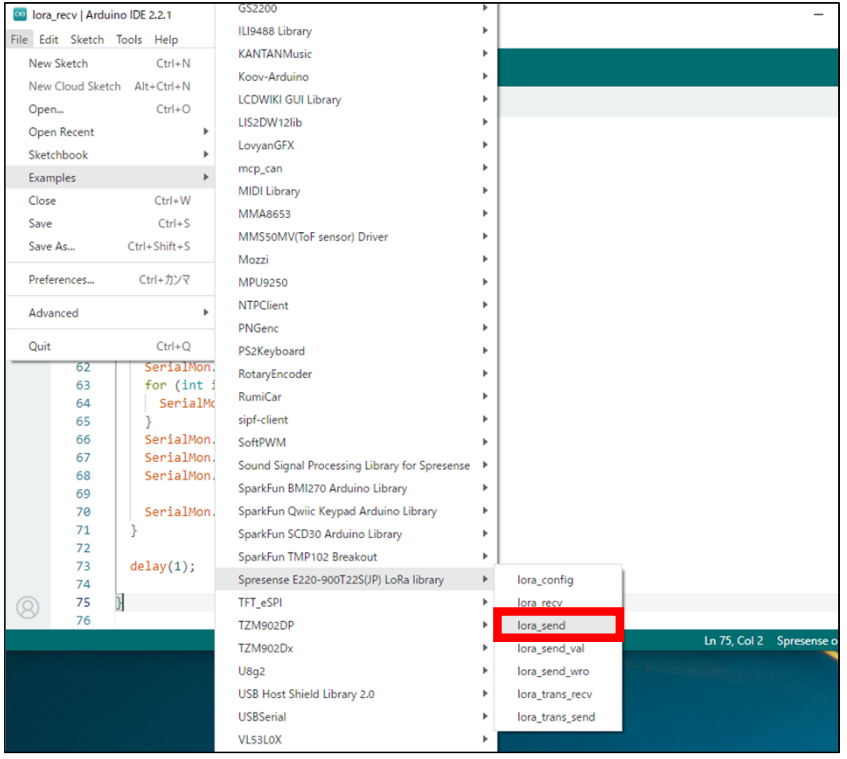
受信したデータがある場合サイズを返す

データサイズ分表示

電波強度を表示

固定サイズモードでLoRa送信機として動かそう！

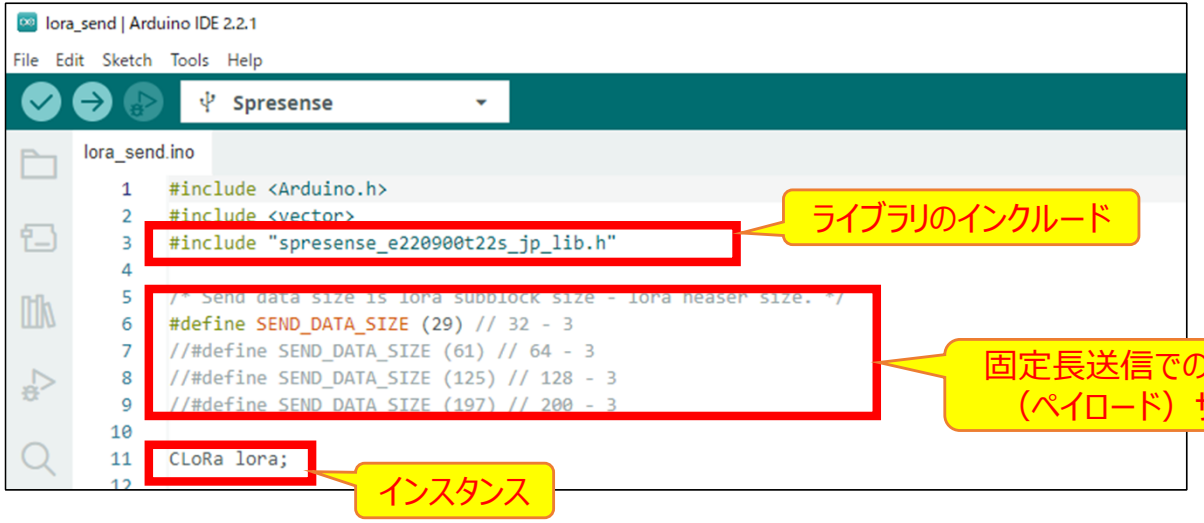
SpresenseをLoRa送信機（固定サイズモード）として動作させるサンプルはこちらになります。



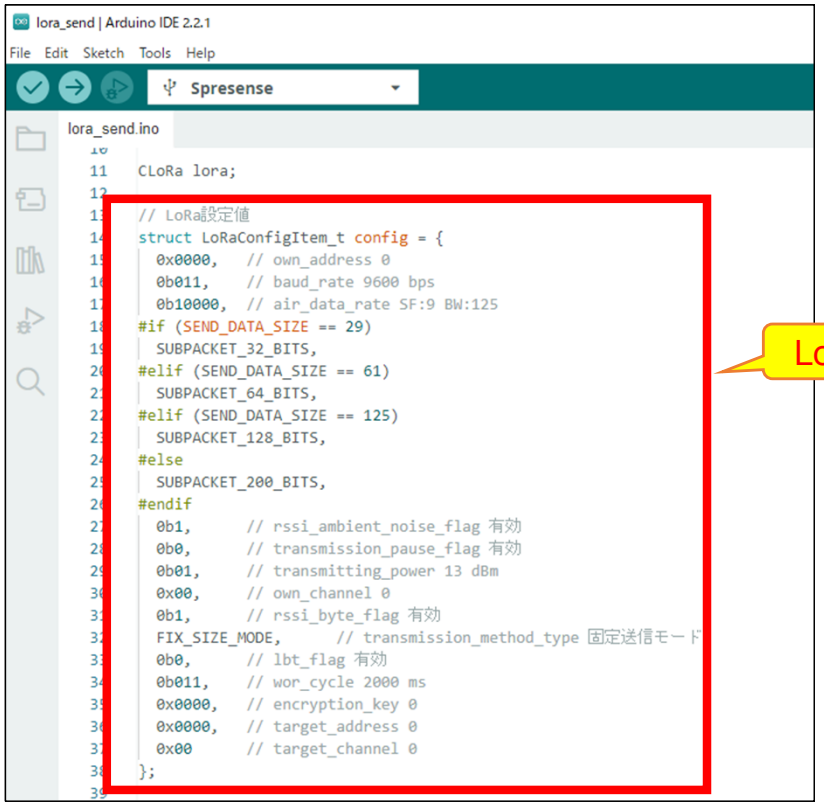
Spresenseで、LoRaを動作させる場合、受信機として動作させるSpresenseを１台と、

送信機として動作させるSpresense１台を１組として動作をさせます。

lora_send のコード解説

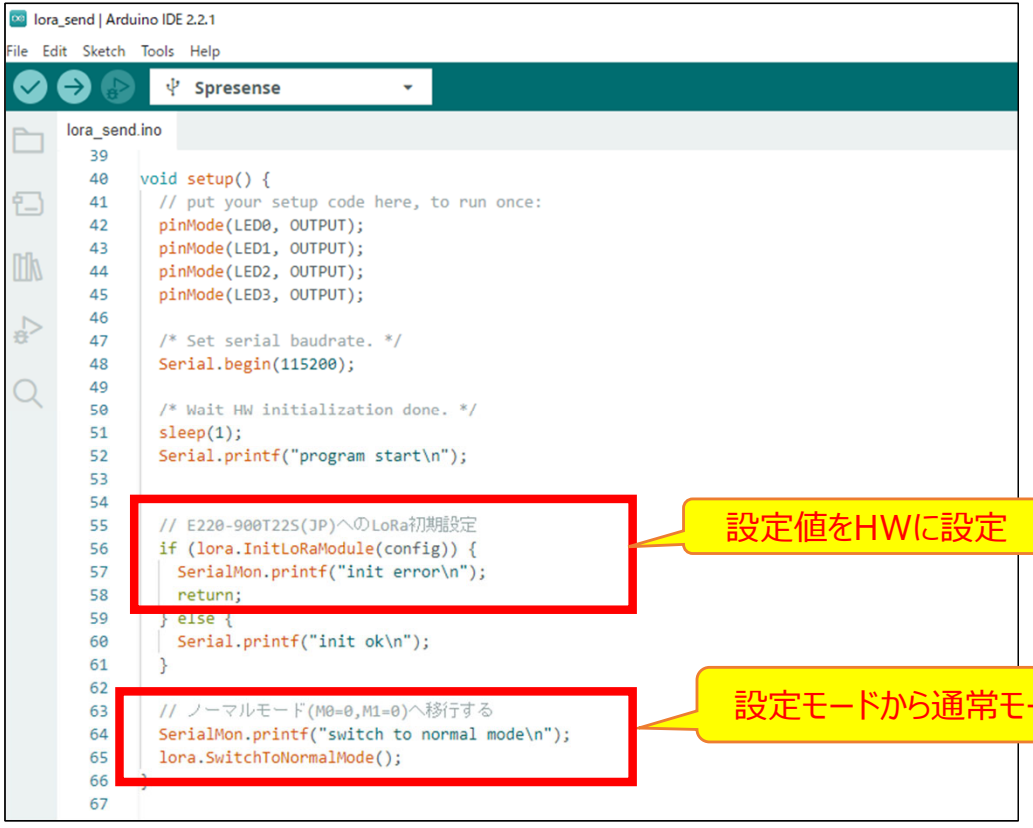


lora_send のコード解説

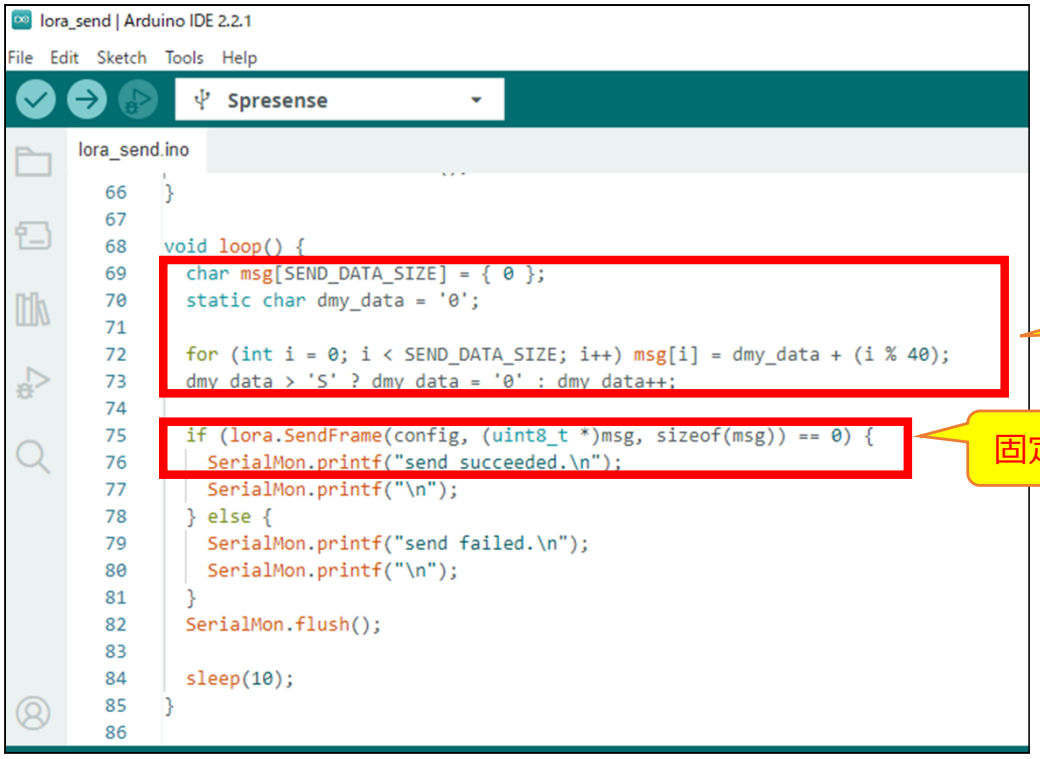


LoRaの各設定

lora_send のコード解説



lora_send のコード解説



ダミーメッセージを生成

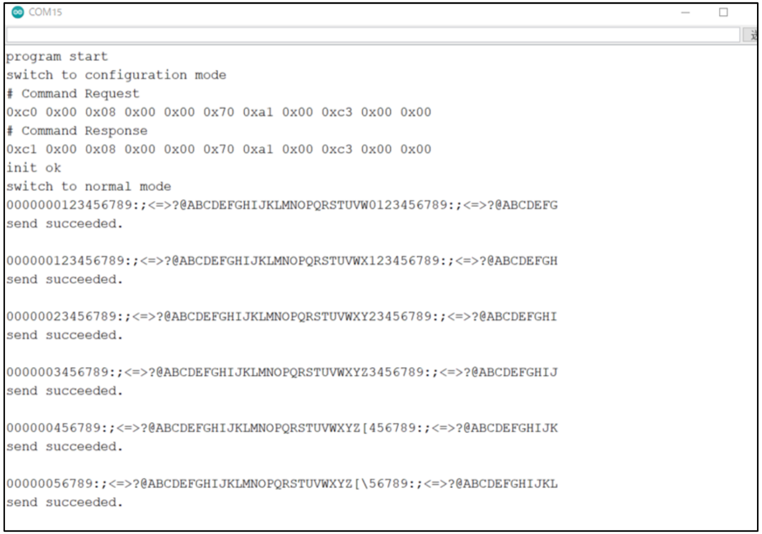
固定長モードでのデータ送信

Configの中のパラメータからヘッダーを生成

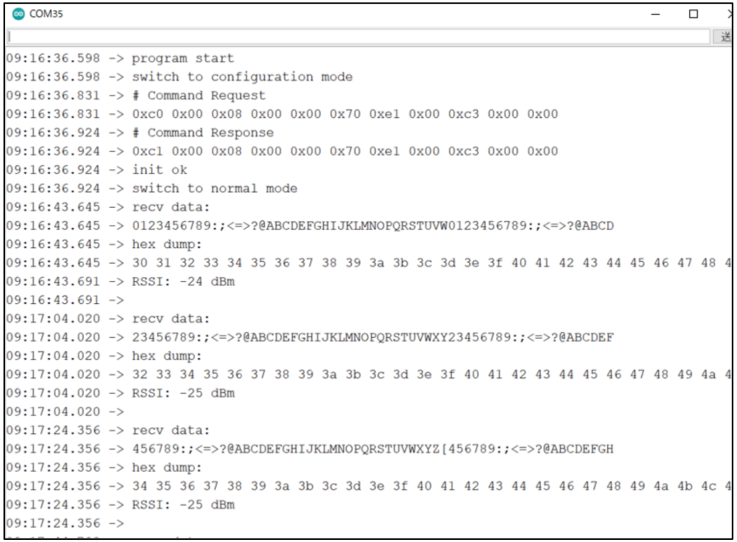
ヘッダーにあるチャネル、アドレスで送信

送受信結果

双方のSpresenseからの送受信結果はこのようになります。



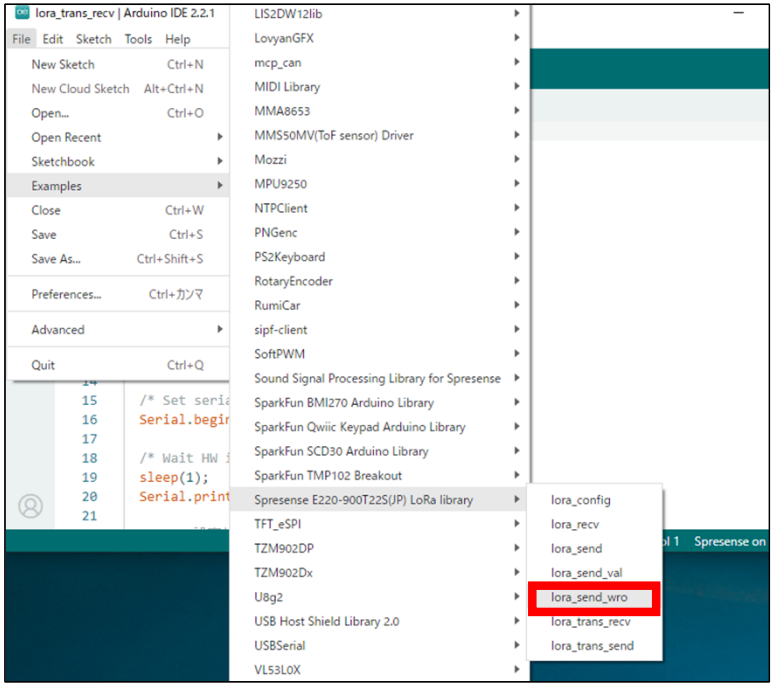
送信結果



受信結果

WROモードでの送信

送信側のSpresenseが送信しか行わない場合は、ノーマルモードではなくWROモードでの送信を行うことで、より電力を抑えることが可能です。



本来、受信側もWROモードでの待ち受けを行うことで、送受信で非常に省電力での動作が可能になるのですが、現時点で、cold sleepからの起動後のし処理がうまくいっていないので、未対応です。

受信はWROモードができたら、サンプルを追加する予定です。

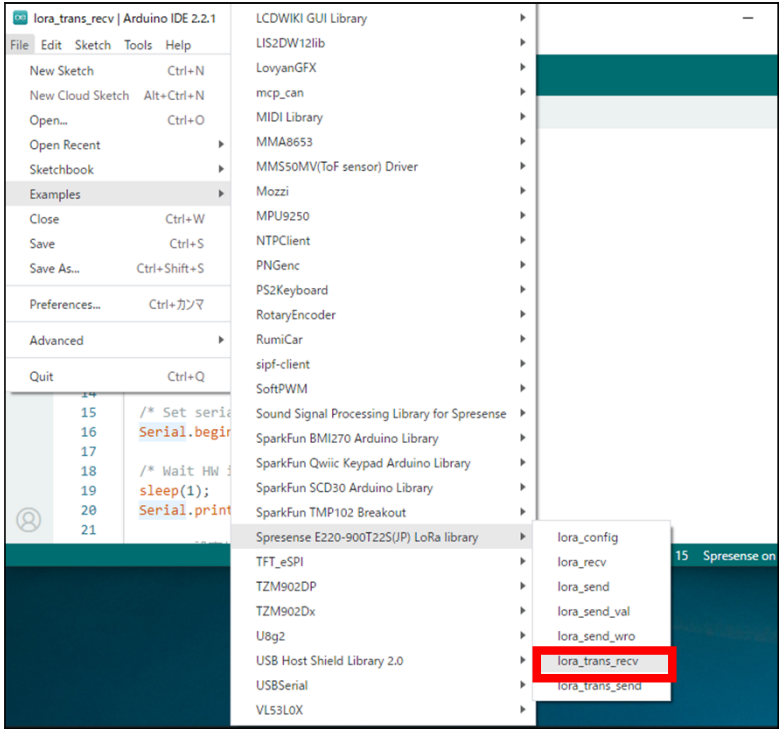
lora_send_wro のコード解説

ノーマルモードとの違いはここだけです。

送受信結果は、ノーマルモードと一緒になります。

トランスペアレントモードでLoRa受信機として動かそう！

SpresenseをLoRa受信機（トランスペアレントモード）として動作させるサンプルはこちらになります。
（固定サイズに比べ受信のタイミング等が不確定ですが、データサイズは自由です。

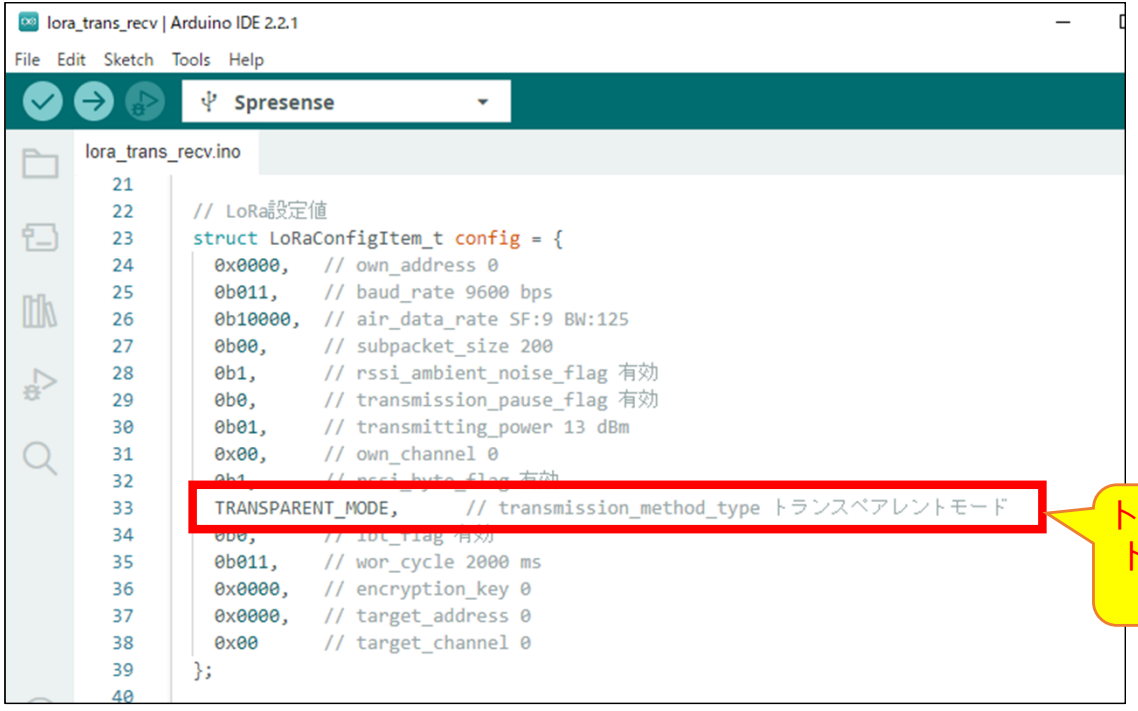


Spresenseで、LoRaを動作させる場合、受信機として動作させるSpresenseを１台と、

送信機として動作させるSpresense１台を１組として動作をさせます。

lora_trans_send のコード解説

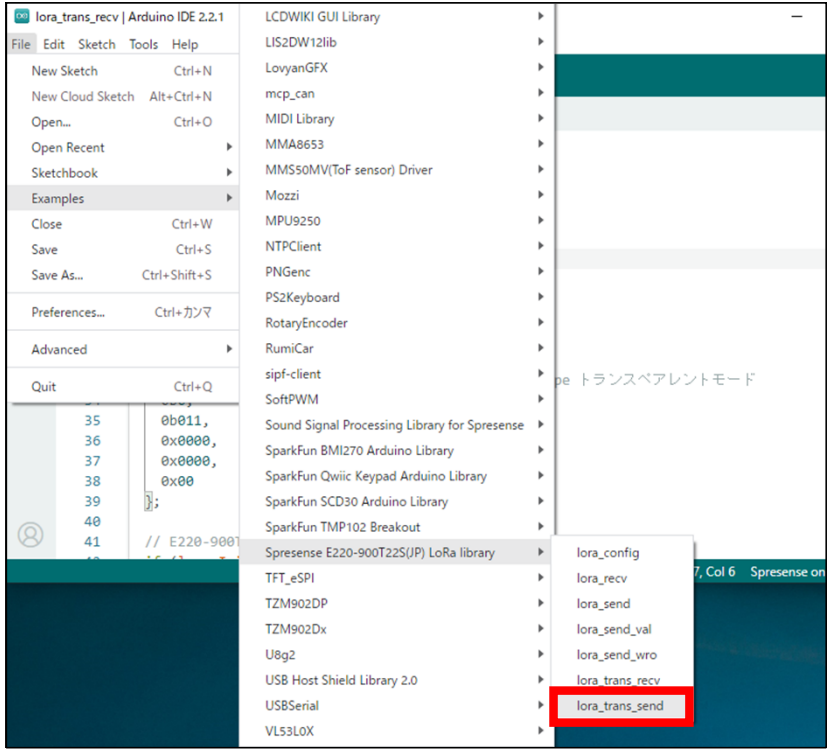
固定サイズモードとの違いはここだけです。



トランスペアレントモードにしないと、
トランスペアレントモードでの送信
を受けれません

トランスペアレントモードでLoRa送信機として動かそう！

SpresenseをLoRa送信機（トランスペアレントモード）として動作させるサンプルはこちらになります。

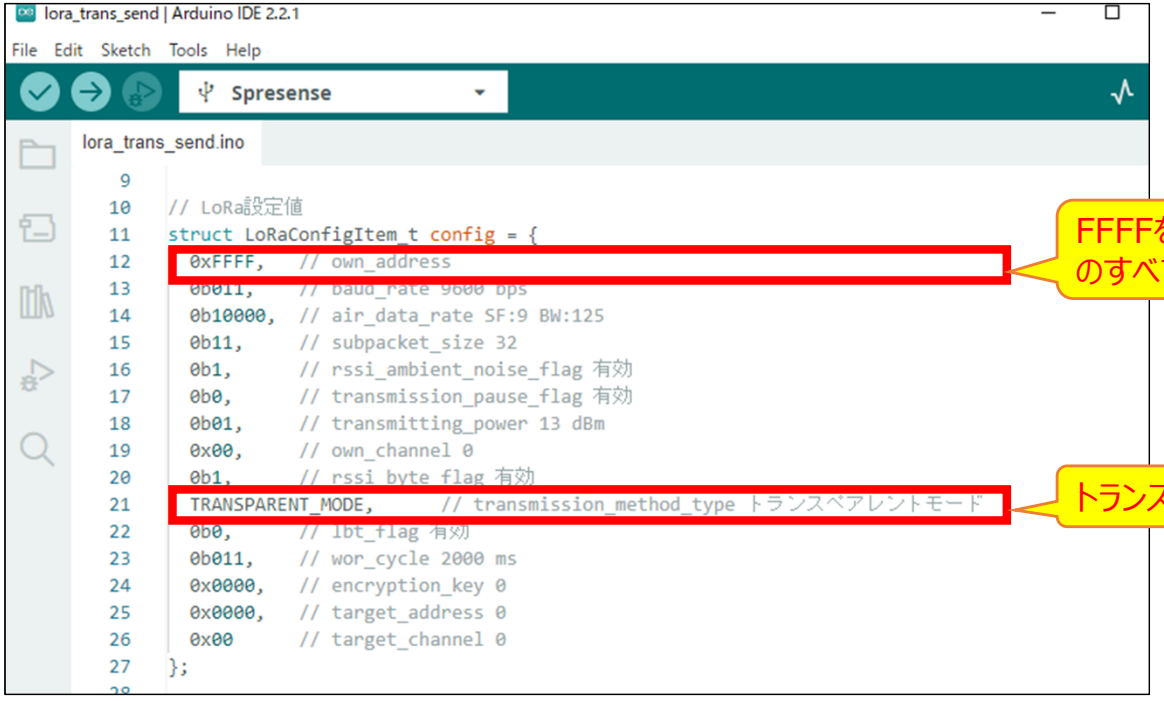


Spresenseで、LoRaを動作させる場合、受信機として動作させるSpresenseを１台と、

送信機として動作させるSpresense１台を１組として動作をさせます。

lora_trans_send のコード解説

固定サイズモードとの違いを示します。



FFFFを設定することで同一チャネルのすべてのアドレスに送信

トランスペアレントモードに設定

lora_trans_send のコード解説

固定サイズモードとの違いを示します。

サイズも変更されます

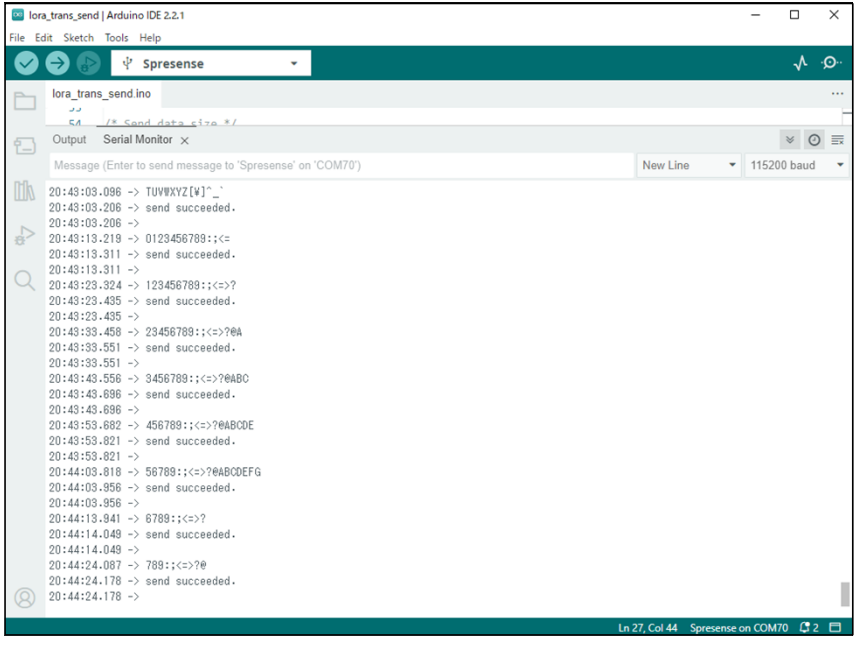
ダミーデータの生成

トランスペアレントモードでの送信

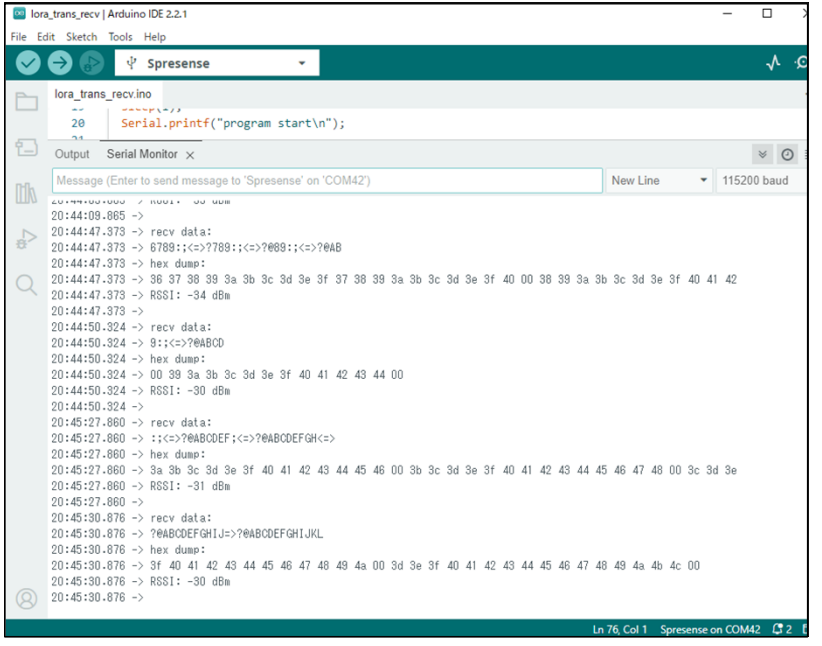
ヘッダーはなくbodyのみ

送受信結果

双方のSpresenseからの送受信結果はこのようになります。



送信結果



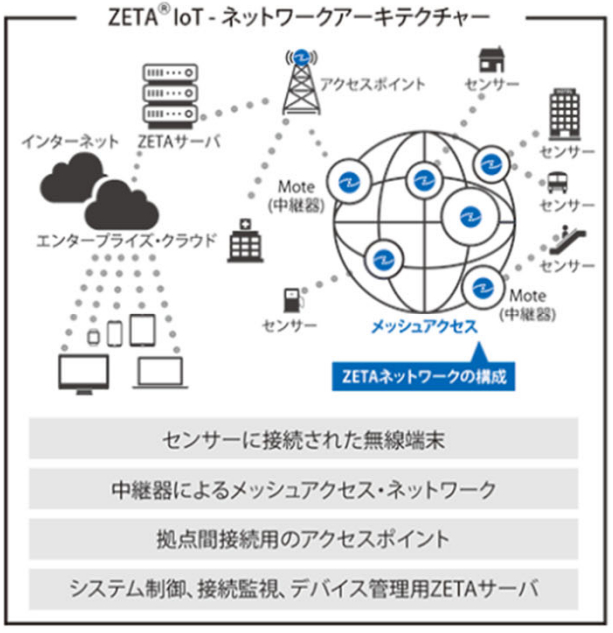
受信結果

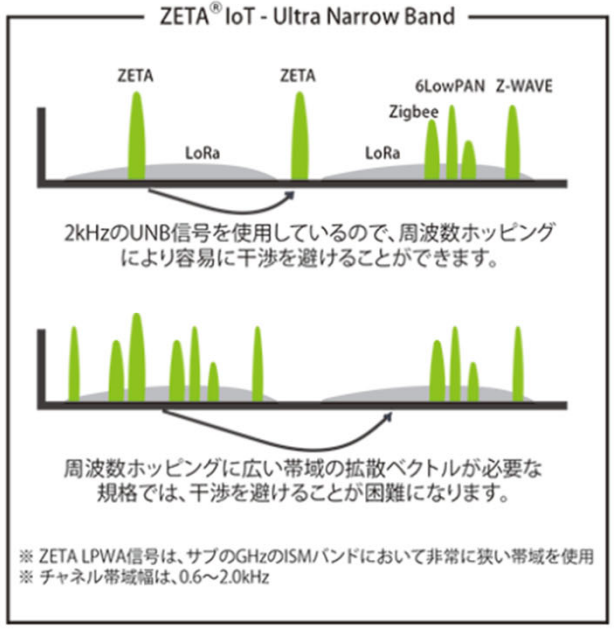
Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
- SpresenseでいろんなLPWAを使ってみよう！
  - Wi-SUN Add-onを使ってみよう！
  - Private LoRa Add-onを使ってみよう！
  - Zeta Add-onを使ってみよう！

ZETA とは？

ZETAは、ZiFiSenseが開発した、**超狭帯域**（UNB: Ultra Narrow Band）、**メッシュネットワーク**による広域での分散アクセス、ローパワーでの**双方向通信**が可能といった特長を持つ、IoTに適したLPWA規格です。





低消費電力
バッテリー寿命5～10年
（通信頻度に依存）

メッシュアクセス
マルチホップ（最大4ホップ）
ネットワーク自動接続

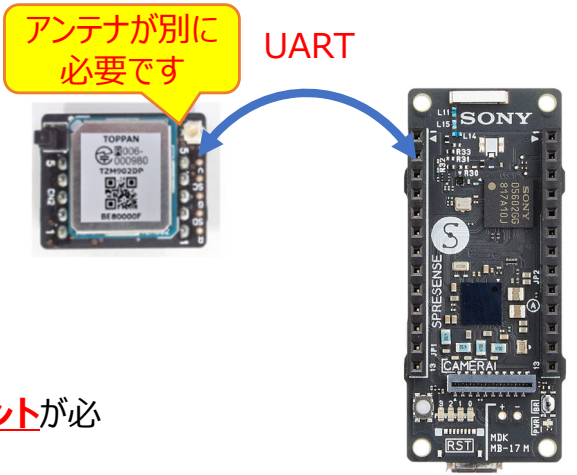
長距離通信
UNB超狭帯域（2KHz）
2～10Km（屋外）

周波数とデータ転送速度
920MHz，429MHz（認証済）
100bps~50kbps

双方向通信
アップリンクデータ
ダウンリンクコントロール

セキュリティ
ネットワーク認証
データ暗号化

Zeta Add-on ボードの紹介



凸版社提供のAdd-onボードです。UARTによる通信で制御されます。

詳しくは、凸版印刷社のサイトをご確認ください。

http://www.toptdc.com/product/zeta/

実際にZetaデバイスを使用する場合、**Zetaサーバ**に接続する**Zetaアクセスポイント**が必要になります。そのため、そちらの準備も必要になります。

http://zeta-factory.shop/shopbrand/ct86/

Allianceメンバーであれば、貸し出し制度があり無料で評価が可能です。

https://zeta-alliance.org/contents/startkit

Zeta ネットワークを使用する流れ（AP設定）

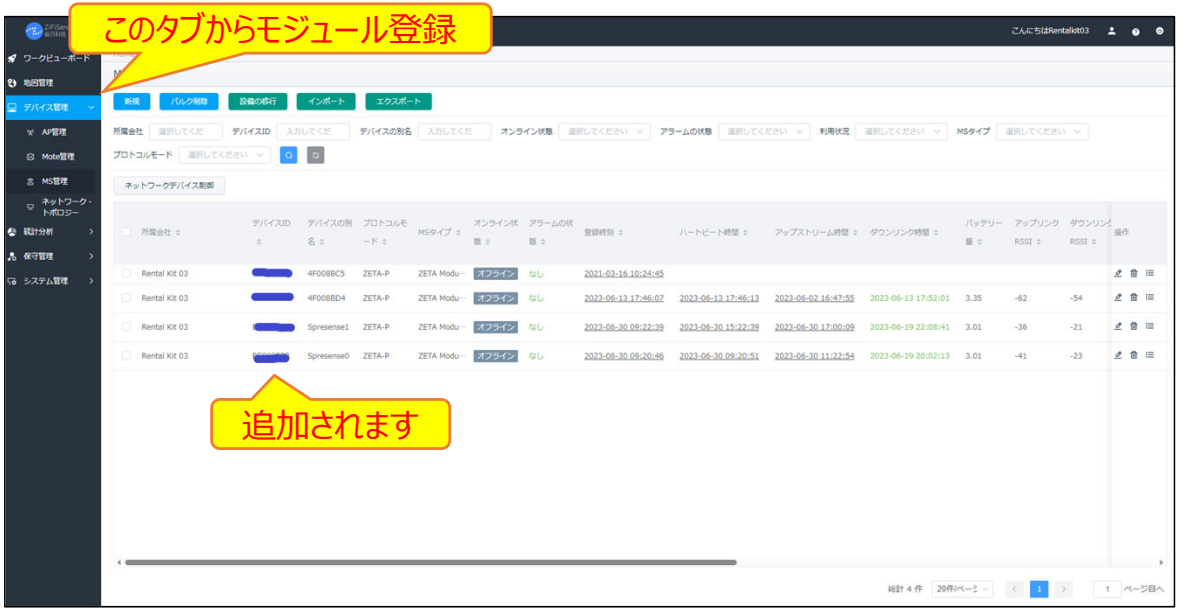
まずは、アクセスポイントの設定とZetaネットワークの登録が必要です。
今回は、テクサー社のアクセスポイント（評価キット）を利用します。

アクセスポイントを説明書通りに接続し、電源を入れます。
テクサー社のデバイス管理サイトにアクセスすると、デバイスの稼働状況が見えます。

※こちらはあくまで**デバイス管理のみ**で**Zetaサービスのサーバ**ではありません。

Zeta ネットワークを使用する流れ（デバイス登録）

Add-onボードについているモジュールIDを登録します。
これによりモジュールの稼働状況もわかるようになります。



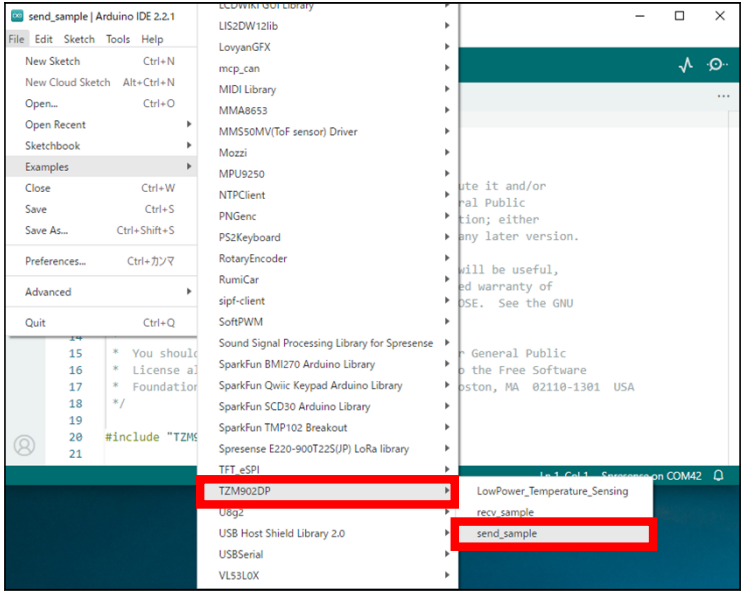
Spresenseを稼働させたとき、こちらがOnになっていないければ通信できていません。

Zetaデバイスからデータを送信しよう！

現時点で、Zetaのライブラリは公開されていません。が、凸版印刷様から事前に頂いたコードをライブラリとして以下に登録しておきました。

こちらをライブラリのインクルードでインクルードしてください。

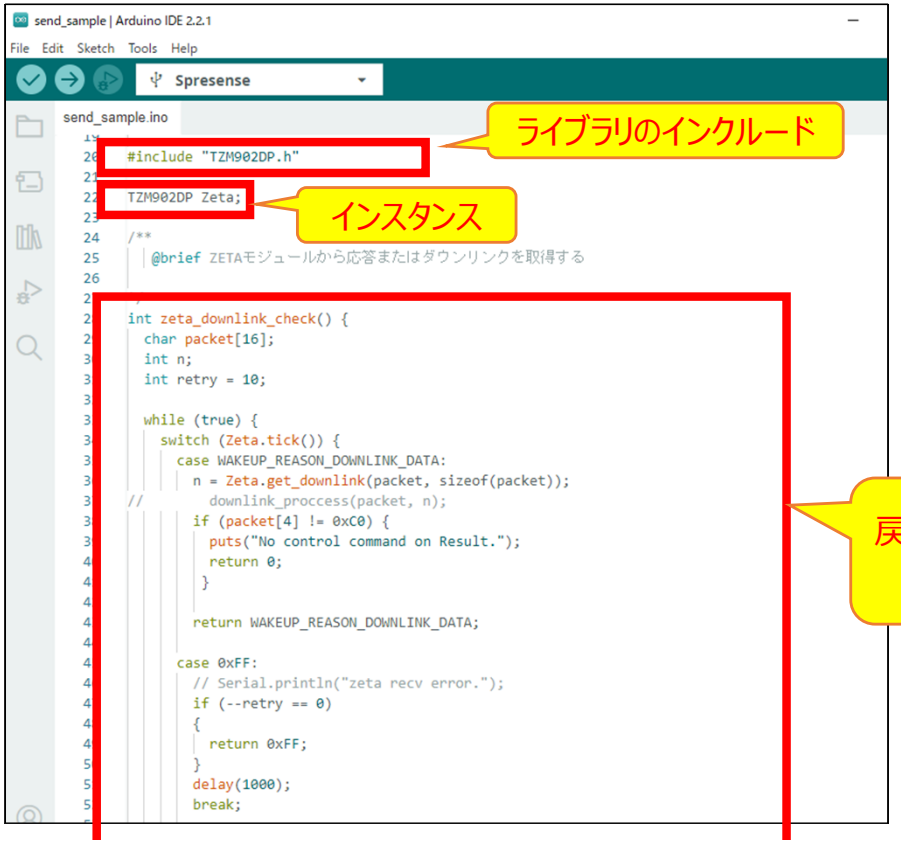
https://github.com/TomonobuHayakawa/TZM902DP/



この中にダミーデータを送信するサンプル（send_sample）を入れておきました。

こちらを利用して、動作をさせてみてください。

send_sample のコード解説



ライブラリのインクルード

インスタンス

戻り値をチェック
する関数

send_sample のコード解説

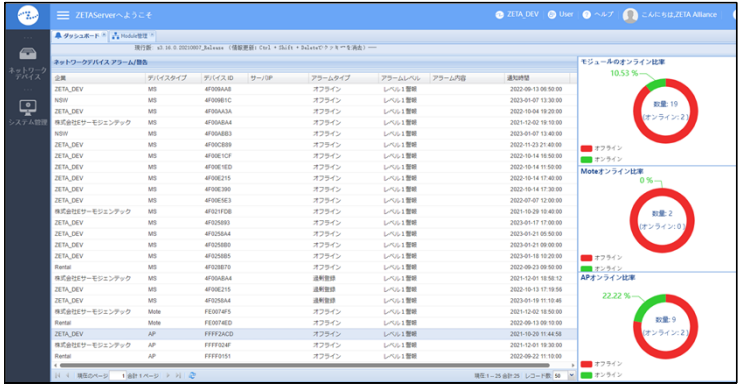
send_sample のコード解説

Zetaサーバー側の設定

Zetaのサーバーは、こちらになります。 http://13.115.15.0:25450/teamcms/login

ログインするとこのようになるのですが、こちらのデバイス管理は実は関係がありません…。

単純にデバイスからの情報をフォワーダーするサーバーになります。



実は**MQTT**/RestAPIは、特に**設定なく使える**ことになります。

NodeREDの設定

NodeREDの環境設定は、「Spresenseからのデータを可視化してみよう！」の中の「フリーブローカを使ってローカルPCで可視化しよう！」の説明を参照してください。

Sony Semiconductor Solutions Corporation © 2019 – 2023

フリーブローカを使ってもっと手軽にローカルPCで可視化

- 外部のサービスを使うと、
  - 契約が必要
  - フリー期
  - 有償

など、制約が発

もっと手軽に見

ローカルPCで

フリーのMQ

を説明します

Sony Semiconductor Solutions Corporation © 2019 – 2023

Node-REDをローカルPCにインストール

https://nodered.jp/docs/getting-started/windows

Node-RED
順に従って、

Node.js
npm

をインストー

その後、Noc

Sony Semiconductor Solutions Corporation © 2019 – 2023

Node-REDの起動

Node-REDを起動します。
※ただし、なぜかPowerShellでの起動には失敗します。
コマンドプロンプトでの実行は成功するのでコマンドプロンプトで起動します。

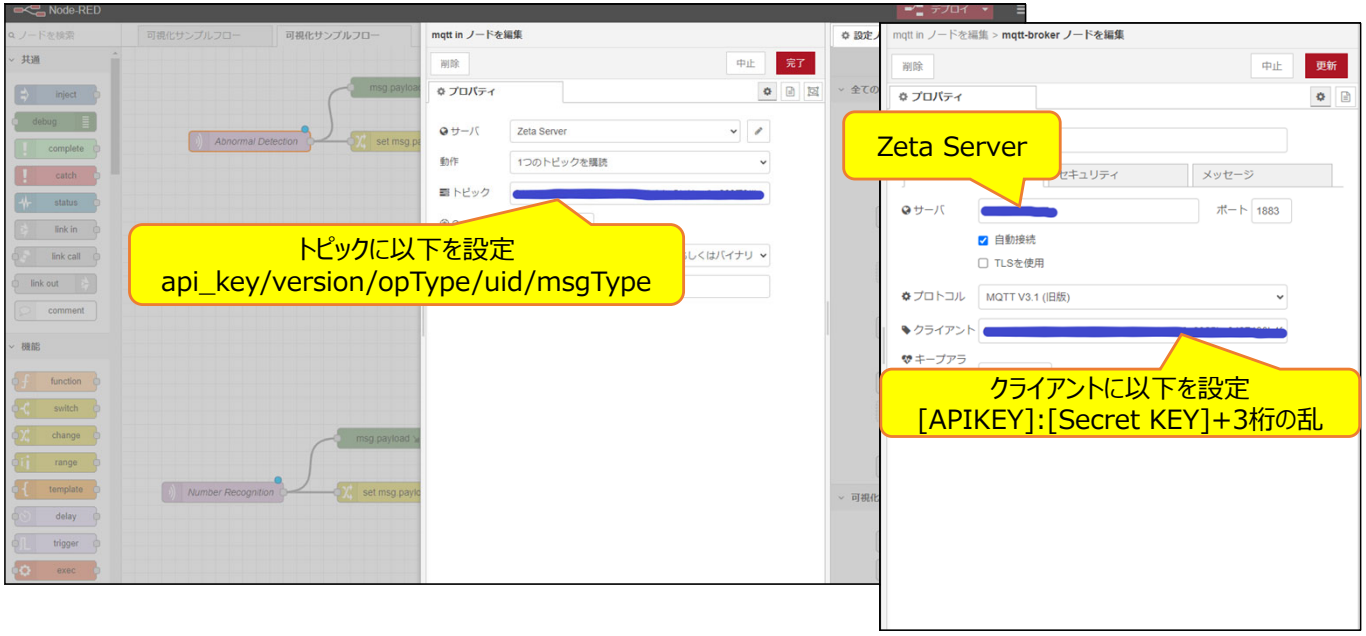
起動すると、
http://localhost:1880
でNode-REDが立ち上がります！

NodeREDの設定

「フリーブローカを使ってローカルPCで可視化しよう！」で説明したMQTTSubscribeのノードの編集を行います。

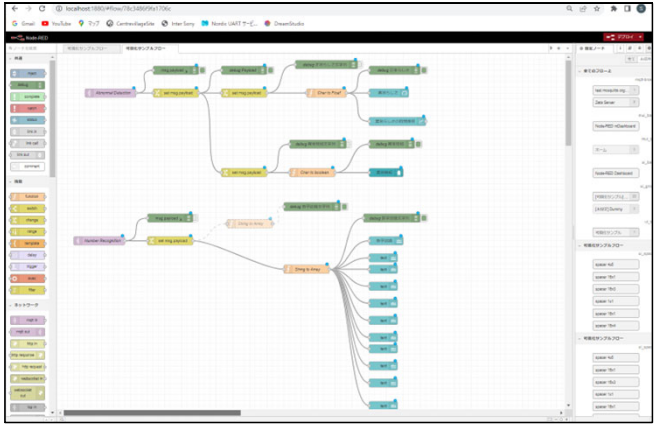
・Broker
・Topic
・Client ID
（・API Key）
（・Secret Key）

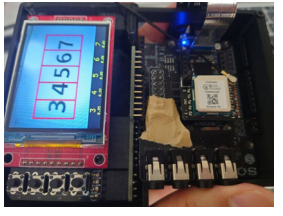
を設定してください。



NodeREDの設定

フローを設定することで、**Zetaからのデータ**を**可視化**することができます。
今回は**AI**を使ったパイプの**異常検知**と**数字認識**をやりました。





Spresenseであれば、様々なLPWAを簡単に試すことができます！

是非、気になるLPWAに挑戦してみてください！