ROHM GROUP



Lazurite Sub-GHz/ Lazurite Pi Gatewayの使い方

2015年6月27日 新規事業開拓プロジェクト 齋藤 栄一 ラピスセミコンダクタ株式会社



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

Lazuriteの開発環境
 Lazurite ソフトウエアダウンロード
 Lazurite IDEのインストール
 「Lチカ」プログラムの作成
 Lazurite Pi Gatewayのセットアップ
 920MHz無線とゲートウェイの使い方
 (参考) クラウドへの接続方法





1. Lazuriteの開発環境

下図はLazute Sub-GHzとLauzite Pi Gatewayの開発環境全体のイメージ です。ファイルをダウンロードし、ボードを組立て、周辺機器との接続さえ行え ば、誰にでも簡単に使用することができます。皆さんはアイディアを実現させる ことに注力できるはずです。





Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

モニタ端末

2-1. Lazurite関連ファイル ダウンロードサイト

♥Lazurite)

ラズライト(青金石/せいきんせき)とは ラピスセミコンダクタ 株式会社の社名のもととなっているラピスラズリの主成分です。

関連ファイルダウンロード

LazuriteはIoTを簡単に実現することができる開発キット/リファレンスデザインです。

Lazurite関連 ドキュメント	HTML形式のドキュメントー式: document.zip ファイルをダウンロードし、ZIPファイルを解凍後、 top.htmlをプラウザで開いて使用してください。	▶ ダウンロード
Lazurite IDE	Lazurite Sub-GHzの間発環境です。 インストール方法については、documentを参照してください。 LazuriteIDE¥hardware、LazuriteIDE¥librariesは、Lazurite Sub-GHzに 搭載しているマイコン ML620Q504用のソースコードであり、LGPL(Lesser GPL)に基づき配布しています。 LazuriteIDE¥BINは、それぞれのソフトウエアのライセンスに基づき再配布し ています。それ以外はGPLに準じてソースコードを配布予定です。 ソフトウエアに対する保証は行っておりません。 自己責任で使用してください。	Lazurite IDE ver0.9.5 インストーラー ・ グウンロード
回路図	Lazurite Sub-GHz Rev.1 回路図(PDF)	▶ ダウンロード
Lazurite Pi Gateway	Lazurite Pi Gatewayを使用するためのRaspberry Pi用イメージファイルで す。 Raspberry Piの種類に応じたファイルをダウンロードしてください。 また、インストール方法やソフトウエアの使用方法はドキュメントを参照 してください。 Raspberry Pi用のイメージファイルはGPLに準じて配布しています。 また、ソースコードはイメージファイル内に含まれています。 ソフトウエアに対する保証は行っておりません。 自己責任で使用してください。	Raspberry Pi B+/A+用 Rasbian イメージファイル (linux-rpi-3.18.yから作成) ・ ダウンロード Raspberry Pi2用 Rasbian イメージ ファイル (linux-rpi-3.18.yから作成) ・ ダウンロード

下記のLazurite専用サイトか らドキュメントやソフトウェア を自由にダウンロードするこ とができます。

Lazurite IDEはLazurite Sub-GHzを動かすソフトウェ アを開発するためのツールで す。

Lazurite Pi Gatewayは Raspberry Piの拡張ボード なのでRaspberry Piのソフト ウェア(Linuxディストリビュー ション)を含めたイメージファ イルです。

http://www.lapis-semi.com/lazurite/jp/download.html



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

2-2. その他に必要なファイル

開発をするにはLazurite 専用のファイル以外に、下記のフリーソフトが必要です。

➤ USBドライバのインストール USBドライバとは、Lazurite Sub-GHzとパソコンを接続するために必要な ドライバです。事前にFTDIのホームページからUSBドライバをダウンロード し、インストールしてください URL: <u>http://www.ftdichip.com/Drivers/VCP.htm</u>

▶ テキストエディタ

Lazurite IDEは、テキストエディタの機能を搭載していません。ソースファイル を編集するためのテキストエディタ(サクラエディタ、Vivi、秀丸など)をご準備 ください。

> イメージファイルの書込みツール Lazurite Pi GatewayのイメージファイルをマイクロSDカードに書き込むため にツールです。

※既に使い慣れたツールがインストールされている場合は必要ありません。

ラピスセミコンダクタ株式会社



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

3. Lazurite IDEのインストール



ダウンロードサイトから 取得したセットアップファ イル(setup_001001_ 2015Jun15.exe)を実行 してください。

実行後、図のようにイン ストーラが起動しますの で順番に実行してくださ い。

インストールは数分で完 了します。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

<u>4.「Lチカ」プログラムの作成(1/6)</u>

ここではLEDを1秒間隔でオンオフさせるプログラムの実例を用いて説明します。

1. test_led.cを作成 test_led.cを作成して、デスクトップのフォルダに保存します。 (ここでは、時間短縮のため予め保存しておきました。)



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

PIS

4.「Lチカ」初期設定(2/6)

2. Micro USBケーブルでPCとボードを接続します。 3. Lazurite IDEを起動し、初期設定を行います。 3-1.「Lチカ」プログラムを開く 3-2. 通信ポートの設定 (ツール > シリアル通信> 通信ポート) 3-3. ボードの設定 ・マイコンボードは、「Lazurite Sub-GHz rev2」を選択 (ツール> マイコンボード) ・オプションは「Lazurite Basic」を選択します。(ツール > オプション) なお、920MHz無線を使用するときに、「Lazurite Sub-GHz」を選択してください。 3-1. プログラムを開く 3-2. 通信ポートの設定 3-3. ボードの設定 - - X LAZURITE IDE LAZURITE IDE - 0 X LAZURITE IDE ファイル プロジェクト ツール ヘルプ ファイル プロジェクト ツール ヘルプ ファイル プロジェクト ツール ヘルプ マイコンボード 💽 LAZURITE Sub-GHz Rev2 マイコンボード ▝▖▋▖▁┓▐▖ オプション オプション ライブラリ ライブラリ ライブラリ シリアルモニタ シリアルモニタ 開く 通信ポート 6 COM133 シリアル通信 シリアル通信 LAZURITE IDE ファイル プロジェクト ツール ヘルプ マイコンボード オプション LAZURITE シリアルモニタ シリアル通信 LAZURITE Sub-GHz Sniffe ROHM GROUP ラピ APIS Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved SEMICONDUCTOR

4.「Lチカ」ビルドとプロフラム転送 (3/6)

4. 「マイコンボードに転送」ボタンを押します。 マイコン用プログラムの生成(ビルド)を行い、終了したら自動的にマインボードへ プログラムを転送します。 ビルドが完了し、マイコンがプログラム転送モード



プログラムの転送が終了したら、ボード上のLEDがON/OFFします。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved



4. その他の機能(4/6)



Serial関数を使用すると、printfデバックと同様なことができます。
 今後はサンプルプログラムを充実させていきます。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

4. Lazurite 関数リファレンスと制約事項(5/6)

- ・標準的なArduinoのAPIは概ね対応しています。詳細はドキュメントを参照してください。
- ・ArduinoのコンパイラはC++ですが、LazuriteのコンパイラはANSI Cです。
 - Classライブラリ、namespace「::」を含む記述ができません。 標準的な関数ではString()クラスが使用できません。
 - 変数の設定は関数の先頭のみです。

C言語の基本制御文	データ型	スケジューラ	API	
			アナログ入出力	時間制御
if else void for文 char switch unsig while int do while unsig continue long return unsig goto short #if unsig #define float #include doub	void char unsigned char int unsigned int long unsigned long short unsigned short float double struct enum typedef	setup() loop ()	analogRead() anaglogWrite()	millis() micros()
			GPIO制御	delay() sleep()
			pinMode(pin, mode) digitalWrite(pin, value) digitalRead(pin)	delayMicroseconds()
			Math	割込み制御
			lowByte() highByte() birRead() bitWrite() bitSet bitClear()	attachInterrupt() detachInterrupt() interrupts() nointerrupts()



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

ラピスセミコンダクタ株式会社

10

4. 「ビルド」によって生成されるファイル (6/6)

Lazurite IDEは実行可能プログラムを生成するときに下記のファイルを出力します。 不要なファイルは「クリーン」で消去することができます。

ファイル名	説明
test_led.ssf	拡張子.ssfは、Cソースを選択したときに自動で生成されるプロジェクトファイルで す。このプロジェクトファイルには、IDEのオプション、使用するライブラリ等が 記述されています。
_prn	コンパイル時に作成される中間ファイルです。
obj	コンパイル時に作成される中間ファイルです。
test_led.abs	拡張子.absは、デバック情報を含んだ実行可能なプログラムファイルです。µEASE という機器を使って本格的なデバックを行うときに使用します。uEASEを使用する と、プログラムステップ実行やメモリの状態を見ることができます。
test_led.bin	拡張子.binは、Binary形式で記述された実行可能プログラムファイルです。
test_led.hex	拡張子.hexは、INTEL HEX方式で記述された実行可能プログラムファイルです。
test_led.map	拡張子.mapはROM/RAMの使用状況などを記載した情報ファイルです。
test_ledc	cというファイルは、オリジナルのソースコードにデフォルトのヘッダファイル の情報を付加したものです。このファイルは実行可能プログラムを生成するときに 自動で生成されます。IDEは本ファイルを使って実行可能なプログラムを生成しま す。※このファイルはコンパイルする度に自動で生成されるため、本ファイルを修 正しても上書きされますので注意してください。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

ラピスセミコンダクタ株式会社

11

5-1. Lazurite Pi Gatewayのセットアップ ~準備~

Lazurite Pi Gateway を使用する前にRaspberry Piや備品などを準備します。

▶ Raspberry Pi (B+, A+)またはRaspberry Pi2を別途準備して下さい。
 ▶ 備品は、電源、マイクロUSBコネクタ、8GB以上のマイクロSDカード、モニター、

キーボード、マウス等を準備して下さい。









5-2. Lazurite Pi Gatewayのセットアップ~SDカード書込み~

Lazurite 専用サイトからダウンロードしたイメージファイルをマイクロSDカードに 書込みます。この作業はWindowsのパソコンで行います。 以下は、Win32DiskImagerというフリーソフトを使用した場合の説明です。

- > パソコンにマイクロSDカードを挿入します。
- ➤ Win32DiskImagerを起動します。
- ▶ イメージファイルを選択します。・・・①
- ▶ 書込み実行します。

Win32 Disk Imager
spberryPi/150401_RPi2_Image/150401_RPi2_LAZURITE.img 📄 [E:¥] 🔹
Copy MD5 Hash:
Frogress
Version: 0.9.5 Cancel Read W Exit

•••(2)

※イメージファイルは約4GBあるため、書込みに数分かかります。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

5-3. Lazurite Pi Gatewayのセットアップ ~組立て~ 14

raspberryPi2、Lazurite Pi Gateway、無線モジュールを図のように組み立てま す。raspberryPi2、Lazurite Pi Gateway は40ピンのコネクタがずれないよう に注意してください。無線ジュールとLazurite Pi Gatewayとのコネクタは小さく 、デリケートなので丁寧に接続してください。





Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved



5-4. Lazurite Pi Gatewayのセットアップ~Raspberry Pi セットアップ~15

RaspberryPiと周辺機器を接続し、マイクロUSBソケット にUSBケーブルを挿し込むと電源が入ります。

Raspberry Piログイン画面

Raspberry Pi: pi #ログイン名 Password: raspberry #パスワード

(文字は表示されません。)

▶ ディスク容量の拡張

Raspberry Pi起動後、マイクロSDカードの容量を最大限に利用できるようディスク 容量の拡張を行います。下記コマンドを実行し、Configuration Toolを起動します。

% sudo respi-config 🛛 🚽

右記画面のようにConfiguration Tool で「1. Expend Filesystem」を択しenter キーを押します。

設定が完了すると画面が切替わります。 この設定は再起動後に有効になります。

% sudo rebood





シャットダウン完了後、マイクロUSBの電源プラグを一旦、抜いてから差し直すと再起動します。



Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

ラピスセミコンダクタ株式会社

マイクロUSBソケット

6-1.920MHz無線とゲートウェイの使い方(1/3)

付属のサンプルプログラムを使って920MHzで簡単に無線通信させることができます。

このサンプルソフトはLazurite Sub-GHzからパケットを送信してLazurite Pi Gatewayで受信させるものです。下記は、送受信のイメージ図です。

尚、無線モジュールは電波認証を取得済み且つ、無線モジュール用のAPIを使え ば国内電波法に準拠したソフトウェアになっていますので安心してお使いいただ けます。

<Lazurite Sub-GHz>

<Lazurite Pi Gateway>





6-2.920MHz無線とゲートウェイの使い方(2/3)

付属のサンプルソフト(Welcome_SubGHz)を選択します。このサンプルソフトは 一定周期パケット送信を行います。





Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved



<u>6-3.920MHz無線とゲートウェイの使い方(3/3)</u>

■Lazurite Pi Gatewayは、rubyとpython3のサンプルソフトが含まれています。GUI操作が可能なpython版が簡単に使用できます。以下のコマンドを実行して、x-terminalを起動してください。% startx 🖉

■インフォコーパス様 クラウドサービスへの簡単接続機能付き "python/gateway2.py"起動画面

付属のスニファツールを実行したときのRasspberryPi画面イメージ

		Lazurite Gatewa	у		- • ×
	Start Stop	☐ Ignore address		Infocorpus	
Ch 24	+ +				
Pwr 20	•				
Rate 100	•	Serial Monitor 🗸			
PANID 0xabcd	_	SAVE	CLEAR LOG		
Jun 22 19:17:31 raspberrypi kernel: [6189.157231] [DRV-802154E]	exit remove			2
date/time headr seq	rxPan rxAddr txAddr rssi	data b'(text)'			
sudo rmmod DRV_802154 Jun 22 19:17:34 raspberrypi kernel: [sudo rmmod DRV_802154 Jun 22 19:17:34 raspberrypi kernel: [6192.227060] [DRV-802154E] exit remove				
sudo insmod /home/pi/driver/sub-ghz/D Jun 22 19:18:10 raspberrypi kernel: [Jun 22 19:18:10 raspberrypi kernel: [RV_802154.ko ch=24 rate=100 6228.481726] [MAC Address: 6228.481748] [DRV-802154E]	pwr=20 panid=0xabcd mode=0 6e 8c 03 00 90 12 1d 00 End of init			
date/time headr seq	rxPan rxAddr txAddr rssi	data b'(text)'			
2015-06-22 19:19 19:023328 0xa821 0x0 ,\x00	0 Oxabcd Ox8c6e Ox87a4 217	b'SensorCorpus,7d4fefbc8d44459	68ca662f85032a9eef9ce7780425	df9274c555a674aeb08df,2,rp_temprature,T,1,	rp_humidity,H,1
SensoRCorpus [[34724, '7d4fefbc8d4445 2015-06-22 19:19:19.227215 0xa821 0x0	968ca662f85032a9eef9ce77804 1 0xabcd 0x8c6e 0x87a4 217	25df9274c555a674aeb08df', 2, [[b'DATA,-46.9,-6.0'	'rp_temprature', 'T', 1], ['	rp_humidity', 'H', 1]]]]	
2,2 2015-06-22 19:19:41.337391 0xa821 0x0	0 0xabcd 0x8c6e 0x87a4 176	b'SensorCorpus,7d4fefbc8d44459	68ca662f85032a9eef9ce7780425	df9274c555a674aeb08df,2,rp_temprature,T,1,	rp_humidity,H,1
SensoRCorpus [[34724, '7d4fefbc8d4445 2015-06-22 19:19:41.541261 0xa821 0x0	968ca662f85032a9eef9ce77804 1 0xabcd 0x8c6e 0x87a4 177	25df9274c555a674aeb08df', 2, [[b'DATA,28.4,61.5'	'rp_temprature', 'T', 1], ['	rp_humidity', 'H', 1]]]]	
2,2 2015-06-22 19:19:53.507250 0xa821 0x0	0 0xabcd 0x8c6e 0x87a4 185	b'SensorCorpus,7d4fefbc8d44459	68ca662f85032a9eef9ce7780425	df9274c555a674aeb08df,2,rp_temprature,T,1,	rp_humidity,H,1
,\xuu SensoRCorpus [[34724, '7d4fefbc8d4445 2015-06-22 19:19:53.713148 0xa821 0x0 2,2	968ca662f85032a9eef9ce77804 1 Oxabcd Ox8c6e Ox87a4 184	25df9274c555a674aeb08df', 2, [[b'DATA,28.5,60.4'	'rp_temprature', 'T', 1], ['	rp_humidity', 'H', 1]]]]	

<Lazurite Pi Gateway>



■受信したデータがコンソール画面 に表示されます。

■Lazurite Sub-GHzにInfoCorpus と接続するためのコードを追加する と、InfoCorpusのロゴが白くなり、ク ラウドサービスにデータを格納する ことができます。 (巻末の参考資料をご参照ください。)





ROHM GROUP

Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

(参考)クラウドへの接続方法~Lazurite Sub-GHz用コード(1/2)~

Lazurite Sug-GHzからセンサーコーパス用のヘッダ情報を付加したパケッ トを送信すると、Lazurite Pi Gateway はスニファツールを介して" SensorCorpus"クラウドシステムへデータを転送します。 最後にカンマが必要 これが無いと1が文字列として認識されません。 センサーコーパス用のヘッダ情報を付加したパケットを送信するための手順 は以下の通りです。 #define SHT21 TEMP 0 SensorCorpusの設定情報であることを示すヘッダ #define SHT21_HUM 1 const char hdr[] = "SensorCorpus,7d4fefbc8 74c555a674aeb08df,2,rp_temprature,T,1,rp_humidity,H,1,"; 「セッションコード」と「センサーの数」 #define SUBGHZ CH 24 #define SUBGHZ_PANID 0xABCD を指定 #define SUBGHZ TXADDR 0x8C6E 以下の順番でセンサの設定情報を指定 Sensor CorpusのセンサーID short get sht21(unsigned char mode); 1. 2. 単位 ゲートウェイの設定に合わせてください データの数 3. SUBGHZ CH = 使用する周波数 センサーの数を2つに設定しているので、 SUBGHZ PANID = PANID void setup(void) センサーの設定情報がもう一つ続いています。 SUBGHZ TXADDR=ゲートウェイのアドレス Serial.begin(115200); Sub-GHzのモジュールを初期設定するためのAPI SubGHz.init(); pinMode(LED,OUTPUT); digitalWrite(LED,HIGH); SubGHz.begin(SUBGHZ_CH, SUBGHZ_PANID, SUBGHZ_100KBPS, SUBGHZ_PWR_1MW); digitalWrite(LED,LOW); SubGHz.send(SUBGHZ_PANID,SUBGHZ_TXADDR, (unsigned char *)&hdr, sizeof(hdr),NULL); digitalWrite(LED,HIGH); Sub-GHzのモジュールでパケット送信するためのAPI SensorCorpusの初期設定情報を送信 ROHM GROUP ラピスセミコンダクタ株式会社

Confidential © 2015 LAPIS Semiconductor Co., Ltd. All Rights Reserved

ΔΡΙς SEMICONDUCTOR

(参考)クラウドへの接続方法~Lazurite Sub-GHz用コード(2/2)~

void loop(void)

{

short temp,humi; char send_data[32]; double d_temp, d_humi;	
Print.init(send_data,sizeof(send_data));	
// Initializing SubGHz.begin(SUBGHZ_CH, SUBGHZ_PANID, SUBGHZ_100KBPS, SUBGHZ_PWR_1MW); <	920MHzの初期設定
temp = get_sht21(SHT21_TEMP); humi = get_sht21(SHT21_HUM);	温度/気圧データの取得
<pre>// convert data from short to double d_temp = temp; d_temp = d_temp / 256; d_humi = humi; d_humi = d_humi / 256; Print.p("DATA,"); Print.p(","); Print.p(","); Print.p(","); // sending data digitalWrite(LED,LOW); SubGHz.send(SUBGHZ_PANID,SUBGHZ_TXADDR, (unsigned char *)&send_data, Print.len(),NULL); digitalWrite(LED,HIGH); // close SubGHz.close(); Sub-GHzのモジュールを省電力化</pre>	固定小数点 → 浮動小数点変換 -るためのAPI
sleep(60*1000); <	
return;	

ROHM GROUP

}