

AVR-IoT開発基板を使うAWSクラウドでのBLE感知器の高速試作

序説

著者: Alin Stoicescu, Microchip Technology Inc.

この応用記述はBLE(Bluetooth[®] Low Energy)感知器をクラウドに接続する方法を記述し、簡単な応用例を含みます。 プロジェクトはATmega4808マイクロコントローラと温度と光の感知値を含むAVR-IoT WG開発基板用に設計されています。 応用はマイクロコントローラがRN4871 Click基板を使ってクラウドに接続することができる既成システムを意図されます。Click基 板はAVR-IoT WG開発基板のmikroBUS™に取り付けられ、Microchip RN4871 BLE単位部を組み入れます。

応用が開始すると、マイクロコントローラは感知器に対してBLE単位部で2つの特性を作成します。マイクロコントローラは5秒毎 に感知器からデータを読んで各々の値で特性を更新します。情報をクラウト、へ送信するのに中継器が必要とされます。 中継器はBLE単位部の終端節点との接続を確立してイサーネットまたはWi-Fi®を通して局所インターネット網からクラウト、サー ビスに接続します。5秒毎に中継器は特性を読んでそのデータをクラウト、に掲載します。

これがBLE、Wi-Fi、イサーネットの単位部を包含して使い易く、良い資料を提供するため、中継器としてRaspberry Pi(RP i)3 B+型基板が使われます。

クラウト、サービスにはAmazonウェブ、サービス(AWS)が使われます。これは使用者にクラウトとの相互通信を許すAWS IoT Gree ngrassと呼ばれる中継器核ソフトウェアを提供します。使用者は単にLambdaで応用プログラミング、インターフェース(API:Applicati on Programming Interfaces)を利用することにより、話題の掲載と購読をすることができます。

マイクロコントローラで使われる周辺機能用ソースコート、はAtmel STARTを使って生成されます。このコート、生成部はプロジェクト、ファイル、ソースコート、の構造を作成します。Atmel STARTは全ての周辺機能とRN4871 Click基板のような開発拡張基板に対する支援を提供します。

以下のGitHub貯蔵庫は特性を読んでそのデータをクラウドに掲載する中継器によって使われるLambdaと共にAVR-IoT WG開発基板に対するプロジェクトのソース コードを含みます。



GitHubでコート、例を見てください。 貯蔵庫を閲覧するにはクリックしてください。

本書は一般の方々の便宜のため有志により作成されたもので、Microchip社とは無関係であることを御承知ください。しおりの[はじめに]での内容にご注意ください。

序≣	\								•• 1
1 1	概要 • • • • • •								•• 3
2	ハードウェア説明								•• 3
	2.1. AVR-IoT	WG開発基板 •••••							3
	2.2. MIKROE	RN4871 Click基板 ·							••• 3
	2.3. Raspberry	/ Pi 3 B+型基板 • • •							••• 4
3.	応用概要・・・								•• 4
	3.1. ソフトウェア要	[件 •••••							••• 4
	3.2. Amazon	フェフ゛サービス ソフトウェア材	^既 要 • • • • • • • • • •						••• 4
	3.3. AVR-IoT	WG開発基板ソフトウェフ	′概要 ・・・・・・			• • • • • • • • • • • • •			••• 4
	3.4. Lambda楒	·要 · · · · · · · · · · · · · · ·	•••••	• • • • • • • • • • • • •	•••••	• • • • • • • • • • • • •	• • • • • • • •		•• 13
4.	応用実演・・・・		•••••	• • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • •		• 14
	4.1. 主応用構	成設定 • • • • • • • • •	•••••	• • • • • • • • • • • • •	•••••	• • • • • • • • • • • • •	• • • • • • • •		•• 14
	4.2. 基板の準	備 • • • • • • • • • • • • • • • • •	•••••	• • • • • • • • • • • • •	•••••	• • • • • • • • • • • • • •	• • • • • • • •		•• 14
	4.3. Lambda <i>0</i> .	作成と読み込み				• • • • • • • • • • • • • • • •	• • • • • • • •		•• 15
_	4.4. クラウト での)感知器7-9可視化					• • • • • • • •		• 16
5.	結び・・・・・・						•••••		• 16
6.	参照 *******				• • • • • • • • • • • • •		•••••	• • • • •	• 17
Mic	rochipウェフ サイト				• • • • • • • • • • • • •		•••••	• • • • •	• 18
おそ	客様への変更通	知サービス・・・・・・	• • • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • •	• • • • •	• 18
おそ	客様支援 •••••		• • • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • •		• 18
Mic	rochipデバイス コ	-ド保護機能 • • • •		• • • • • • • • • • •	• • • • • • • • • • • •		•••••	• • • • •	• 18
法的	的通知 ••••••	• • • • • • • • • • • • • • • • • • • •	•••••	• • • • • • • • • • •	• • • • • • • • • • • •	• • • • • • • • • • • •	•••••	• • • • •	• 18
商权	票 •••••	• • • • • • • • • • • • • • • • • • • •	•••••	• • • • • • • • • • • •		• • • • • • • • • • • •	• • • • • • •		• 19
DN	Vによって認証さ	れた品質管理シス	<u>-</u> Д •••••	• • • • • • • • • • • •			•••••	• • • • •	• 19
世界	界的な販売とサ−	ビス ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・							• 20

1. 概要

この応用記述はBLE物のインターネット(IoT)プロシェクトによって必要とされるハート・ウェアとソフトウェアの構成設定をどう得るかを説明します。例 プロジェクトとAWS構成設定の機能も詳述されます。

例プロジェクトでは提示されたデバイスが光と温度の感知器からのデータを読みます。光感知器はアナログ信号として光の強度を提供する 一方で、温度感知器はI²Cインターフェースを通してデータを送ります。データの遷移と展開を強調するため、データをクラウトドに送信するのに5 秒の読み取り間隔が使われました。デバイスはUSARTインターフェースを通してRN4871 BLE装置と情報を交換し、感知器に対するサービス と特性を作成することによって開始し、毎回の読み取り後にそれらを更新します。

Raspberry Pi基板はBluetooth装置に接続し、Greengrass核を使って特性を読んでそのデータをクラウトに掲載します。GreengrassはRas pberry Pi基板のハートウェア構成部品を使う能力があるAWSによって提供された中継器ソフトウェアです。従って、この中継器の核は近くの BLE節点装置と相互通信することができ、クラウトに安全な通信を提供することができます。

事前要件:

- ・統合開発環境(IDE:Integrated Development Environment):
 - 1. インストールされた最新のデバイス一括を持つ(このプロジェクトで使われる)Atmel Studio Ver.7
 - 2. XC8 Ver. 2.05コンパイラを持つMPLAB® X Ver. 5.15
- ・Microchip AVR-IoT WG開発基板
- ・ MIKROEからのBluetooth Low Energy RN4871 Click基板
- Raspberry Pi 3 B+型
- ・ Amazon ウェフ^{*} サーヒ^{*}ス(AWS) アカウント

2. ハート・ウェア説明

2.1. AVR-IoT WG開発基板

AVR-IoT WG開発基板は物のインターネット環境で使われることを意図されています。この基板はATmega4808マイクロコントローラ、通信用のATWINC1510 Wi-Fi単位部、安全性のためのATECC608A CryptoAuthentication™デバイス、データ採取用のTEMT6000光感知器とMCP9808温度感知器のような多数のIoT指向デバイスを組み込んでいます。

基板はより良い使用者体験と応用での容易な相互やり取り用の4つの使用者LEDと2つの使用者釦、それと基板を現場で使うことができるようにリチウムポリマー(LiPo)充電器を統合します。基板は多様性を提供して使用者にそれらのシステムの強化を許すmikroBUS差し込み口も特徴です。この応用記述で提示されるように、使用者は感知器から駆動装置まで、または新しい通信単位部までも、いくつかのClick基板を追加または積み重ねることができます。このプロジェクトではAVR-IoT WG開発基板にBLE Click単位部が接続されました。

基板はUSBまたはJ101電池プラグに接続された外部のLiPo電池から給電することができます。ATmega4808マイクロコントローラとRN4871 Click基板は3.3Vで給電されます。回路図についてのより多くの詳細は次のリンクで見つけることができます。

 $http://ww1.microchip.com/downloads/en/DeviceDoc/AVR-IoT_WG_Schematics.pdf$

MicrochipのATWINC1510はIEEE[®] 802.11b/g/n IoT網制御器です。これはWi-Fiと網の能力をもたらす既存のMCU解決策に対して 理想的なアドオンです。ATWINC1510は内部フラッシュ メモリだけでなく、UARTとSPIを含む複数の周辺機能も提供します。

MicrochipのATECC608Aは高度な楕円曲線暗号(ECC:Elliptic Curve Cryptography)を持つ安全素子です。これはIoTシステムに対し て楕円曲線デジタル署名算法(ECDSA:Elliptic Curve DigitalSignature Algorithm)署名照合認証と共に暗号化/復号用の鍵共有を提 供する超安全な楕円曲線ディフィー ヘルマン(ECDH:Elliptic Curve Diffie-Hellman)安全規約を統合します。加えて、ATECC608Aは安全 性に基づいてハートウェアを強化して統合された新暗号化規格(AES:Advanced Encryption Standard)ハートウェア加速器を提供します。

この基板はIoT開発を促進するために選ばれましたが、応用記述の目的がBLEデータをクラウトに送信することなので、ATWINC1510単位部とATECC608A CryptoAuthentication™ デバイスはこのプロジェクトで使われません。

2.2. MIKROE RN4871 Click基板

RN4871 Click基板はmiktoBUS差し込み口に挿入することができるBLE基板です。これは3.3Vから給電され、UART周辺機能を使うこ とによってホストマイクロコントローラへ接続され、ASCII命令経由での制御を許します。基板は送出準備可(RTS:Ready To Send)と送出解 消(CTS:Clear To Send)のUART信号に対する支援とチャネルも提供し、これは任意選択でこの応用では使われません。

RN4871は低電力Blutooth(Bluetooth low-energy)単位部です。これはBluetooth 4.1に基づく製品よりも最大2.5倍の単位処理量の改善ともっと安全な接続を届けるBluetooth階層4.2版に基づく完全に統合されたソフトウェアを持ちます。

この単位部はUARTから仲間のBLE装置へデータを転送するのに使われる透過UARTを支援します。既定ボーレートは115200シンボル/s ですが、命令インターフェースから変更することができます。UART上でのデータ交換は送信用の1つと受信用の1つで、2つのチャネルを通し て行われます。

この単位部は汎用属性特質特性(GATT)を支援し、最大5つの公開と4つの私的の使用者定義GATTサービスを実装することができ、 各サービスは8つまでの特性を提供します。

BLEが許されたスマートフォンやBluetoothインターネット中継器とインターフェースされると、応用は世界の何処からでも監視、制御、更新することができます。従って、RN4871単位部はIoT応用に対して完璧です。

2.3. Raspberry Pi 3 B+型基板

Raspberry Pi 3 B+型開発基板はLinux™オペレーティング・システムの様々な配給、承認済み開放ソース ソフトウェア ライブラリ、トライハ、使用者に 対する応用を走らせます。これらは他の装置との容易なインターフェースを許します。基板はIoTに有利な終端節点との通信用の組み込 みBLE単位部とクラウト、接続と通信用のWi-Fiとイサーネット ソケットを提供します。RPi 3 B+はARM V717°ロセッサ基本構造を持ち、この実演 応用で中継器として使われます。使用者はRPiのオペレーティング・システムと中継器ソフトウェアのために大きさで最低8GハイトのmicroSDカート が必要です。

3. 応用概要

3.1. ソフトウェア要件

MicrochipのMPLAB[®] XとAtmel Studioはマイクロコントローラ用の応用開発に使われる統合開発環境(IDE)です。この資料で提示された応用はAtmel Studio 7版で開発されましたが、このプロジェクトはFile(ファイル)⇒Import(インホート)⇒Atmel Studio Project(Atmel Studioプロジェクト)からMPLAB Xでインポートすることもできます。このプロジェクトの正しい機能性のためにMPLAB X IDE 5.15版とXC8コンハパラ 2.05版が必要とされます。

Atmel Studio 7版 IDEはC/C++またはアセンフリコートで書かれた使用者応用を書き、構築してデバッグするための継ぎ目の無い使い易 い環境を与えます。Atmel Studio 7版は8ビットと32ビットのAVRマイクロコントローラ、新しいSoC無線系統、SAMマイクロコントローラを支援し、継 ぎ目なしにデバッガと開発キットに繋げます。

Microchipはウェブに基づくソフトウェアとしてAtmel Studioで利用可能なAtmel STARTと呼ばれるコート生成用の道具を提供します。Atmel STARTの両版共、インターネット接続を必要とします。Atmel STARTはAtmel Studio 7版、IAR Embedded Workbench[®]、または汎用makfi le生成用のコート、プロジェクト生成を支援します。

使用者はAtmel Studioがインストールされた最新のデバイス一括を持つことを保証しなければなりません。これは次の段階を通して行うことができます。使用者はTools(ツール)⇒Device Pack Manager(デバイス一括管理部)でクリックしなければならず、その後に'Check for Upda tes(更新調査)'を押し、'Install(インストール)'タブで'Install all updates(全ての更新をインストール)'を選んでください。

3.2. Amazon ウェフ サービス ソフトウェア概要

Amazonウェブ サービスを使うため、使用者はアカウントが必要です。これは無料で作成することができ、複数の装置に対して最大1年間、 中継器ソフトウェアとAWSの使用を許します。クラウト アカウントと(Greengrassと呼ばれる)中継器構成設定はこの資料の焦点ではなく、詳細 に説明されません。AWSは既にAWS IoT Greengrassでの開始に際してで見つけることができ、十分に文書化されて書かれた開発者 の手引きを提供しています。Greengrass開発者の手引きは提示される情報の焦点に基づいて章で分割されます。それらの章の全て は単位部と呼ばれます。

単位部1はRaspberry Piに対するオペレーティング システムのダウンロート と読み込み方法、基板とコンピュータ間の接続確立方法を説明します。 この単位部は基板がGreengrassを安全に走らせることができるように行わなければならない構成設定も段階的に説明します。

単位部2はRaspberry Piで中継器ソフトウェアをダウンロードしてインストールする方法を記述します。これはクラウド接続に必須の公開鍵と私的 鍵、それと証明書と共に来ます。この単位部は中継器の開始方法と中継器を扱う処理を調べる方法の詳細を提供します。

従って、これら2つの単位部はRaspberry Pi基板の構成設定とGreengrass核での開始に必須です。

単位部3はLambdaでのクラウト・とRPi間のデータ交換に集中します。この単位部はRaspberry PiにLambdaを追加して展開する方法を説明します。AWS Lambdaはサーハーを提供して管理することなしにコートを走らせる機能です。Lambdaは特定の話題(トピック)でメッセージ待ち行列遠隔測定伝送(MQTT:Message Queuing Telemetry Transport)メッセージを送ることによってクラウト・にアクセスすることができます。 この送出手順は掲載(パブリッシング)と呼ばれます。顧客は特定の話題を購読(サブスクライブ)してメッセージを受け取ることができます。

この資料で提示される応用の目的がクラウト、にBLEデータを送り、これがLambdaを使うことによって実行されるので単位部3も必要とされます。他の単位部はこの応用記述全体を通して必要とされないか、または使われません。

注: このプロジェクトではGreengrass核(GGC) 1.7版が使われました。

3.3. AVR-IoT WG開発基板ソフトウェア概要

このプロジェクトはAtmel Studioを使って開発され、必要とされる周辺機能はAtmel STARTを使って構成設定されました。使用者はAtm el Studio IDE内でfile(ファイル)⇒New(新規)⇒Atmel Start Project(Atmel STARTプロジェクト)を押すことによって新しいプロジェクトを作成 することができます。使用者はデバイスを選別してAVR-IoT WG開発基板で使うマイクロコントローラであるATmega4808を選らばなくてはな りません。

プロジェ小全体はGitHub貯蔵庫で提供されます。本項の目的はAtmel STARTコード生成部と生成されたコードの構造を使用者に習熟させて全ての周辺機能の使い方とそれの構成設定を説明することです。

REATE NEW PROJECT								
elect device or board before creat	ing a new project. You can filt	er devices and boards by v	vhat software you nee	d and also with hardwa	ire requirements su	ch as memory si	zes.	
FILTERS		RESULTS						
HARDWARE	GØ	atmega4808	\otimes	Show all O Show	only boards OS	now only devices	i.	
		Name	Architecture	Package	Pins	Flash	SRAM	0
		ATmega4808-XFR	AVR	SSOP28	28	48 KB	6 KB	3
-ind software		ATmega4808-MFR	AVR	VQFN32	32	48 KB	6 KB	ß
MIDDLEWARE	\odot	ATmega4808-AFR	AVR	TQFP32	32	48 KB	6 KB	C.
+ Crypto	*	ATmega4808 A	AVR IOT WG					3
Fan Click	0 0							
+ Graphics	*							
DRIVERS	\odot							
AC	0 0							
ADC	0 0							
CRC	0							
DAC	0 0							
Digital Glue Logic	0 0	4 of 1006 boards and d	levices		CREATE			

周辺機能は以下のように構成設定されなければなりません。

- 1. CLKCTRLドライハで主クロックを次のように構成設定してください。
 - ・CLKSEL: Main Clock Source(主クロック元): 20MHz Internal Oscillator (OSC20M) (20MHz内部発振器)
 - ・ PEN: Prescaler enable(前置分周器許可): 枠をチェック
 - PDIV: Prescaler division(前置分周数) : 10

int bot thinke could c							
c	ilkctrl 🄅		BOD 🔅	SLPCTRL		CPUINT	
			CLF Clock Cor	CTRL			
GENERAL		COMPONEN	T SETTINGS		COMP	ONENT SIGNALS	
User guide		Driver:	Drivers:CLKCTR	L:Init			
DRIVERS:CLKCTRL:INIT (CLKC	TRL) CONFIGUR/	ATION ON CLK	CTRL				
MAIN CLOCK CONFIGURATION			Enable: 🗸	OSCULP32K DIV 32			Enable:
CLKSEL: Main Clock Source:	20MHz Inte	rnal Oscillator (O	SC20M) ~	CLKSEL: Main Clock Source:		32KHz Internal Ultra Low Po	wer Oscillator
PEN: Prescaler enable:	0			16MHZ OSCILLATOR CONFI	GURATION		Enable
PDIV: Prescaler division:	0 10		~				Enable.
LOCKEN: lock enable:	0			KUNSTUBT, KUITStandby.	Ŭ		
CLKOUT: System clock out:	0			32.768KHZ CRYSTAL OSCILL	ATOR CON	FIGURATION	Enable:
20MHZ OSCILLATOR CONFIGURA	TION		Enable: 🗸	TCD0 CLOCK CONFIGURATIO	N		Enable:
RUNSTDBY: Run standby:	0			CLKSEL: clock select:	0	20MHz Internal Oscillator (O	ISC20M)
32KHZ INTERNAL ULP OSCILLATO		4	Enable: 🗸	SYNCPRES: Syncronization p	rescaler: 🕜	1	
	•						
RUNSTDBY: Run standby:	•						
RUNSTDBY: Run standby:			Enable:				
RUNSTDBY: Run standby: EXTERNAL CLOCK CONFIGURATIO	DN 22768		Enable: 🗸				
RUNSTDBY: Run standby: EXTERNAL CLOCK CONFIGURATIO External Oscillator Frequency:	32768		Enable: 🗸				
RUNSTDBY: Run standby: EXTERNAL CLOCK CONFIGURATIO External Oscillator Frequency: SYSTEM CLOCKS	32768		Enable: 🗸 dec v Enable: 🗸				
RUNSTDBY: Run standby: EXTERNAL CLOCK CONFIGURATIO External Oscillator Frequency: SYSTEM CLOCKS CPU clock source:	32768 Main Clock	(CLK_MAIN)	Enable: 🗸 dec v Enable: 🗸				

注:結果の主システム クロックは2MHzの周波数を持ちます。

2. CPUINTドライバで割り込みを構成設定してください。

割り込みは通信が必要な時に必ず行われるように計時器とUARTデータ交換に使われます。割り込み機能を許可するには全体割り込みが許可されることが必要です。

・CPU_SREG: Global Interrupt Enable(全体割り込み許可):枠をチェック

図3-	3.割り込み構成設定		
	MY SOFTWARE COMPONENTS		?
BOARD	CLKCTRL	BOD 🔅 SLPCTRL 🔅 CPUINT 🔅	Ŷ
Pash 🖉		CPUINT CPUINT	\otimes
NMUX	GENERAL User guide DRIVERS:CPUINT:INIT (CPUINT) CONFIGURAT	COMPONENT SETTINGS COMPONENT SIGNALS Drivers: Drivers:CPUINT:Init TION ON CPUINT	C
	GLOBAL INTERRUPT CONFIGURATION	INTERRUPT SCHEDULING CONFIGURATION	
CLOCKS	CPU_SREG: Global Interrupt Enable:	LVLORR: Round-robin Scheduling	
$\overline{\bigcirc}$	Include Pin Change ISR harness in driver_isr.c:	LVLOPRI: Interrupt Level Priority:	hex v
	CVT: Compact Vector Table: Image: Compact Vector Table: IVSEL: Interrupt Vector Select: Image: Compact Vector Select:	LVL1VEC: Interrupt Vector with High 🛛 0x0 Priority:	hex v

- 3. ADCドライバを追加して構成設定してください。
 - これは'Add software component(ソフトウェア構成部品を追加)'を押して'ADC'を探すことによって行うことができます。

]3-4. ADCソフトワェ/構成部品追加		
DD SOFTWARE COMPONENTS		
dc		
Name	Description	Add
– 🛸 Middleware		
– 🗁 Mixed Signal Click		
SADC Click	ADC click carries the MCP3204 12-bit Analog-to-Digital converter which communicates with the target microcontroller over SPI interface.	÷
SADC2 Click	ADC 2 click carries MCP3551/3,a 22-bit ADC with automatic internal offset and gain calibration. It has total unadjusted error of less than 10 ppm, and low-output noise of 2.5 uV RMS.	÷
SADC3 Click	ADC3 click carries MCP3428 16-bit multichannel analog-to-digital converter. It has four pairs of screw terminals onboard, letting you access the chip's four differential input channels.	÷
– 🗁 Sensors Click		
Surrent Click	Current click is used for measurement of electric current.lt features INA196 current shunt monitor, MCP3201 12-bit ADC, MAX6106 voltage reference as well as two screw terminals.	÷
- Ö Drivers		
O ADC	Analog-to-digital converter (ADC).	\oplus
O VREF	Voltage Reference for ADC, DAC, and AC peripherals, providing programmable voltages for each reference source.	Ð
	Add component(s)	Cancel

ADCは光感知器によって生成されたアナログ出力を読むのに使われ、次のようにAVR-IoT WG開発基板の現在のピン配列構成に 基づいて構成設定されなければなりません。

- Driver
- AIN/5
- RESSEL: ADC Resolution(ADC分解能)
- ・ MUXPOS: Analog Channel Selection Bits(アナログ チャネル選択): ADC input pin 5
- ENABLE: ADC Enable(ADC許可)

・ PRESC: Clock Prescaler(クロック前置分周器)

- : Drivers:ADC:Basic
- : PD5枠をチェック
- : 10-bit mode
- : ADC Input : 枠をチェック

: VDD

- :
 - : CLK_PER divided by 16 (16分周したCLK_PER)
- ・REFSEL: Reference Selection(基準電圧選択)

	MY SOFTWARE COMPONENTS				?
🕎 разнвоакр	CLKCTRL	AL BOD	DC_0 🔅	CPUINT	
XUMUI		Al MCode basic driver: Peripheral initializa	DC_0 tion + API to support simple AD0	C use-case	\otimes
	GENERAL	COMPONENT SETTINGS		COMPONENT SIGNALS	
	User guide	Driver: Drivers:ADC:Ba	sic ×	AIN/0: PD0	
S		CLOCKS		AIN/1: PD1	
CLOCH		ADC: Main Clock (CL	K_MAIN) (2 MHz)	AIN/2: PD2	
ē	Remove component		,	AIN/3: PD3	
				AIN/4: PD4	
				AIN/5: PD5	
				AIN/6: PD6	
				AIN/7: PD7	
				AIN/14: PF4	
				AIN/15: PF5	
	MY SOFTWARE COMPONENTS				?
ARD					
HBOA	DRIVERS:ADC:BASIC (ADC BASIC) CONFI	SURATION ON ADCO			C
DAS	ADVANCED CONFIGURATION		BASIC CONFIGURATION		
oốt	DBGRUN: Debug run:		ENABLE: ADC Enable:		~
×	RESSEL: ADC Resolution: 0 10-bi	t mode 🗸 🗸	REFSEL: Reference Selection:	VDD	~
INMU			SAMPLING CONFIGURATION	L	Enable:
ē	MUXPOS: Analog Channel Selection @ ADC Bits:	nput pin 5 V			
	INTERRUPT CONFIGURATION				
CLOCKS	RESRDY: Result Ready Interrupt 🛛 🛛 🗌 Enable:				

4. RN4871用にUSARTドライバを追加して構成設定してください。

MicrochipはRN4871 Click基板用の支援を提供します。これは'Add software component(ソフトウェア構成部品を追加)'を押して'R N4871'を探すことによって行うことができます。

図3-6. RN4871ソフトウェア構成部品追	自加	
ADD SOFTWARE COMPONENTS		×
rn4871		
Name	Description	Add
- 🛸 Middleware		
– 🗁 Wireless Connectivity Click		
📚 RN4871 Click	RN4871 click carries RN4871 Bluetooth 4.2 low energy module from Microchip. It uses ASCII Command Interface over UART for communication with target MCU.	÷
	Add component(s)	ancel

コート、生成部はこの基板によって必要とされる構成設定の殆どを提供します。使用者によって実行されなければならない変更はそれらがmikroBUSに合わせることができるようにピンの位置に関連されるものだけです。このUSARTの実体を認識できるようにUSAR T_RN4871に改名されます。全体割り込みは前で許可されていて通信のために行われ、故にUSARTは割り込み要求(IRQ)動作に設定されなければなりません。

• RST PIN : PA0



- - © 2019 Microchip Technology Inc.

Rename component(構成部品改名)	:	USART_AN4871
--------------------------	---	--------------

- - : PC0

: PC1

・Mode(動作形態)

• RXD • TXD

: Async IRQ mode (非同期割り込み動作)

図3-8. RN4871 USART構成設定	
------------------------	--

	MY SOFTWARE COMPONEN	TS					?
👰 разнвоакр	CLKCTRL	- ¢	BOD 🔅	USART_RN4871	°	CPUINT 🔅	
PINMUX			USART UART drive	_RN4871 r in IRQ mode			\otimes
	GENERAL	COMPONE	NT SETTINGS		COMPONEN	NT SIGNALS	
	User guide	Driver:	Drivers:USART:	Basic 🗸	RXD:	PC1 ~	
KS.	Pename component	Mode:	Async IRQ Mod	le 🗸	TXD:	PC0 ~	
CLOC		Instance:	USART1	~			
$\overline{\bigcirc}$	Remove component	CLOCKS					
		USART:	Main Clock (CL	K_MAIN) (2 MHz)			
							Q
		wobe) contrigoratio	IN ON USARTI				
	Printf support:	8	~	LBME: Loop-back Mode Ena	ible: 0		
	TX Buffer Size:	8	~	bbarron, bebag ran.	•		
	BASIC CONFIGURATION						
	RXEN: Reciever enable: 0						
	PMODE: Parity Mode:	No Parity	~				
	SBMODE: Stop Bit Mode: 0	1 stop bit	~				
	CHSIZE: Character Size:	Character size: 8 bit	~				
	Baud Rate: 0	115200	dec v				

5. I²Cドライバを追加して構成設定してください。

これは'Add software component(ソフトウェア構成部品を追加)'を押して'I²C'を探すことによって行うことができます。

ame	Description	Ado
Middleware		
- 🗁 Interface		
I2C Simple Master	Simplified I2C Master	\oplus
– 🗁 Interface Click		
SBI2C Click	USB I2C click carries an MCP2221 USB-to-UART/I2C converter which supports full-speed USB (12 Mb/s), I2C with up to 400 kHz clock rates and UART baud rates between 300 and 115200.	÷
– 🗁 Sensors Click		
Accel3 Click I2C	Accel 3 click carries the H3LIS331DL a low-power high-performance 3-axis linear accelerometer.	÷
Syro Click I2C 🤝	Gyro Click features L3GD20 three-axis digital gyroscope module capable of providing the measured angular rate, as well as ambient temperature.	÷
– 🗁 Storage Click		
SEEPROM Click	It features 24C08WP - a highly reliable, high performance CMOS technology serial 8K EEPROM in DIP packaging. The 24C08WP EEPROM uses the I2C addressing protocol.	÷
SEEPROM3 Click	EEPROM3 click carries Atmel's AT24CM02 DIP-8 socket EEPROM chip with 256 KB of memory. The board communicates with the target board MCU through the mikroBUS I2C interface with speeds up to 1 MHz.	÷
O Drivers		
O 12C	Inter-Integrated Circuit (I2C), two wire serial interface usually used for on-board low-speed bi-directional communication between controllers and peripherals.	Ð

I²C温度感知器によって生成された値を読むために主装置として使われ、次のように構成設定されなければなりません。

- Driver
- ・Mode(動作形態)
- SCL
- SDA
- ・SCL Frequency(SCL周波数)
- ・Trise(上昇時間)

: Drivers:I2C:Master : Interrupt (割り込み)

- : PA3
- : PA2
- : 100000

- : 0 ・ENABLE: Enable TWI Master(TWI週装置許可): 枠をチェック

	· · · · · ·
1212-10	14(:)(新田) 14(:)(二)

	in the second second			
	MY SOFTWARE COMPONENTS		?	
ASHBOARD		I2C_0 I2C Master driver (interrupt r	mode)	\otimes
	GENERAL	COMPONENT SETTINGS	COMPONENT SIGNALS	
12	User guide	Driver: Drivers:I2C:Master	SCL: PA3	~
×	Rename component	Mode: Interrupt	SDA: PA2	~
NWN		CLOCKS		
	Remove component	TWI: Main Clock (CLK_MAIN) (2 M	/Hz)	
	DRIVERS:12C:MASTER (INTERRUPT) CONFIG	URATION ON TWIO		C
ş	BASIC CONFIGURATION	ADVANCE	DCONFIGURATION	Enable:
CLOC	SCL Frequency: 0 100000	dec v		
Ð	Trise: 0	dec v		
	ENABLE: Enable TWI Master: 🥑 🗸			
_	FMPEN: FM Plus Enable: 0			

6. PC用のUSARTドライバを追加して構成設定してください。

これは'Add software component(ソフトウェア構成部品を追加)'を押して'USART'を探すことによって行うことができます。

図3-11. USARTソフトウェア構成部。	品追加	
ADD SOFTWARE COMPONENTS		×
usart		
Name	Description	Add
- 📚 Middleware		
SWIFI_AVR	WiFi is a local area wireless computer networking technology. Atmel SmartConnect WINC1500 network controller SoC allows WiFi for any MCUs though an SPI/USART interface.	÷
E O Drivers		
O USART	Universal Synchronous Asynchronous Receiver Transmitter (USART) to transfer data from one device to the other.	÷
	Add component(a)	ancol
	Add component(s)	ancel

このUSARTの実体はコンピュータとのデータ交換に使われます。AVR-IoT WG開発基板は使用者がコンピュータ上のシリアルCOMポートを開いてマイクロコントローラがそのUSART周辺機能でデータを受け取ることができることを意味する、USB-TTL変換器としても働くことができるデバッガを含みます。この通信は殆どサービスと特性の作成、特性の処理部取得とBLE装置のMAC7トレス取得のようなRN487 1構成設定に使われます。この通信は使用者が望む場合に記録や命令用の使用者インターフェースとしても働くことができます。ボーレートはPC側のCOMポートの値と同じである限り、どの値にも変更することができます。このUSARTの実体を認識できるようにUSAR T_TERMINALに改名されます。このUSARTの実体は次のように構成設定されなければなりません。

- ・Rename component(構成部品改名): USART_TERMINAL
- RXD TXD

- - :
- ・Mode(動作形態)
- : PF0

: PF1

- : Async IRQ mode (非同期割り込み動作)
- ・ Baud Rate(ボーレート)
- : 115200

図3-	12. PC USART構成設定	Ē					
	MY SOFTWARE COMPONE	NTS					?
SHBOARD	USART_TERMINAL UART driver in IRQ mode					\otimes	
	GENERAL	COMPON	ENT SETTINGS		COMPONEN	T SIGNALS	
121	User guide	Driver:	Drivers:USART:	Basic 🗸	RXD:	PF1 V	
×	Rename component	Mode:	Async IRQ Mod	e ~	TXD:	PF0 ~	
NMU		Instance:	USART2	~			
ā.	Remove component	CLOCKS					
÷÷		USART:	Main Clock (CL	(_MAIN) (2 MHz) 🛛 🗸			
	DRIVERS-USART-BASIC (ASYNC IRO MODE) CONFIGURATION ON USART2						
OCKS	DRIVER CONFIGURATION			ADVANCED CONFIGURATIO	N		
U U U	Printf support:			LBME: Loop-back Mode Enab	ole: 🛛 🗌		
\bigcirc	RX Buffer Size:	8	~	DBGRUN: Debug Run:	0		
	TX Buffer Size:	8	~				
	BASIC CONFIGURATION						
	RXEN: Reciever enable:						
	TXEN: Transmitter Enable:						
	PMODE: Parity Mode:	No Parity	~				
	CHSIZE: Character Size:	Character size: 8 bit	~				
	Baud Rate: 0	115200	dec v				
			dec v				

7. 感知器読み込み用Timer(計時器)ドライバを追加して構成設定してください。

これは'Add software component(ソフトウェア構成部品を追加)'を押して'Timer'を探すことによって行うことができます。

図3-13. Timerソフトウェア構成部品追加		
ADD SOFTWARE COMPONENTS		×
timer		
Name	Description	Add
= O Drivers		_
O Timer	Task timer functionality using a timer counter peripheral.	\oplus
O WDT	Watchdog Timer (WDT) for monitoring correct operation and make it possible to recover from error situations such as runaway or deadlock code.	÷
	Add component(s)	ancel

Timer(計時器)は応用作業間の遅延をもたらすのに使われます。計時器の目的は5秒を計数して時間が経過した時に割り込みを 生成することです。割り込みが到来すると、応用は感知器を読んでRN4871 BLE単位部の特性を更新し、その後に次の割り込み を待ちます。

: 枠をチェック

計時器の供給元は2MHzに構成設定された主システム クロックです。次式は定期(PER)レジスタの値を得るのに使われます。

 $PER = CLKSEL \times 5\% = \frac{SYSTEM_CLOCK}{256} \times 5\% = \frac{2MHz}{256} \times 5\% \Rightarrow 39062 = 0x9896$

計時器は次のように構成設定されなければなりません。

・ CLKSEL: Clock Selection(クロック選択)

- -

・PER: Period(周期)

: System Clock/256 (システム クロック/256) : 0x9896 (割り込み)

・Include ISR harness in driver_isr.c(driver_isr.cにISR制御を含む): 枠をチェック

OVF: Overflow	Interrupt(溢れ割り込み)
---------------	-------------------

×3-	14. Timer 体成設定							
	MY SOFTWARE COMPONE	NTS						?
SHBOARD	TIMER_0 TCA Init driver in Normal Mode						\otimes	
] DA				COMPONENT SIGNALS				
eốt	🖠 User guide	Driver:	Drivers:TCA:Init	~	WO/0:		~	
		Mode:	Normal Mode	~	WO/1:		~	
MUX		CLOCKS			WO/2:		~	
PIN	🔟 Remove component	TCA:	Main Clock (CLF	(_MAIN) (2 MHz)	WO/3:		~	
			L	,	WO/4:		~	
					WO/5:		~	
CLOCKS	DRIVERS:TCA:INIT (NORMAL MOD	E) CONFIGURATION ON T	TCA0					C
<u> </u>	CONFIGURATION			INERRRUPT CONFIGURATIO	N			
Ű	ENABLE: Module Enable:			Include ISR harness in driver	_isr.c: 🗸			
	DBGRUN: Debug Run: 🕜			CMP0: Compare 0 Interrupt:	0			
	ALUPD: Auto Lock Update: 0		CMP1: Compare 1 Interrupt:	0				
	CLKSEL: Clock Selection: 0	System Clock / 256	\sim	CMP2: Compare 2 Interrupt:	0			
	CNT: Count: 0	0x0	hex v	OVF: Overflow Interrupt:	0 🗸			
	PER: Period:	0x9896	hex v	EVENT CONFIGURATION				

これら全ての周辺機能の構成設定後、プロジェクトは生成の準備が整い、使用者はウェブに基づくAtmel START版については'EXP ORT PROJECT(プロジェクトをエクスポート)'釦、その後に'DOWNLOAD PACK(一括をダウンロート')'を押す、またはAtmel Studio版で 頁の下部から単に'GENERATE PROJECT(プロジェクトを生成)'を押すことができます。コート'生成部はプロジェクトの構造とファイルを作 成し、使用者に残された要件は主応用を提供するだけです。

3.4. Lambda概要

LambdaはGreengrass中継器が終端装置とクラウドと相互通信することができる方法の1つです。Lambdaはクラウドで構成設定や編集をされ、Raspberry Pi基板で展開されます。この機能はクラウド接続と話題指定の購読と掲載に使われる応用プログラミング インターフェース(API: Application Programming Interface)を提供するソフトウェア開発キット(SDK:Software Development Kit)と共に来ます。Raspberry Piとクラウド間のメッセージはMQTT規約を通して送られますが、Greengrass核はこの手順を処理し、従って、これらの詳細は使用者に意識されません。

Lambdaはいくつかのプログラミング言語とプログラミング言語の版で書くことができます。Greengrassに対してAWSによって提供された単位 部3の指導書がPython™ 2.7に基づき、従って、使用者に対してLambdaの構造と手続きを理解して習熟させることが容易なため、この 応用で使われたのものはPython 2.7で書かれています。

Lambdaの流れ図が次に記述されます。

- 1. Lambdaはクラウド通信に対して責任を持つGreengrassクライアントを作成します。
- 2. BLEの接続と通信に対して責任を持つツールを呼び出し、それの仮想使用者イン ターフェースに入ります。他のRaspbianで利用可能な他のBLE向きツールを使うことが できます。
- 3. MACアトレスを使ってBLE終端装置に接続します。ツールは装置操作を提供せず、 MACアトレスが装置毎に異なるので、使用者はMACアトレスを提供する責任があり ます。MACアトレスを得る方法のより多くの情報は4.2.2. 命令状態で見つけること ができます。
- 4. 一旦接続が確立されてしまうと、処理部番号が後続する'char-read-hnd'命令を 使って温度と光の特性で利用可能なバイを読み、データを包んで話題の'BLE/d ata'と共にクラウト'へ掲載します。この段階は繰り返しで、この活動は5秒毎に1度 実行されます。話題名と実行間の時間は使用者選択で、変更することができま す。特性の読み込みは特性が作成された時に割り当てられた処理部を通して行 われます。処理部を得る方法のより多くの情報は4.2.2. 命令状態で見つけること ができます。





4. 応用実演

4.1. 主応用構成設定

Atmel STARTコート、生成部はプロジェクトの構造とシステムの初期化と共に周辺機能ファイルとAPIを作成します。使用者は全てのものが普通 に開始するのを保証するためにBLE単位部のリセット手順を追加し、感知器の値を読む関数を書き、BLE単位部がPCから制御できる ようにUSART周辺機能の信号線を相互接続しなければなりません。使用者はたまに感知器を読んで特性を更新しなければなりません。これらの段階の全てが既に書かれていてGitHub貯蔵庫プロジェクトで提供されます。

4.2. 基板の準備

使用者はAVR-IoT WG開発基板のmikroBUSにRN4871 Click基板を挿入し、その後にそれをマイクロUSBを通してノートPCに接続しな ければなりません。GitHub貯蔵庫からダウンロートした主プロジェクトは'AVR-IoT_WG_with_BLE'フォルダで見つけることができ、'AVR_BL E_Project.atsln'ファイルをダブル クリックすることよって開くことができます。使用者はAtmel Studioを使って、またはAVR-IoT WG開発基 板と連携する大容量記憶装置にhexファイルをトラッグ&トロップすることによって直接基板に書くことができます。

プロジェクトは、応用が5秒毎に感知器を読んで特性を更新する応用状態と、RN4871基板と相互通信するのに使うことができる命令状態の2つの状態に分けられます。

4.2.1. 応用状態

応用は応用状態で開始し、初期化とRN4871単位部の構成設定検査によって始まり、これらが未だ存在していなければ光と温度の感知器用のサービスと特性を作成します。5秒毎に応用は感知器を読んで感知器の値で特性を更新します。

4.2.2. 命令状態

命令状態は使用者に直接命令を送ることとRN 4871単位部の機能を調べることを許します。命 令状態はPCで走行している(Hercules やTera Termのような)シリアルソフトウェア端末で'/'文字を 送ることによって何時でも入ることができます。 この文字を使うと、応用は使用者に新しい状態 を返答します。

COMポートを開く前に、使用者はボーレートがAtm el STARTで選んだ2つ目のUSART実体と同じ (115200bps)であることを確実にしなければなり ません。この応用記述でCOMポートはHercules 端末を使って開かれます。AVR-IoT WG開発 基板のデバッガが流れ制御を必要とするため、 使用者はDTRとRTSが活性にされることを確実 にしなければなりません。

Received/Sent data	Ceria	I ————
Serial port COM11 opened	Name COM Baud	11 🔽
	Data	size
	Parity none	
	Hand: OFF	shake 👻
	Mode Data	Ţ
. Madaa 6		🗙 Close
		Wg FW update

命令状態で使用者は以下の命令を使わなければなりません。

- 1. 装置のMACアドレスを調べてください。この情報は後でLambdaによって必要とされます。MACアドレスは返答の最初の行です。
 - •送出:'d'
 - ・返答:'BTA=D88039F37559'
 - ・返答: 'Name=RN4870-7559'
 - 返答: 'Connected=no'
 - •返答: 'Authen=2'
 - •返答: 'Features=0000'
 - ・返答: 'Services=CO'
- 2. 基板が利用可能な最新のファームウェアを使うことを確実にしてください。この時点では1.30版です。
 - •送出:'v'
- ・返答: 'RN4871 V1.30 3/18/2018 (c)Microchip Technology Inc'

命令状態で使用者はBLE RN4871単位部使用者の手引きで記述されるどの命令も利用することができます。

注: この状態では応用がもはや感知器を読んで特性を更新せず、故に使用者は応用状態に切り替え戻すのに再び'/'を送らなけれ ばなりません。

4.3. Lambdaの作成と読み込み

Raspberry PiにGreengrass核をインストールして準備するため、使用者はAWS IoT Greengrassでの開始に際して開発の手引きに従わな ければなりません。この開発者の手引きは最低単位部3まで確認されなければなりません。

Greengrass手続き完了後、使用者はこれらの段階に従ってLambdaの構成設定を続けることができます。

- 1. 使用者はGreengrass核が未だ活性でないことを確実にしなければなりません。核を停止するため、使用者はRPiで新しい端末ウィンドウを開き、核の場所へ行って停止命令を実行しなければなりません。
 - ・新しい端末ウィンドウを開くためにCtrl+Alt+Tを押してください。
 - ・核の場所を開くために'cd /greengrass/ggc/core'を入力してください。
 - ・核を停止するために'sudo ./greengrassd stop'を入力してください。
- 2. Lambdaとその依存物はGitHub貯蔵庫で提供されます。使用者は'AWS_Lambda'フォルダを開いてテキストエディタで'lambda_functio n.py'ファイルを開かなければなりません。
- 3. 'DEVICE'変数がRN4871単位部のMACアドレスを表し、4.2.2. 命令状態で得られるもので変更されなければなりません。MACアドレスは6バイトを持ち、使用者は各バイト間にコロン':'を追加しなければなりません。
 - DEVICE = "D8:80:39:F3:75:59"
- 4. 使用者はそれ自身のフォレダではなく、'AWS_Lambda'フォレダの内容全体を.zipとして書庫化しなければなりません。その後、AWS Greengrass指導書単位部3で記述されるように、使用者はクラウドにその書庫をアップロードしてそれをRaspberry Piで展開しなけれ ばなりません。購読作成時に使われる話題は'BLE/data'です。
- 5. 既定により、RN4871は認証と暗号を持つ接続を必要とし、RPiのBLE単位部は既定でこれを実装しません。RPiのBLEに対して装置が責任を持つ名前は'hci0'で、'hciconfig'命令の使用はこの単位部の既定状態を示します。

```
pi@raspberrypi:~ $ sudo hciconfig hci0
hci0: Type: Primary Bus: UART
BD Address: B8:27:EB:09:B1:6A ACL MTU: 1021:8 SC0 MTU: 64:1
UP RUNNING
RX bytes:813 acl:0 sco:0 events:53 errors:0
TX bytes:2524 acl:0 sco:0 commands:53 errors:0
```

暗号での認証は'encrypt'引数を使って活性化されなければなりません。単位部状態調査に上と同じ命令を使い、新しい'AUT H'と'ENCRYPT'の機能を見ることができます。

```
pi@raspberrypi:~ $ sudo hciconfig hci0 encrypt
pi@raspberrypi:~ $ sudo hciconfig hci0
hci0: Type: Primary Bus: UART
BD Address: B8:27:EB:09:B1:6A ACL MTU: 1021:8 SCO MTU: 64:1
UP RUNNING AUTH ENCRYPT
RX bytes:827 acl:0 sco:0 events:55 errors:0
TX bytes:2534 acl:0 sco:0 commands:55 errors:0
```

6. 全ての構成設定が実行され、今やGreengrass核を開始することができます。

- ・新しい端末ウィンドウを開くためにCtrl+Alt+Tを押してください。
- ・核の場所を開くために'cd /greengrass/ggc/core'を入力してください。
- ・核を開始するために'sudo ./greengrassd start'を入力してください。
- 注:上の構成設定の全ては新しいLambdaの読み込み設定時、初回にだけ従わなければなりません。RPi基板でリセット実行後は段階5 と段階6だけ従わなければなりません。

4.4. クラウドでの感知器データ可視化

使用者は私的なAWSアカウントを開き、AWS Greengrass核開発者の手引きの単位部3で提示されるように、購読を試験しなければなりません。Lambdaが掲載したデータの話題は'BLE/data'で使用者はこれを購読しなければなりません。

図4-2. クラウト 試験	
Subscriptions	
Subscribe to a topic Publish to a topic	Subscribe Devices publish MQTT messages on topics. You can use this client to subscribe to a topic and receive these messages. Subscribton topic BLE/data Subscribte to topic Max message capture ? Quality of Service ? Quality of Service ? Quality of Service ? Quality of Service ? Auto-format JSON payloads (improves readability) Auto-format JSON payloads (improves readability) Display payloads (in hexadecimal)
	Publish Specify a topic and a message to publish with a QoS of 0. Specify a topic to publish to, e.g. myTopic/1 1 (*nessage*: "Hello from AHS IOT console"

購読後、ウインドウは下の画像でのように5秒毎に感知器の値を表示します。

<u> </u>							
Subscriptions	BLE/data		Export Clear	Pause			
Subscribe to a topic Publish to a topic BLE/data X	Publish Specify a topic and a message to BLE/data	n ANS 107 console*	Publish to to	opic			
	BLE/dətə	May 8, 2019 2:04:50 PM +0300	Export	t Hide	Â		
	BLE/data	May 8, 2019 2:04:50 PM +0300	Export	t Hide			
	BLE/data Temperature: 26	May 8, 2019 2:04:45 PM +0300	Export	t Hide			
	BLE/data	May 8, 2019 2:04:45 PM +0300	Export	t Hide			

5. 結び

この応用記述で提供された段階に従い、使用者はクラウドに対して中継器を通してBLEデータを送る簡単な解決策を得ることができます。MicrochipのAVR-IoT WG開発基板は使用者にRM4871 Click基板に対してそれらのプロジェクトに新しい機能を追加することを許すmikroBUS取り付け口を提供します。Atmel STARTコート、生成部はこの基板に対する支援、基板のインターフェース構成設定と既定設定を提供します。中継器ソフトウェアは構成設定と機能に関する詳細を提供する開発者の手引きと共にAWSによって提供されます。AVR-I oT WG開発基板用プロジェクトと中継器用LambdaはGitHub貯蔵庫で提供されます。

<mark>6</mark>. 参照

- 1. AVR-IoT WG開発基板
- 2. AVR-IoT WG開発基板回路図
- 3. BLE RN4871 Click基板
- 4. BLE RN4871単位部使用者の手引き
- 5. Raspberry Pi 3 B+型
- 6. AWS開発者の手引き
- 7. GitHub貯蔵庫

Microchipウェフ゛サイト

Microchipはhttp://www.microchip.com/で当社のウェブ サイト経由でのオンライン支援を提供します。このウェブ サイトはお客様がファイルや情報を容易に利用可能にする手段として使われます。お気に入りのインターネット ブラウザ を用いてアクセスすることができ、ウェブ サイトは以下の情報を含みます。

- ・製品支援 データシートと障害情報、応用記述と試供プログラム、設計資源、使用者の手引きとハートウェア支援資料、最新ソフトウェア配布と 保管されたソフトウェア
- ・全般的な技術支援 良くある質問(FAQ)、技術支援要求、オンライン検討グループ、Microchip相談役プログラム員一覧
- ・Microshipの事業 製品選択器と注文の手引き、最新Microchip報道発表、セミナーとイベントの一覧、Microchip営業所の一覧、代理 店と代表する工場

お客様への変更通知サービス

Microchipのお客様通知サービスはMicrochip製品を最新に保つのに役立ちます。加入者は指定した製品系統や興味のある開発ツール に関連する変更、更新、改訂、障害情報がある場合に必ず電子メール通知を受け取ります。

登録するにはhttp://www.microchip.com/でMicrochipのウェブ サイトをアクセスしてください。"Support"下で"Customer Change Notificati on"をクリックして登録指示に従ってください。

お客様支援

Microchip製品の使用者は以下のいくつかのチャネルを通して支援を受け取ることができます。

- ・代理店または販売会社
- ・最寄りの営業所
- ・現場応用技術者(FAE:Field Aplication Engineer)
- ・技術支援

お客様は支援に関してこれらの代理店、販売会社、または現場応用技術者(FAE)に連絡を取るべきです。最寄りの営業所もお客様の手助けに利用できます。営業所と位置の一覧はこの資料の後ろに含まれます。

技術支援はhttp://www.microchip.com/supportでのウェブ サイトを通して利用できます。

Microchipデバイスコート、保護機能

Microchipデバイスでの以下のコード保護機能の詳細に注意してください。

- ・Microchip製品はそれら特定のMicrochipデータシートに含まれる仕様に合致します。
- ・Microchipは意図した方法と通常条件下で使用される時に、その製品系統が今日の市場でその種類の最も安全な系統の1つであると考えます。
- コード保護機能を破るのに使用される不正でおそらく違法な方法があります。当社の知る限りこれらの方法の全てはMicrochipのデー タシートに含まれた動作仕様外の方法でMicrochip製品を使用することが必要です。おそらく、それを行う人は知的財産の窃盗に関 与しています。
- ・Microchipはそれらのコードの完全性について心配されているお客様と共に働きたいと思います。
- ・Microchipや他のどの半導体製造業者もそれらのコードの安全を保証することはできません。コード保護は当社が製品を"破ることができない"として保証すると言うことを意味しません。

コート、保護は常に進化しています。Microchipは当社製品のコート、保護機能を継続的に改善することを約束します。Microchipのコート、保護機能を破る試みはデジタルシニアム著作権法に違反するかもしれません。そのような行為があなたのソフトウェアや他の著作物に不正なアクセスを許す場合、その法律下の救済のために訴権を持つかもしれません。

法的通知

デバイス応用などに関してこの刊行物に含まれる情報は皆さまの便宜のためにだけ提供され、更新によって取り換えられるかもしれま せん。皆さまの応用が皆さまの仕様に合致するのを保証するのは皆さまの責任です。Microchipはその条件、品質、性能、商品性、 目的適合性を含め、明示的にも黙示的にもその情報に関連して書面または表記された書面または黙示の如何なる表明や保証もし ません。Microchipはこの情報とそれの使用から生じる全責任を否認します。生命維持や安全応用でのMicrochipデバイスの使用は完 全に購入者の危険性で、購入者はそのような使用に起因する全ての損害、請求、訴訟、費用からMicrochipを擁護し、補償し、免責 にすることに同意します。他に言及されない限り、Microchipのどの知的財産権下でも暗黙的または違う方法で許認可は譲渡されま せん。

商標

Microchipの名前とロゴ、Mcicrochipロゴ、AnyRate、AVR、AVRロゴ、AVR Freaks、BitCloud、chipKIT、chipKITロゴ、CryptoMemory、Cr yptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KeeLoq、KeeLoqロゴ、Kleer、LANCheck、LINK MD、maXStylus、maXTouc h、MediaLB、megaAVR、MOST、MOSTロゴ、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32ロゴ、Prochip Designer、QTou ch、SAM-BA、SpyNIC、SST、SSTロゴ、SuperFlash、tinyAVR、UNI/O、XMEGAは米国と他の国に於けるMicrochip Technology Incor poratedの登録商標です。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTou ch、Precision Edge、Quiet-Wireは米国に於けるMicrochip Technology Incorporatedの登録商標です。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、CodeGuard、CryptoAut hentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherG REEN、In-Circuit Serial Programming、ICSP、INICnet、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet^{DI}、memBrain、 Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified^{DI}、MPLAB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、SAM-ICE、 Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、View Sense、WiperLock、Wireless DNA、ZENAは米国と他の国に於けるMicrochip Technology Incorporatedの商標です。

SQTPは米国に於けるMicrochip Technology Incorporatedの役務標章です。

Silicon Storage Technologyは他の国に於けるMicrochip Technology Inc.の登録商標です。

GestICは他の国に於けるMicrochip Technology Inc.の子会社であるMicrochip Technology Germany II GmbH & Co. KGの登録商 標です。

ここで言及した以外の全ての商標はそれら各々の会社の所有物です。

© 2019年、Microchip Technology Incorporated、米国印刷、不許複製

DNVによって認証された品質管理システム

ISO/TS 16949

Microchipはその世界的な本社、アリゾナ州のチャント・ラーとテンヘ。、オレゴン州グラシャムの設計とウェハー製造設備とカリフォルニアとイント・の設計センターに対してISO/TS-16949:2009認証を取得しました。当社の品質システムの処理と手続きはPIC® MCUとdsPIC® DSC、KEELOQ符号飛び回りデバイス、直列EEPROM、マイクロ周辺機能、不揮発性メモリ、アナログ製品用です。加えて、開発システムの設計と製造のためのMic rochipの品質システムはISO 9001:2000認証取得です。

日本語© HERO 2019.

本応用記述はMicrochipのAN3083応用記述(DS00003083A-2019年5月)の翻訳日本語版です。日本語では不自然となる重複する 形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意訳されている部分もあります。必要に応じて一部 加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。



米国

世界的な販売とサービス

本社

2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 技術支援: http://www.microchip.com/ support ウェブ アトレス: http://www.microchip.com アトランタ Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 オースチン 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 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 インデアナポリス Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 ロサンセルス Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 D-J- NC Tel: 919-844-7510 ニュ**ーヨーク** NY Tel: 631-435-6000 サンホセ CA Tel: 408-735-9110 Tel: 408-436-4270 カナダ - トロント Tel: 905-695-1980

オーストラリア - シト・ニー Tel: 61-2-9868-6733 中国 - 北京 Tel: 86-10-8569-7000 中国 - 成都 Tel: 86-28-8665-5511 中国 - 重慶 Tel: 86-23-8980-9588 中国 - 東莞 Tel: 86-769-8702-9880 中国 – 広州 Tel: 86-20-8755-8029 中国 – 杭州 Tel: 86-571-8792-8115 中国 - 香港特別行政区 Tel: 852–2943–5100 中国 - 南京 Tel: 86-25-8473-2460 中国 - 青島 Tel: 86-532-8502-7355 中国 - 上海 Tel: 86-21-3326-8000 中国 - 瀋陽 Tel: 86-24-2334-2829 中国 - 深圳 Tel: 86-755-8864-2200 中国 – 蘇州 Tel: 86-186-6233-1526 中国 - 武漢 Tel: 86-27-5980-5300 中国 - 西安 Tel: 86-29-8833-7252 中国 - 廈門 Tel: 86-592-2388138 中国 - 珠海 Tel: 86-756-3210040

亜細亜/太平洋

亜細亜/太平洋

イント - ハンガロール

Tel: 91-80-3090-4444 イント・ニューデリー Tel: 91-11-4160-8631 イント・フネー Tel: 91-20-4121-0141 日本 - 大阪 Tel: 81-6-6152-7160 日本 - 東京 Tel: 81-3-6880-3770 韓国 - 大邱 Tel: 82-53-744-4301 韓国 - ソウル Tel: 82-2-554-7200 マレーシア – クアラルンプール Tel: 60-3-7651-7906 マレーシア ー ヘ・ナン Tel: 60-4-227-8870 フィリピン ー マニラ Tel: 63-2-634-9065 シンガポール Tel: 65-6334-8870 台湾 - 新竹 Tel: 886-3-577-8366 台湾 - 高雄 Tel: 886-7-213-7830 台湾 - 台北 Tel: 886-2-2508-8600 タイ ー バンコク Tel: 66-2-694-1351 ベトナム ー ホーチミン Tel: 84-28-5448-2100

欧州 オーストリア – ウェルス Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 テンマーク - コヘンハーケン Tel: 45-4450-2828 Fax: 45-4485-2829 フィンラント – エスホー Tel: 358-9-4520-820 フランス – パリ Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 トイツ – ガルヒング Tel: 49-8931-9700 ドイツ – ハーン Tel: 49-2129-3766400 トイツ – ハイルブロン Tel: 49-7131-72400 ドイツ – カールスルーエ Tel: 49-721-625370 ドイツ - ミュンヘン Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 ドイツ - ローセンハイム Tel: 49-8031-354-560 イスラエル - ラーナナ Tel: 972-9-744-7705 イタリア ー ミラノ Tel: 39-0331-742611 Fax: 39-0331-466781 イタリア ー パドバ Tel: 39-049-7625286 オランダーデルーネン Tel: 31-416-690399 Fax: 31-416-690340 ノルウェー - トロンハイム Tel: 47-72884388 ホ[°]ーラント[゛]ー ワルシャワ Tel: 48-22-3325737 ルーマニア – ブカレスト Tel: 40-21-407-87-50 スペイン - マドリート Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 スウェーデン – イェーテホリ Tel: 46-31-704-60-40 スウェーデン – ストックホルム Tel: 46-8-5090-4654 イキ・リス - ウォーキンガム Tel: 44-118-921-5800 Fax: 44-118-921-5820

Fax: 905-695-2078