

# RA2E3 グループ

RA2E3 HS4001 低消費電力センサシステム実装例

要旨 (Introduction)

本アプリケーションノートでは、RA ファミリデバイスで動作する HS400x 温湿度センサで取得したデー タを OLED モジュール SSD1306 に表示するアプリケーションについて説明します。

本アプリケーションは低消費電力モードのソフトウェアスタンバイモードを使用します。ソフトウェアス タンバイモードは通常モードと比べ、CPU やほとんどの周辺機能を停止させることで消費電力を最小限に 抑えることができます。MCU は 500 ミリ秒ごとのウェイクアップ割り込み処理と4 秒ごとの OLED センサ データ更新処理以外はソフトウェアスタンバイモードで長時間動作します。ウェイクアップイベントの発生 には AGT タイマを使用します。

### 対象デバイス (Target Device)

RA2E3

本アプリケーションを他のマイコンに適用する場合は、適用先のマイコンの仕様に合わせて修正し、十分 な評価を行ってください。

必須リソース (Required Resources)

本アプリケーションに必要なリソースは下記の通りです。

ハードウェア

- FPB-RA2E3 Fast Prototyping Board (RTK7FPA2E3S00001BE)
- ・ 相対湿度センサ Pmod<sup>™</sup> ボード QCIOT-HS4001POCZ (QCIOT-HS4001POCZ)
- OLED モジュール
  - SSD1306 コントローラ
  - 解像度: 128 x 64 ドットマトリクスパネル
  - 電源電圧: 3.3 V
  - インタフェース: I2C
  - スレーブアドレス: 0x3c (7-bit address)
- ジャンパワイヤ(オスメスタイプ)4本

※SEGGER J-Link<sup>™</sup>オンボードを使用するため、別途エミュレータは不要

#### 開発ツール・ソフトウェア

- e<sup>2</sup> studio IDE version 2025-01
- Renesas Flexible Software Package (FSP) version 5.8.0
- GCC ARM Embedded Toolchain version 13.2.1.arm-13-7



# 目次

1	はじめに	3
1. 1.1		o
1.1	えんちょう えんしょう しょうしょう しょう	. ७ २
1.2		. U
1.0	1上 1 本	. –
2	RA2E3 MCU	5
21	ブロック図	
2.1	> - > > Elimination	6
2.2		. 0
3.	システム概要	6
3.1	導入	6
3.2	使用する周辺機能	6
3.3	使用する端子	7
3.4	動作手順	7
4.	ハードウェア	7
5.	ソフトウェア	9
5.1	統合開発環境	9
5.2	動作概要	9
5.3	フローチャート	10
5.3.	1 メイン処理	10
5.3.2	2 タイマ割り込み処理	11
5.4	ファイル構成	12
6.	e <sup>2</sup> studio の FSP を使用してミドルウェアとドライバを追加する方法	13
6.1	HS400X 温湿度センサ (HS400X Temperature/Humidity Sensor)	13
6.2	低消費電力モード (Low Power Modes)	15
6.3	I2C 通信デバイス (I2C Communication Device)	17
7.	プロジェクトのインポートとビルド方法	18
8.	低消費電力プロジェクトのデバッグ方法	18
_		
9.	e <sup>2</sup> studio でテハックする際の Tips	19
9.1	IO Registers 表示	19
9.2	[IO Registers] ビューのカスタマイズ	19
9.3	ブレークポイントの設定	20
10	サンプルプロジェクト	21
10.		- 1
11	参考ドキュメント	21
		- '
改訂	「記録	22

### 1. はじめに

#### 1.1 概要

本アプリケーションは、RA2E3 Fast Prototyping Board (FPB-RA2E3) と相対湿度センサ Pmod<sup>™</sup>ボード QCIOT-HS4001POCZ を使用して、温湿度データを取得し、OLED モジュール SSD1306 の画面に室内の湿 度と温度を表示するプログラムです。低消費電力モードのソフトウェアスタンバイモードを使用しており、 CPU の動作時間を最小限に抑え、消費電力を削減します。

RA2E3 Fast Prototyping Board は、R7FA2E3073CFL マイコンを搭載し、様々なアプリケーションの試作 開発に特化した評価ボードです。SEGGER J-Link<sup>™</sup>オンボードを内蔵し、追加のツール無しでプログラムの 書き込み/デバッグが可能です。さらに Arduino Uno<sup>™</sup>および Pmod<sup>™</sup>インタフェースを標準搭載、マイコンの 全端子にアクセス可能なスルーホールなど、高い拡張性を有しています。

HS4001 は高精度、超低消費電力、完全校正の相対湿度・温度センサです。この MEMS センサは独自の センサーレベル保護機能を備えており、高い信頼性と長期安定性を確保しています。HS4001 は I2C デジタ ル出力を持ち、完全に校正・温度補正されます。

SSD 1306 は 128x64 の有機 EL(OLED)を使用した I2C 制御のグラフィックディスプレイモジュールコン トローラです。SSD1306 は Arduino や Raspberry Pi などでライブラリが多数用意されている定番の IC で す。

電源供給	USB 電源 (5 V)
動作電圧 (MCU)	3.3 V
OLED 表示パターン	14 文字×4列 (128×64 ドット)

1.2 主な技術パラメータ



### 1.3 仕様

- QCIOT-HS4001POCZ で室内の湿度と温度を検出します。
- AGT を利用して、500 ミリ秒ごとに割り込み処理を行い、4 秒ごとに OLED 画面上の温度・湿度 データの表示を更新します。
- ソフトウェアスタンバイモードは、消費電力を削減するための低消費電力モードとして使用されます。



図 1.1 に MCU の状態とモード遷移イベントを示し、図 1.2 に動作モードと消費電流の概念図を示します。

図 1.1 MCU の状態とモード遷移イベント



図 1.2 動作モードと消費電流の概念図



#### 2. RA2E3 MCU

RA2E3 グループは 48MHz 動作の Arm<sup>®</sup> Cortex<sup>®</sup>-M23 コアをベースとし、最大 64KB のコードフラッシュ と 16KB の SRAM メモリを搭載した RA ファミリのエントリーライン シングルチップ MCU です。

RA2E3 グループは内蔵周辺機能を最適化することで、コスト重視のアプリケーションに最適です。 RA2E3 グループの特長の1つである超低消費電力は、loT アプリケーションやバッテリ駆動システムで求め られるエネルギー効率の高いシステム設計に貢献し、バッテリの長寿命化を実現します。

RA2E3の詳細は下記リンクをご参照ください。

http://renesas.com/ra2e3

仕様の詳細は下記リンクをご参照ください。

https://www.renesas.com/document/mah/ra2e3-group-users-manual-hardware

RA2E3 ユーザーズマニュアル ハードウェア編 (R01UH0992)

### 2.1 ブロック図

RA2E3 のブロック図を図 2.1 に示します。

RA2E3 48	MHz 32-Bit Arm <sup>®</sup> Co	ortex <sup>®</sup> -M23 Core	NVIC   SWD   MTB
Memory	<b>₩</b> Analogue	<b>Ö</b> Timers	<u> В ниі</u>
Code Flash (64KB, 32KB)	12-bit ADC (13ch)	GPT 32-bit (1ch)	
SRAM (16KB) Parity	remperature Sensor	AGT 16-bit (8ch)	
Data Flash (2KB)		WDT	
	System	Safety	Security
I2C x 1	Sys Tick	Memory Protection Unit	128 bit Unique ID
SCI x 4	DTC	SRAM Parity Check	
SPI x 1	On-Chip Oscillator HOCO (24,32,48,64MHz),	Clock Frequency Accuracy Measurement	
	ILOCO (32KHZ), ILOCO (15KHZ)	CRC Calculator	Package
	Low Power Modes	IWDT	
	ELC	Data Operation Circuit	LQFP 32, 48
	Port Function Select	Flash Area Protection	QFN 32, 48
	RTC	ADC Self Test	

図 2.1 RA2E3 のブロック図



RA2E3 グループ

### 2.2 RA2E3 低消費電力モード

RA2E3 ユーザーズマニュアル(R01UH0992)の表「各低消費電力モードの動作状態」では、低消費電力 モードへの移行条件、CPU および周辺モジュールの状態、および各モードからの解除条件について説明し ています。

使用可能な低消費電力モードは次のとおりです。

- スリープモード
- ソフトウェアスタンバイモード
- スヌーズモード

ソフトウェアスタンバイモードは、SRAMの内容を保持したまま CPU とほとんどの周辺機能を停止させることで、消費電力を最小限に抑えます。MCU は、外部割り込み、RTC アラーム、または AGT アンダーフローイベントによってこのモードからウェイクアップできます。以下の周辺機能はソフトウェアスタンバイモードでも動作可能です。

- AGT (低消費電力非同期汎用タイマ): カウントを継続し、ウェイクアップをトリガできます
- RTC: アラームベースのウェイクアップを生成できます
- IRQ 端子: 外部信号の受信時に MCU をウェイクアップできます
- IIC およびその他の通信周辺: ウェイクアップイベントが発生するまで無効のままです

### 3. システム概要

#### 3.1 導入

本アプリケーションでは、RA2E3 MCU、デジタル温湿度センサ、OLED ディスプレイモジュールを使用 します。MCU(RA2E3) が室内の温湿度を検出した後、ユーザは OLED 画面でデータを確認できます。500 ミリ秒ごとに AGT 割り込みを発生させ、4 秒ごとに温湿度のデータを取得し、OLED の表示を更新しま す。

なお、CPU が動作しない時間は低消費電力モードのソフトウェアスタンバイモードで動作します。

図 3.1 にシステム構成を示します。



図 3.1 システム構成

### 3.2 使用する周辺機能

表 3.1 に使用する周辺機能と用途を示します。

表 3.1 使用する周辺機能

Peripheral Function	Usage
I2C (IIC0)	センサから温湿度データの取得
	OLEDを制御して温湿度データの表示
AGT1	500 ミリ秒ごとにクロックサイクルをカウント



3.3 使用する端子

表 3.2 に使用する端子を示します。

表 3.2 使用する端子

Pin Name	Description
P400/SCL0	クロック信号: I2C バスを介してセンサ(HS4001)および OLED と通信
P401/SDA0	データ信号: I2C バスを介してセンサ(HS4001)および OLED と通信
VCC	電源
GND	グランド

#### 3.4 動作手順

- 1. 電源が供給されると、システムの初期化を開始します。
- 2. 初期化後、OLED(SSD1306)に "RENESAS" およびデモタイトルを表示します。
- 3. AGT はカウントを開始した後、500 ミリ秒ごとに割り込みを発生させ、ソフトウェアスタンバイモー ドを解除します。
- タイマ割り込み処理を実行後、MCU(RA2E3)はソフトウェアスタンバイモードで動作し、AGT からの 次の割り込みを待ちます。ソフトウェアスタンバイモードを8回解除(4 秒経過)するまで、ソフトウェ アスタンバイモードと通常モードの遷移を繰り返します。
- 5. MCU(RA2E3)は4秒ごとに温湿度データを取得して OLED(SSD1306)に送信し、OLED 画面の表示を 更新します。

表示パターン

R	Е	Ν	E	S	А	S							
F	Ρ	В	-	R	А	2	E	3		D	Е	Μ	0
Т	е	m	р		х	х		х	С				
Н	u	m	i		х	х		х	%				



4. ハードウェア

ここでは、アプリケーションで使用されるハードウェア製品と接続構成について説明します。

QCIOT-HS4001POCZ の詳細については、下記リンクをご参照ください。

https://www.renesas.com/en/products/sensor-products/environmental-sensors/humidity-temperature-sensors/qciot-hs4001pocz-relative-humidity-sensor-pmod-board

SSD1306の詳細については、下記リンクをご参照ください。

https://www.solomon-systech.com/product/ssd1306/

図 4.1 に FPB-RA2E3PMOD インタフェースを示します。図 4.2 に FPB-RA2E3、QCIOT-HS4001POCZ と SSD1306 の接続構成を示します。図 4.3 にハードウェア構成を示します。





図 4.1 FPB-RA2E3 PMOD インタフェース



図 4.2 FPB-RA2E3、QCIOT-HS4001POCZ と SSD1306 の接続構成



#### 図 4.3 ハードウェア構成



### 5. ソフトウェア

### 5.1 統合開発環境

このサンプルプロジェクトは、表 5.1 に示す条件で確認されています。

表 5.1 動作確認条件

Item	Description
ボード	FPB-RA2E3
デバイス	RA2E3 (R7FA2E3073CFL)
動作周波数	High-speed on-chip oscillator (HOCO) clock: 48 MHz
	System clock (ICLK): 48 MHz
	Peripheral module clock B (PCLKB): 24 MHz
	Peripheral module clock D (PCLKD): 48 MHz
動作電圧	3.3 V
統合開発環境 (e <sup>2</sup> studio)	e <sup>2</sup> studio 2025-01
FSP	5.8.0 from Renesas Electronics Corp.
ツールチェーン	13.2.1.arm-13-7
(GCC ARM Embedded)	
HS4001 Library	HS400X Temperature/Humidity Sensor (rm_hs400x)
Low Power Modes driver	Low Power Modes (r_lpm)
Timer driver	Timer, Low-Power (r_agt)

#### 5.2 動作概要

(1) リセット/初期化

電源を投入すると初期化処理を実行します。OLED に電源が供給され、表示がクリアされます。その後、 デフォルトで "Renesas Electronics" などの文字が表示されます。HS4001 が初期化されます。IIC0 と I/O ポート端子も初期化されます。



図 5.1 OLED 画面の初期化

(2) 測定モード

初期化後、MCU はセンサ測定データの取得を開始します。

(3) 表示モード

測定後、MCU は表示するための情報を OLED に送信します。



図 5.2 温湿度データの表示画面



5.3 フローチャート

5.3.1 メイン処理

図 5.3 にメイン処理のフローチャートを示します。



図 5.3 メイン処理



### 5.3.2 タイマ割り込み処理

図 5.4 にタイマ割り込み処理のフローチャートを示します。



図 5.4 タイマ割り込み処理



### 5.4 ファイル構成

ファイル構成は下記の通りです。



図 5.5 ファイル構成



6. e<sup>2</sup> studio の FSP を使用してミドルウェアとドライバを追加する方法

ここでは、コンフィグレータにミドルウェアと HAL ドライバを FSP の Stacks に追加する方法を紹介し ます。紹介するスタック構成はアプリケーションで必要とされるスタック構成です。

なお、本アプリケーションはプロジェクトをインポートすることでも利用可能です。インポートの手順は、「7. プロジェクトのインポートとビルド方法」を参照してください。

6.1 HS400X 温湿度センサ (HS400X Temperature/Humidity Sensor)

- 1. e<sup>2</sup> studio を起動
- 2. 新規プロジェクトを作成

[Board] の [...] から RA2 > RA2E3 > FPB-RA2E3 を選択

Renes	sas RA C/C++ Project s RA C/C++ Project			_	
Device a	and Tools Selection				
Device St FSP Vers Board: Device: Core: Languaç	Selection sion: 5.8.0 FPB-RA2E3 R7FA2E3073CFL CM23 GM23 @ C O C++	> 	Board Description Fast Prototyping Board Visit https://www.rene manual, quick start gu projects, etc. Device Details TrustZone Pins Processor	f for RA2E3 MCU Group sascom/ra/fgb-ra2e3 to get kit ide, errata, design package, exar No 48 Cortex-M23	user's mple
IDE Proje	ect Type		Debugger		
e² studio	io managed build	~	J-Link ARM		~
- Toolchair LLVM Er GNU AF	ins imbedded Toolchain for Arm RM Embedded				
13.2.1.ar	rm-13-7 v <u>Manage 1</u>	Foolchains			
?			< 戻る(B) 次/	<b>\(N) &gt;</b> 終了(F)	キャンセル

図 6.1 新規プロジェクトの作成

3. HS4001 センサミドルウェアを [Stacks] タブに追加

New Stack > Sensor > HS400X Temperature/Humidity Sensor (rm\_hs400x)

Threads 🐑 New Thread 🎪 Remove 📄	HAL/Common Stacks	New Stack	Δ1	Ren	emove
v 🧟 HAL/Common 🔮 g_ioport I/O Port (r_ioport)	g.joport I/O Port (r.joport)       ①		An Analog Audio Bootloader Connectivity DSP Input Monitoring Motor Networking Power	> > > > > > > > > > > >	
Objects 🕢 New Object > 🛍 Remove			Sensor Storage System	> > + + + + + + + + + +	FS1015 Flow Sensor (rm_fs1015)     FS2012 Flow Sensor (rm_fs2012) [Deprecated]     FS3000 Flow Sensor (rm_fs3000)
			Transfer	→ → ⊕	<ul> <li>HS300X Temperature/Humidity Sensor (rm_hs300x)</li> <li>HS400X Temperature/Humidity Sensor (rm_hs400x)</li> </ul>
Summary BSP Clocks Pins Interrupts Event Li	nks Stacks Components		Search	+ + + +	OB1203 Light/Proximity/PPG Sensor (m_ob1203) [Deprecated] RRH46410 Gas Sensor Module (m_rh46410) RRH62000 All-in-one Air Quality Module (m_rrh62000) TXDD Video Concerned Air Quality Module (m_rrh62000)

図 6.2 [Stacks]タブへの追加



4. ターゲットボードの仕様に合わせて r\_iic\_master または r\_sci\_i2c を追加

Threads 🚯 New Thread 🔬 Remove 📄	HAL/Common Stacks	🔊 New Stack > 🏥 Extend Stack > 🔞 Remove
	<ul> <li>g.joport I/O Port (r_joport)</li> <li>g.jhs400x,sensor0 H5400X Temperature/Humidity</li> <li>g.sensor (m_hrs400x)</li> <li>g.comms_j2c_device0 I2C communication Device (rm_comms_j2c)</li> <li>g.comms_j2c_usel</li> <li>g.g.comms_j2c_usel</li> <li>g.g.comms_j2c_usel<td>12C Master (r_jic_master) 12C Master (r_jsci_j2c)</td></li></ul>	12C Master (r_jic_master) 12C Master (r_jsci_j2c)

図 6.3 r\_iic\_master または r\_sci\_i2c の追加

5. I2C master driver のプロパティはターゲットボードの仕様に合わせて設定

[Stacks] タブの[I2C master driver] をクリックすると、[Properties]ウィンドウにプロパティが表示されます。

- anjeens	🐑 New Object > 🛍 Remove	g_i2c_master0 I2C Master (r_iic_master)				
		0				
		Add DTC Driver for Transmission [Optional]				
	< c					
Summary	BSP Clocks Pins Interrupts Event Links Stacks Con	nponents				
Denti						
Propertie	is X problems XY-F-JJJJ-					
g_i2c_m	aster0 I2C Master (r_iic_master)					
	Property	Value				
Settings	× Common					
API Info	Parameter Checking	Default (BSP)				
	DTC on Transmission and Reception	Disabled				
	10-bit slave addressing	Disabled				
	✓ Module g i2c master0 I2C Master (r iic master)					
	Name	g_i2c_master0				
	Channel	<b>Q</b> 0				
	Channel					
	Rate	Standard				
	Rate Custom Rate (bps)	G Standard 0				
	Rate Custom Rate (bps) Rise Time (ns)	G Standard 0 120				
	Cutariner Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns)	Standard     Standard     120     120				
	Charanen Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%)	Standard     O     120     50				
	Chariner Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address					
	Charine Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode					
	Charate Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode	Image: Standard         0           0         120           120         50           0x00         7-Bit           Short Mode         50				
	Charmen Rate Custom Rate (bps) Rise Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode Timeout during SCL Low	Standard         0           120         120           50         50           0x00         7-Bit           Short Mode         Enabled				
	Charlier Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode Timeout during SCL Low Callback	Standard  Standard  C Standard  C Standard  C Standard  Standard  C Standard  Standar				
	Charlier Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode Timeout Mode Callback Interrupt Priority Level					
	Charmen Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode Timeout Mode Timeout during SCL Low Callback Interrupt Priority Level	Control Contr				
	Charmen Rate Custom Rate (bps) Rise Time (ns) Fall Time (ns) Duty Cycle (%) Slave Address Address Mode Timeout Mode Timeout during SCL Low Callback Interrupt Priority Level V Pins SCL0	Standard         0           120         120           120         120           50         0x00           7-Bit         Short Mode           Enabled         @ m_comms_i2c_callback           Priority 2         P400				

図 6.4 I2C Master Driver のプロパティ設定



6. 使用する端子を設定

使用する端子は [Pins]タブで確認可能です。

Type filter text	Name	Value	Lock	Link	
te d' Derinherels	Pin Group Selection	_A only			
Peripherals	Operation Mode	Enabled			
Analog:ADC	✓ Input/Output				
> CLKOUI:CLKOUI	SCL0	💙 P400	- <b>f</b>	4	
Connectivity:IIC	SDA0	✓ P401	i i i i i i i i i i i i i i i i i i i	4	
> Connectivity:SCI					
> Connectivity:SPI					
> V Debug:JTAG/SWD					
> Interrupt:ICU					
> Interrupt:KINT	<				>
> System:CGC	Madula namai JICO				
> 🗸 System:SYSTEM	woodle name. Inco				
> TRG:ADC(Digital)	Usage: For IIC,	use same Pin Group for SDA/SC	L signals		
> TRG:CAC	-Please	refer to the MCO Oser's Manua			
> 🛩 Timers:AGT	~				
Pin Function Pin Number					
Function Pin Number					

図 6.5 使用する端子の確認

- 6.2 低消費電力モード (Low Power Modes)
- 1. LPM のドライバを [Stacks] タブに追加

New Stack > Power > Low Power Modes (r\_lpm)

2. 低消費電力モードの詳細を設定

今回、[Low Power Mode] は [Software Standby mode]を、[Wake Sources] は [AGT1 Underflow] を選択します。



## RA2E3 グループ

# RA2E3 HS4001 低消費電力センサシステム実装例

<	g_hs400x_sensor0 HS400X Tempera g_lpm0 Low Power Modes (r_lpm) >	) Port	<ul> <li>g_hs400x_set HS400X Temperature</li> <li>Sensor (rm_h</li> </ul>	nsor0 /Humidity Is400x)	<ul> <li>g_lpm0 Low Power Modes (r_lpm)</li> </ul>
Objects	. New Object > M. Remove		· ·		
		<			,
Summary	BSP Clocks Pins Interrupts Event Links	Stacks Comp	ponents		
Probler	ms Console Properties X Si	mart Browser	Smart Manual		
g_lpm0	Low Power Modes (r_lpm)				
c	Property			Value	
Settings	✓ Common				
API Info	Parameter Checking			Default (	BSP)
	Standby Limit			Disabled	
	✓ Module g Ipm0 Low Power Modes (r	lpm)			
	✓ General				
	Name			g_lpm0	
	Low Power Mode			Software	Standby mode
	Output port state in standby an	d deep stand	by	Not Supp	orted
	Supply of SOSC clock to periph	eral function i	in standby	Not Supp	orted
	Startup speed of the HOCO in S	tandby and S	nooze modes	Not Supp	orted
	Flash mode in sleep or snooze			Not Supp	orted
	✓ Deep Sleep and Standby Options				
	✓ Wake Sources				
	IRQ0				
	IRQ1				
	IRQ2				
	IRQ3				
	IRQ4				
	IRQ5				
	IRQ6				
	IRQ7				
	IWDT				
	Key Interrupt				
	LVD1 Interrupt				
	LVD2 Interrupt				
	RTC Alarm			U	
	RTC Period				
	AGI1 Underflow				
	AGTI Compare Match A			U	
	AGTT Compare Match B				

図 6.6 Low Power Modes のプロパティ

3. AGT のドライバを [Stacks]タブに追加

New Stack > Timers > Low-Power (r\_agt)

4. AGT の詳細を設定

タイマの周期やクロックソースなどを設定します。ここでは、クロックソースとしてサブクロックを選 択します。

※RA2E3 では、BOM の削減や I/O ポートの節約のため、AGT のクロックソースとして LOCO を選択 することも可能です。その場合は LOCO の精度を確認してください。



<	q_hs400x_sensor0 HS400X ♀ New Object > Remove	ensor0 'e/Humidity _hs400x) <	<ul> <li>g_lpm0 Low Power Modes (r_lpm)</li> <li>Image: Comparison of the second seco</li></ul>	<ul> <li>g_timer1 Timer, Low-Power (r_agt)</li> <li>&gt;</li> </ul>
Summary	3SP Clocks Pins Interrupts Even	t Links Stacks	Components	
Properties	× Problems Smart Bro	owser Cor	nsole Search IO Regi	sters Debug
				📑 🖬 🖓
g_timer1	Timer, Low-Power (r_agt)			
Sottings	Property			Value
Setungs	✓ Common			
API Info	Parameter Checking			Default (BSP)
	Pin Output Support			Enabled
	Pin Input Support			Disabled
	✓ Module g_timer1 Timer, Low-I	Power (r_agt)		
	✓ General			
	Name			g_timer1
	Counter Bit Width			AGT 16-bit
	Channel			1
	Mode			Periodic
	Period			500
	Period Unit			Milliseconds
	Count Source			SUBCLOCK
	> Output			
	> Input			
	> Interrupts			
	✓ PINS			

図 6.7 AGT のプロパティ

6.3 I2C 通信デバイス (I2C Communication Device)

1. OLED のミドルウェアを [Stacks] タブに追加

New Stack > Connectivity > I2C Communication Device (rm\_comms\_i2c)

2. I2C Shared Bus を追加

温湿度センサと OLED はマルチスレーブで接続されるため、使用されている"I2C Shared Bus"を選択 します。

Use > g\_comms\_i2c\_bus0 I2C Shared Bus (rm\_comms\_i2c)



図 6.8 I2C Shared Bus の追加

3. I2C Communication Device のプロパティ設定

今回はプロパティを下記のように設定します。

- Slave Address: 0x3c
- Callback: oled\_comms\_i2c\_callback



# RA2E3 HS4001 低消費電力センサシステム実装例

4 4 4 4 4	New Thread & Remove      g_com     g_hs400x_sensor0 HS400X Temperature     g_jpm0 Low Power Modes (r_jpm)     g_timer1 Timer, Low-Power (r_agt)     g_comms_i2c_device1 I2C Communicat	g_comms_i2c_device112C Communication Device (rm_comms_i2c) Stacks g_comms_i2c_device112C Communication Device (rm_comms_i2c)
<	> E	*
Objects	🐑 New Object > 🔬 Remove	g_comms_i2c_bus0 I2C Shared Bus (rm_comms_i2c)
	(1)	
	\$	g_i2c_master0 I2C Master (r_iic_master)
	0	
	<b></b>	Add DTC Driver for Transmission [Optional]
Summary	BSP   Clocks   Pins   Interrupts   Event Links   Stacks   Cor	nponents
Propertie	s × Problems スマート・プラウザー	
	s i2c device1 I2C Communication Device (rr	n_comms_i2c)
g_comm		
g_comm	Droporty	Value
g_comm Settings	Property	Value
g_comm Settings API Info	Property Common Parameter Checking	Value Default (RSP)
g_comm Settings API Info	Property Common Parameter Checking Module or communication	Value Default (BSP)
g_comm Settings API Info	Property Common Parameter Checking Module g_comms_i2c_device1 I2C Communication Name	Value Default (BSP) n a comms i2c device1
g_comm Settings API Info	Property Common Parameter Checking Module g_comms_i2c_device1 I2C Communication Name Semanhore Timeout (RTOS only)	Value Default (BSP) n g_comms_i2c_device1 0vFFFFFFF
g_comm Settings API Info	Property Common Parameter Checking Module g_comms_i2c_device1 I2C Communicatio Name Semaphore Timeout (RTOS only) Slave Address	Value Default (BSP) n g_comms_i2c_device1 0xFFFFFFF 0x3c
g_comm Settings API Info	Property  Common  Parameter Checking  Module g_comms_j2c_device1 I2C Communicatio Name Semaphore Timeout (RTOS only) Slave Address Address Address	Value Default (BSP) n g_comms_i2c_device1 0xFFFFFF 0x3c 7-Bit
g_comm Settings API Info	Property  Common  Parameter Checking  Module g_comms_j2c_device1 I2C Communicatio Name  Semaphore Timeout (RTOS only)  Slave Address Address Mode Callback	Value           Default (BSP)           n           g_comms_i2c_device1           0xFFFFFFF           0x3c           7-Bit           oled comms_i2c_callback

図 6.9 I2C Communication Device のプロパティ

- 7. プロジェクトのインポートとビルド方法
- 1. e<sup>2</sup> studio を起動
- 2. 表示メニューの [ファイル] をクリックし、[インポート]を選択
- [インポート] ウィンドウが表示されるため、[一般] カテゴリを開き、[既存プロジェクトをワークスペースへ]を選択し、[次へ] をクリック
- [アーカイブ・ファイルの選択] でプロジェクトファイルを選択 選択後、[プロジェクト]に指定したプロジェクト"RA2E3\_HS4001LowPowerSensorSystemExample" が表示されていることを確認し、[終了]をクリック。[インポート] ウィンドウが閉じます。
- 5. [configuration.xml] を開き、[Configurator]ウィンドウの [Generate Project Content] をクリック
- 6. プロジェクト・エクスプローラーでプロジェクト名をクリック
- 7. [プロジェクト] メニューから [プロジェクトのビルド] を選択し、ビルドを開始
- 8. 低消費電力プロジェクトのデバッグ方法
- 1. 表示メニューの[実行] をクリックし、[デバッグの構成]を選択
- 2. [Debugger] タブの [Connection Settings] サブタブで下記を設定
  - J-link
    - Script File: CM\_low\_power\_debug.JLinkScript
    - Low Power Handling: Yes



スクリプトファイルはこのサンプルプロジェクトに含まれており、デバッグ設定で指定することで使用で きます。

※スクリプトファイルは、デフォルトでプロジェクトに適用されていません。スクリプトファイルを設定せ ずにデバッグ接続を行った場合、MCU が正しく LPM に遷移しない可能性があります。また、遷移したとし ても消費電力が想定よりも高くなる可能性があります。



図 8.1 スクリプトファイルの指定

9. e<sup>2</sup> studio でデバッグする際の Tips

#### 9.1 IO Registers 表示

[Window] メニュー > [ビューの表示] > [その他…] を選択し、[ビューの表示]ウィンドウを開き、デバッ グ > IO Registers を選択する。

検索ボックスに参照したいレジスタ名を入力するとレジスタを簡単に見つけることができます。

例: AGT レジスタ

新一規・ワイフト・フ(N)							Q : 图 I 题 C/C++
外観	, oled_ssd1306.c	▲ hal_entry.c ×	🖬 main.c 👘 🗖	プレークポイント	プロジェクト・エクスプローラー	イベントポイント	IO Registers × "3
ビューの表示いり	> CanTaucht A				AGT	~ 🕹 🔂 🖪	8 🖻 🎜 🔕 🎓 🔍 🚍 🛅 🗎 🖻
パースパクティブ(R)	Capitouch )= )			名前	值	アドレス	詳細
1174	Debugger Con	sole		> 🔒 AGTO		0x40084000	Low Power Asynchronous Genera
7E9-999(G)	> Debug source			V 🛃 AGT1		0x40084100	Low Power Asynchronous Genera
設定(P)	Motor Middley	vare Configurator (C	(E)	atter AGT	0x2f59	0x40084100	AGT Counter Register
	Motor7-970	- (QE)		ITE AGTCMA	0x2000	0x40084102	AGT Compare Match A Register
	語 アウトライン		Alt+シフト+Q,0	AGTCMB	0x2000	0x40084104	AGT Compare Match B Register
	🤨 エラー・ログ		Alt+シフト+Q,L	> In AGTCR	0x03	0x40084108	AGT Control Register
	📮 コンソール		Alt+シフト+Q,C	> liter AGTMR1	0x61	0x40084109	AGT Mode Register 1
	5φ シグナル			> and AGTMR2	0x00	0x4008410a	AGT Mode Register 2
	🔿 スマート・ブラウザ	-		> and AGTIOC	0x05	0x4008410c	AGT I/O Control Register
		L		> IIII AGTISR	0x00	0x4008410d	AGT Event Pin Select Register
	なテバッグ			> and AGTCMSR	0x00	0x4008410e	AGT Compare Match Function Se
	N TIL-b			AGTIOSEL	0x00	0x4008410f	AGI Pin Select Register
			44-2-74-00	> BUS		0x40003000	BUS Control
	· JU-9/11/F		AIT+97F+QB	> To CAC		0x40044600	Clock Frequency Accuracy Measu
	109171-12	くフローラー		> A CRC		0x40074000	Cyclic Redundancy Check Calcula
	● メモリー			> Topoc		0x40010000	Debug Function
	↓ メモリーブラウザー			> PDC		0x40034100	Data Operation Circuit
	🛋 モジュール					0x40003400	Event Link Controller
	お辞 レジスター					0x40041000	Each I/O Registers
	ジェアセンブル			GPT164		0x40078400	General PWM 16-bit Timer 4
	60 x			> - GPT165		0x40078500	General PWM 16-bit Timer 5
	() 案行可能7ァイ)			> 5 GPT166		0x40078600	General PWM 16-bit Timer 6
				> 5 GPT167		0x40078700	General PWM 16-bit Timer 7
			414-21-014	> 5 GPT168		0x40078800	General PWM 16-bit Timer 8
	···· 20:50		AIT+97F+QV	> 🛃 GPT169		0x40078900	General PWM 16-bit Timer 9
	[]] []] 题		Alt+シフト+Q,X	> 🛃 GPT320		0x40078000	General PWM 32-bit Timer 0
	その他(o)		Alt+シフト+Q.Q	<			>

#### 図 9.1 [IO Registers]表示

9.2 [IO Registers] ビューのカスタマイズ

レジスタ名を右クリックし、[選択されたレジスターに追加]を選択する。[選択されたレジスター]タブで、 指定したレジスタのみが選択されていることを確認できます。

例: AGT1 レジスタ



# RA2E3 HS4001 低消費電力センサシステム実装例

変数 ブレークポイント	プロジェクト・コ	クスプローラー 式	11	ントポイント	IO Registers >	< "1 🗖	
		AGT ~	Û,	û 🗉 🖻 🍣	🔕 🎓 🔍 🚍	1	8
名前	値	アドレス		詳細			^
> 🚼 AGT0		0x40084000		Low Power Asy	ynchronous Gen	eral Purpose Ti	ï
✓ 3 AGT1		0x40084100		Low Power Asy	ynchronous Gen	eral Purpose Ti	ï
1010 AGT		0.40004100		AGT Counter R	Register		
1889 AGTCMA	(二) 選択され	たレシスターに追加		AGT Compare	Match A Regist	er	
State AGTCMB	(字) 選択され	たレジスターから除去		AGT Compare	Match B Regist	er	
> 1010 AGTCR	📄 他のグル	- プをすべて閉じる		AGT Control R	egister		
> IIII AGTMR1	🧈 リフレッシ	Lvyシュ		AGT Mode Reg	gister 1		
> IIII AGTMR2	ス リフレッシ	ュ抑止		AGT Mode Reg	gister 2		
> IIII AGTIOC		ー・・ーー ブループを展開		AGT I/O Contro	ol Register		
> IIII AGTISR		7問/3		AGT Event Pin	Select Register		
> Mon AGTCMSR		検索 選択されたレジスターの管理 展開されたグループを印刷		AGT Compare	Match Function	Select Registe	e
> III AGTIOSEL	🔍 検索			AGT Pin Select	t Register		
> The BUS	🛛 🎓 選択され			BUS Control			-
> TA CAC	📑 展開され			Clock Frequence	cy Accuracy Me	asurement Circ	51
	💾 保管		-	Dobug Functio		ulator	-
<	באב 🔝			Debug Functio		>	×
すべてのレジスター 選択されたレ	びん 🔲 値 (バイナ	⁺リ−)を表示					

### 図 9.2 [選択されたレジスターに追加]

変数 ブレークボイント	プロジェクト・エクスプロー	-ラ- 式 イ^	ベントポイント IO Registers 🗙 🔭 🗖 🗖
	AGT	~ 🗘	🕆 🕀 🛱 🌫 🔕 🎓 🔍 🚍 🛅 📄 🛃 🕴
名前	値	アドレス	詳細
∽ 🛼 AGT1		0x40084100	Low Power Asynchronous General Purpose Time
😽 AGT	0x2f59	0x40084100	AGT Counter Register
😽 AGTCMA	0x2000	0x40084102	AGT Compare Match A Register
🖓 AGTCMB	0x2000	0x40084104	AGT Compare Match B Register
> 😽 AGTCR	0x03	0x40084108	AGT Control Register
> 🔛 AGTMR1	0x61	0x40084109	AGT Mode Register 1
> 🔐 AGTMR2	0x00	0x4008410a	AGT Mode Register 2
> 🔐 AGTIOC	0x05	0x4008410c	AGT I/O Control Register
> 鍋! AGTISR	0x00	0x4008410d	AGT Event Pin Select Register
> 🔐 AGTCMSR	0x00	0x4008410e	AGT Compare Match Function Select Register
> 🕍 AGTIOSEL	0x00	0x4008410f	AGT Pin Select Register
<	-		>
すべてのレジスター 選択されたレジス	9-		

図 9.3 [選択されたレジスター]

### 9.3 ブレークポイントの設定

デバッガ起動時、エディタのアドレスが表示されているエリア内の行をダブルクリックするとブレークポイントが設定されます。

そのエリア内の行を右クリックすると、[Toggle Software Breakpoint] または [Toggle Hardware Breakpoint] を選択することで、ブレークポイントの種類を直接選択することができます。



### 図 9.4 ブレークポイントの設定



244 246 247 00002bde 248 00002be6 249 00002bea 250 00002bf0 251 252	<pre>     * @brief Common callback function called if     void rm_comms_i2c_callback (i2c_master_call     {         fsp_err_t err = FSP_SUCCESS;         rm_comms_i2c_instance_ctrl_t * p_ctrl         i2c_master_instance_t * p_driver         rm_comms_callback_args_t comms_i2;     } } </pre>	<pre>in the I2C driver callback function. lback_args_t * p_args)                                 = (rm_comms_i2c_instance_ctu r_instance = (i2c_master_instance_t *) 2c_args;</pre>
右クリック 253 00002bf8	Toggle Software Breakpoint Toggle Hardware Breakpoint	
257 258 259 00002c0e 260	<b>ブレークポイントの切り替え(K)</b> ブレークポイントを追加(A) Add Dynamic Printf	<b>Ctrl+シフト+B</b> Ctrl+ダブル・クリック
	ブレークポイントを使用不可にする(D) ブレークポイント・プロパティー(P) ブレークポイント型(R)	シフト+ダブル・クリック Ctrl+ダブル・クリック >
	ジレーフボイフド型(K) Switch Default e2 studio Breakpoint type to S 選択したファイルのビルド(F)	Software Ctrl+Alt+シフト+B
	選択したファイルのクリーン(L)	Ctrl+Alt+シフト+C

図 9.5 ブレークポイントの種類の選択

10. サンプルプロジェクト

サンプルプロジェクトは、ルネサスエレクトロニクス Web サイトから入手できます。

11. 参考ドキュメント

下記ドキュメントの最新版は、ルネサスエレクトロニクス Web サイトから入手できます。

- RA2E3 Fast Prototyping Board (R20UT5128)
   <u>https://www.renesas.com/document/mat/fpb-ra2e3-v1-users-manual</u>
- RA2E3 ユーザーズマニュアル ハードウェア編 (R01UH0992)
   <a href="https://www.renesas.com/document/mah/ra2e3-group-users-manual-hardware">https://www.renesas.com/document/mah/ra2e3-group-users-manual-hardware</a>
- RA ファミリ、RX ファミリ、RL78 ファミリ、RZ ファミリ HS400x サンプルソフトウェアマニュアル (R01AN6333)



# 改訂記録

		改訂内容					
Rev.	発行日	ページ	ポイント				
Rev.1.01	May.26. 25	-	初版				



#### 製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテク ニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部 リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオン リセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入に より、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」について の記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識 されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した 後に切り替えてください。リセット時、外部発振子(または外部発振回路)を用いたクロックで動作を開始するシステムでは、クロックが十分安定 した後、リセットを解除してください。また、プログラムの途中で外部発振子(または外部発振回路)を用いたクロックに切り替える場合は、切り 替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、V<sub>IL</sub>(Max.)からV<sub>IH</sub>(Min.)までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、V<sub>IL</sub>(Max.)からV<sub>IH</sub>(Min.)までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

リザーブアドレス(予約領域)のアクセス禁止
 リザーブアドレス(予約領域)のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス(予約領域)があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッ シュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ幅射量などが異なる場合が あります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

### ご注意書き

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

標準水準: コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等 高品質水準:輸送機器(自動車、電車、船舶等)、交通制御(信号)、大規模通信機器、金融端末基幹システム、各種安全制御装置等 当社製品は、データシート等により高信頼性、Harsh environment向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のあ る機器・システム(生命維持装置、人体に埋め込み使用するもの等)、もしくは多大な物的損害を発生させるおそれのある機器・システム(宇宙機 器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等)に使用されることを意図しておらず、これら の用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その 責任を負いません。

- 7. あらゆる半導体製品は、外部攻撃からの安全性を100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリ ティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害(当社製品または当社製品が使用されてい るシステムに対する不正アクセス・不正使用を含みますが、これに限りません。)から生じる責任を負うものではありません。当社は、当社製品ま たは当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行 為(「脆弱性問題」といいます。)によって影響を受けないことを保証しません。当社は、脆弱性問題に起因しまたはこれに関連して生じた損害に ついて、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品 性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
- 8. 当社製品をご使用の際は、最新の製品情報(データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等)をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
- 9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする 場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を 行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客 様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を 行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行って ください。
- 10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用 を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことに より生じた損害に関して、当社は、一切その責任を負いません。
- 11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
- 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
- 13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
- 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的 に支配する会社をいいます。
- 注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

#### 本社所在地

〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア) www.renesas.com

#### 商標について

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

### お問合せ窓口

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