

RA ファミリ、RX ファミリ、RL78 ファミリ

FS3000 サンプルソフトウェアマニュアル

要旨

本アプリケーションノートでは、RA ファミリ、RX ファミリ、RL78 ファミリで動作する FS3000 フロー センサのサンプルソフトウェアについて説明します。

動作確認デバイス

RA0E1 グループ

- RA2E1 グループ
- RX65N グループ
- RL78/G23 グループ

動作確認センサボード

Air Velocity Sensor Pmod[™] Board (US082-FS3000EVZ)

本アプリケーションノート記載の設定例は、上記のセンサボードを使用する場合の例です。

- そのため、組み込み対象回路に合わせて、以下の設定を見直す必要があります。
- 割り込み信号回路: 「6.4 割り込み信号回路に関する注意」を参照してください。
- --- RESET 信号回路: 「6.5 RESET 信号回路に関する注意」を参照してください。

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分 評価してください。



目次

1.	概要	4
1.1	用語/略語	4
2.	動作確認環境	5
2.1	RA 動作確認環境	5
2.2	RX 動作確認環境	8
2.3	RL78 動作確認環境	9
3.	FS3000 センサ仕様	.10
3.1	センサ仕様概要	10
3.2	センサ機能	10
4	ᅭᆞᆕᇻᇧᇧᆿᅣᅭᆮᇴᄮᄰ	4.4
4.	リンフルノフトワエア 仕様	.
4.1	サンフルソフトワエア構成	.11
4.2	Sensor Control Module API 関数仕様	. 11
4.2.1	Sensor Control Module API	.11
4.2.2	・ API (2月カイト	12
4.3	サフフルサフトウェア Non-OS 版メイン処理フロー	13
4.4	リンフルフフトウェア 03 版タイン処理フロー	15
5.	Configuration 設定	.18
5.1	。 FS3000 Sensor Control Module 設定	18
5.1.1	RA ファミリ	18
5.1.2	RX ファミリ	18
5.1.3	RL78 ファミリ	19
5.2	I2C Communication Middleware (COMMS_I2C)設定	20
5.2.1	RA ファミリ	20
5.2.2	RX ファミリ	21
5.2.3	RL78 ファミリ	22
5.3	I2C ドライバ設定	23
5.3.1	RA ファミリ	23
5.3.2	RX ファミリ	26
5.3.3	RL78 ファミリ	29
6.	デバイス変更ガイド	.30
6.1	RA サンプルプロジェクト	30
6.1.1	サンプルプロジェクトのインポート	30
6.1.2	FSP Configurator の設定変更	32
6.1.3	ツールチェイン設定変更	37
6.2	RX サンプルプロジェクト	38
6.2.1	サンプルプロジェクトのインポート	38
6.2.2	デバイスの変更	40
6.2.2 6.2.3	デバイスの変更 Smart Configurator 設定の変更	40 42
6.2.2 6.2.3 6.2.4	デバイスの変更 Smart Configurator 設定の変更 ツールチェイン設定変更	40 42 45



6.3.1	サンプルプロジェクトのインポート	46
6.3.2	2 デバイスの変更	48
6.3.3	3 Smart Configurator 設定の変更	50
6.3.4	↓ 生成コードの変更	
6.3.5	う ツールチェイン設定変更	55
6.3.6	ジ LLVM ビルド時の注意	57
6.4	割り込み信号回路に関する注意	58
6.5	RESET 信号回路に関する注意	58
6.6	Renesas Pmod Sensor Board のデイジ・チェーン接続時のプルアップ抵抗処理	58
7.	風速データの確認方法	59
改訂	記録	61

商標

FreeRTOS[™]は Amazon Web Serices, Inc.の登録商標です。

Pmod[™]は Digilent Inc.の商標です。



1. 概要

本ソフトウェアは、FS3000-1005 風速センサのデータの取得、および演算を行うためのサンプルプログ ラムです。MCU に内蔵されている I2C を用い RA FSP、RX FIT、RL78 コードジェネレータの I2C ドライ バとの組み合わせによってセンサから ADC データの取得および風速値の演算を行います。

1.1 用語/略語

用語とその略語を以下に示します。

用語	略語
FS3000 Sensor Control Module	Sensor Control Module
	RA MCU の場合、"rm_fs3000"
	RX MCU の場合、"r_fs3000_rx"
	RL78 MCU の場合、"r_fs3000"
I2C Communication Middleware	COMMS_I2C
	RA MCU の場合、"rm_comms_i2c"
	RX MCU の場合、"r_comms_i2c_rx"
	RL78 MCU の場合、"r_comms_i2c"
I2C ドライバ	RA MCU の場合、"r_iic_master", "r_sci_i2c", "r_iica_master"
	RX MCU の場合、"r_riic_rx", "r_sci_iic_rx"
	RL78 MCU の場合、"r_iica_master"
Serial Communications Interface	RA MCU の場合、"SCI"もしくは"SCI I/F"
	RX MCU の場合、"SCI"もしくは"SCI I/F"
Serial Array Unit	RA MCU の場合、"SAU"もしくは"SAU I/F"
	RL78 MCU の場合、"SAU"もしくは"SAU I/F"
I2C Bus Interface	RA MCU の場合、"IIC"もしくは"IIC I/F"
	RX MCU の場合、"RIIC"もしくは"RIIC I/F"
I2C Bus Interface (IICA)	RA MCU の場合、"IICA"もしくは"IICA I/F"
Serial Interface IICA	RL78 MCU の場合、"IICA"もしくは"IICA I/F"
I2C Bus Interface、	"I2C I/F"
I2C Bus Interface (IICA)、	
Serial Interface (IICA)	
の総称	
割り込みコントローラの総称	"ICU I/F" (Interrupt Controller Unit)
汎用 I/O ポート	"GPIO"もしくは"GPIO I/F"
Renesas Pmod Type 6A センサ ボードの1ピン(#1)	"IRQ#"(割り込み発生時にL出力)

表 1-1 用語/略語一覧



2. 動作確認環境

2.1 RA 動作確認環境

本ソフトウェアの RA 動作確認環境を以下に示します。

(1) Evaluation Kit for RA2E1 (EK-RA2E1)

表 2	2-1	EK-RA2E1	動作確認環境
-----	-----	----------	--------

項目	内容
デモボード	RTK7EKA2E1S00001BE (EK-RA2E1)
使用マイコン	RA2E1 (R7FA2E1A92DFM:64pin)
動作周波数	48MHz
動作電圧	5V
統合開発環境	Renesas Electronics e ² studio 2024-07
Cコンパイラ	GNU ARM Embedded 13.2.1.arm-13-7
オプション設定	ISO C99 (-std = c99)
要確認	最適化レベル:デフォルト設定 (-O2)
FSP	v5.5.0
RTOS	FreeRTOS v10.6.1
エミュレータ	On board (J-LINK)
変換ボード	Interposer Board for Pmod Type2/3 to 6A (US082-INTERPEVZ)
センサボード	Air Velocity Sensor Pmod Board (US082-FS3000EVZ)

表 2-2 RA2E1 使用メモリ量

領域	サイズ (Non-OS) [Bytes]	サイズ (FreeRTOS) [Bytes]		
ROM	1,548	3,478 (注 1)		
RAM	136	372		

注 メモリサイズはサンプルコードと FS3000 Sensor Control Module、COMMS_I2C を対象に計算してい ます。FreeRTOS の場合は、スレッドのメモリサイズは計算に含めていません。

注1 Relax 機能による 1,572 バイト増加分を含みます。



図 2-1 EK-RA2E1 HW 接続図



(2) RA0E1 Fast Prototyping Board (FPB-RA0E1)

表 2-3 FPB-RA0E1 重	鳨作確認環境
-------------------	--------

項目	内容
デモボード	RTK7FPA0E1S00001BJ (FPB-RA0E1)
使用マイコン	RA0E1 (R7FA0E1073CFJ:32pin)
動作周波数	32MHz
動作電圧	5V
統合開発環境	Renesas Electronics e ² studio 2024-07
Cコンパイラ	GNU ARM Embedded 13.2.1.arm-13-7
オプション設定	ISO C99 (-std = c99)
	最適化レベル:デフォルト設定 (-Oz)
FSP	v5.5.0
エミュレータ	On board (J-LINK)
変換ボード	Interposer Board for Pmod Type2/3 to 6A (US082-INTERPEVZ)
センサボード	Air Velocity Sensor Pmod Board (US082-FS3000EVZ)

表 2-4 RA0E1 使用メモリ量

領域	サイズ [Bytes]
ROM	1,306
RAM	124
注 メモリサイズはち	ナンプルコードと FS3000 Sensor Control Module、COMMS_I2C を対象に計算してい

ます。



図 2-2 FPB-RA0E1 HW 接続図



(3) Interposer Board 利用について

Interposer Board は、SCI I/F の Pmod Type 2A/Type 3A コネクタに対して、簡易 IIC 機能に切り替えることで、Pmod Type 6A センサを接続ための I/F 変換ボードです。

そのため、SAU I/F の Pmod Type 2A/Type 3A コネクタに対しては、適用できません。ただし、IICA I/F に 切り替えることで利用できる場合があります。MCU のハードウェアマニュアルを参照してください。

Pmod I/F	接続先の MCU Serial I/F	動作可否			
Type 2A、	SCI I/F、	 Interneggy Deard を利田士でに動作します。 (注 2)			
Туре ЗА	IICA I/F(注 1)	Interposer Board を利用すると動作します。 (注 2)			
	SAU I/F	Interposer Board 有無に関わらず、動作しません。			
Type 6、	SCI I/F、IIC I/F	Interneour Poord 毎」で動作します。(注 2)			
Туре 6А	SAU I/F、IICA I/F	Interposer Board 無して動作します。 (注 2)			

表 2-5 Pmod I/F と Serial I/F と Interposer Board 有無による動作可否

注1: SAU I/F 用に設けられた端子ですが、マルチファンクション端子割り当てにより、IICA 端子に切り替え 可能な場合に適用できます。切り替え可能な場合の信号接続を以下に示します。

Pmod Pin	Type 2A /Type 3A	接続先 SAU I/F ICU I/F GPIO I/F	Multi- Function IICA I/F 切り替え		Interposer Board		Renesas Pmod Type 6A Sensor Board
#1	CS/CTS	GPIO		\leftrightarrow		\leftrightarrow	IRQ# (注3)
#2	MOSI/TXD	SAU TXD	SDAA	\leftrightarrow		\leftrightarrow	RESET#
#3	MISO/RXD	SAU RXD	SCLA	\leftrightarrow		\leftrightarrow	IIC_SCL
#4	SCK/RTS	GPIO		\leftrightarrow		\leftrightarrow	IIC_SDA
#7	INT	IRQ#		\leftrightarrow		\leftrightarrow	BUSY#
#8	RESET	GPIO		\leftrightarrow	\blacklozenge	\leftrightarrow	ENABLE
#9	CS2/GPIO	GPIO		\leftrightarrow	\bullet	\leftrightarrow	POWER_ON
#10	CS3/GPIO	GPIO		\leftrightarrow	$\mathbf{+}$	\leftrightarrow	GPIO

注2: IRQ 信号を利用する場合、Pmod Sensor Board 上の Pmod #1 に IRQ 信号が接続されていることを確認してください。

注3:割り込み信号回路に関して、「6.4 割り込み信号回路に関する注意」を参照してください。

適用例:FPB-RA0E1 PMOD1 が該当します。



図 2-3 FPB-RA0E1 の PMOD1 Type 2A/Type 3A を使った IICA 利用の場合の接続図



2.2 RX 動作確認環境

本ソフトウェアの RX 動作確認環境を以下に示します。

(1) RX65N Envision Kit

表	2-6	RX65N Envision Kit 動作確認環境
衣	∠-0	KADDIN EIIVISION KIL 對作唯認現場

項目	内容
デモボード	RPBRX65N (RX65N Envision Kit)
使用マイコン	RX65N (R5F565NEDDFB: 144pin)
動作周波数	120MHz
動作電圧	5V
統合開発環境	Renesas Electronics e ² studio 2023-01
Cコンパイラ	Renesas Electronics CC-RX V.3.02.00
オプション設定	C99 言語(-lang = c99)
	最適化レベル:デフォルト設定 (レベル 2)
FIT	Board Support Packages (r_bsp) v7.20
	FS3000 Sensor Middleware (r_fs3000_rx) v1.00
	IIC Communication Driver Interface Middleware (r_comms_i2c_rx) v1.20
	RIIC Multi Master I2C Driver (r_riic_rx) v2.49
	Simple IIC Driver (r_sci_iic_rx) v2.49
RTOS	FreeRTOS Kernal 10.4.3-rx-1.0.1、FreeRTOS Object 10.4.3-rx-1.0.1
エミュレータ	On board (E2OB)
変換ボード	Interposer Board for Pmod Type2/3 to 6A (US082-INTERPEVZ)
センサボード	Air Velocity Sensor Pmod Board (US082-FS3000EVZ)

表 2-7 RX65N 使用メモリ量

領域	サイズ(Non-OS) [Bytes]	サイズ(FreeRTOS) [Bytes]
ROM	1,638	1,844
RAM	118	141

注 メモリサイズはサンプルコードと FS3000 Sensor Control Module、COMMS_I2C を対象に計算してい ます。FreeRTOS の場合は、スレッドのメモリサイズは計算に含めていません。



図 2-4 RX65N Envision Kit HW 接続図

(2) Interposer Board 利用について

SCI I/F が接続された Pmod Type 2A/Type 3A コネクタに、Interposer Board を追加した場合、Pmod Type 6A Sensor Pmod Board を使用できます。



2.3 RL78 動作確認環境

本ソフトウェアの RL78 動作確認環境を以下に示します。

(1) RL78/G23-128p Fast Prototyping Board (RL78/G23-128p FPB)

表 2	2-8	RL78/G23-128p	FPB	動作確認環境
-----	-----	---------------	-----	--------

項目	内容
デモボード	RTK7RLG230CSN000BJ (RL78/G23-128p FPB)
使用マイコン	RL78/G23 (R7F100GSN2DFB: 128pin)
動作周波数	32MHz
動作電圧	3.3V
統合開発環境	Renesas Electronics e ² studio 2024-07
Cコンパイラ	Renesas Electronics CC-RL V1.14.00
	LLVM for RL78 17.0.1.202409
オプション設定	コンパイラのデフォルト設定に対して、以下の設定を追加
	CC-RL :C99 言語規格(-lang = c99)、最適化レベル デフォルト設定 (-Odefault)
	LLVM :GNU ISO C99 (-std = gnu99)、最適化レベル デフォルト設定 (-Og)
SIS / CG	Board Support Packages (r_bsp) v1.70
	FS3000 Sensor Middleware (r_fs3000) v1.02
	IIC Communication Driver Interface Middleware (r_comms_i2c) v1.11
	IIC 通信 (マスタモード) v1.6.0
エミュレータ	On board (COM Port)
センサボード	Air Velocity Sensor Pmod Board (US082-FS3000EVZ)

表 2-9 RL78/G23-128p FPB 使用メモリ量

領域	サイズ [Bytes] (CC-RL)	
ROM	1,905	
RAM	85	

注 メモリサイズはサンプルコードと FS3000 Sensor Control Module、COMMS_I2C を対象に計算してい ます。



図 2-5 RL78/G23-128p FPB HW 接続図

(2) Interposer Board 利用について

Interposer Board は、RA/RX 用 SCI I/F の Pmod Type 2A/Type 3A コネクタに対して、簡易 IIC 機能に切り替えることで、Pmod Type 6A センサを接続ための I/F 変換ボードです。

そのため、SAU I/F が接続された Pmod Type 2A/Type 3A コネクタに、Interposer Board を追加した場合 であっても、Pmod Type 6A Sensor Pmod Board を**使用できません。**



3. FS3000 センサ仕様

3.1 センサ仕様概要

FS3000 フローセンサは、MEMS サーモパイルベースのセンサを利用した表面実装タイプの風速モジュー ルです。FS3000 は、12 ビット分解能のデジタル出力を備えています。モジュールの特性を表すパラメータ 等、センサモジュールについては、FS3000 のデータシートを参照してください。

3.2 センサ機能

FS3000 サンプルソフトウェアではフローセンサのうち FS3000-1005 風速センサに対応しています。 (FS3000-1015 センサは未対応)

センサは電源投入と同時に測定を開始します。

センサデータを取得する場合、Flow Data Read コマンドを送信し、それに続いて、5 バイトのデータを Read します。コマンドについては、<u>FS3000</u>のデータシートを参照してください。

風速データは 12bits で、2 バイト目(Byte 2)の下位 4bit が有効なデータになります。

4. サンプルソフトウェア仕様

サンプルソフトウェアパッケージには RA2E1 グループの Non-OS 版と FreeRTOS 版、RA0E1 グループ の Non-OS 版、RX65N グループの Non-OS 版と FreeRTOS 版、RL78/G23 グループの Non-OS 版(CC-RL / LLVM)の合計 7 つのプロジェクトが含まれます。

RXの FreeRTOS の設定方法については FAQ を参照してください。

4.1 サンプルソフトウェア構成

サンプルソフトウェアのレイヤ構成を以下に示します。

図 4-1 ソフトウェアレイヤ図

4.2 Sensor Control Module API 関数仕様

4.2.1 Sensor Control Module API 関数一覧

Sensor Control module API は以下の関数が含まれます。

関数 API の詳細は以下を参照してください。

RA Flexible Software Package Documentation, Renesas Sensor Control Modules Firmware Integration Technology (R01AN5892) Renesas Sensor Control Modules Software Integration System (R01AN6192)

関数	機能
RM_FS3000_Open()	センサ制御開始処理
RM_FS3000_Close()	センサ制御終了処理
RM_FS3000_Read()	センサデータ取得処理
RM_FS3000_DataCalculate()	センサデータ結果演算処理

表 4-1	Sensor	Control	Module	API	関数一	-覧
-------	--------	---------	--------	-----	-----	----

4.2.2 API 使用ガイド

API 関数の使用条件について、想定する関数コールの順番を遷移図として示します。

図 4-2 関数 API 遷移図

関数毎の呼び出し条件は以下の通りです。

- RM_FS3000_Open():
- RM_FS3000_Close():
- RM_FS3000_Read():
- (1) FS3000 開始時、(7) RM_FS3000_Close()後の再開始
- (6) 各処理の正常終了または異常終了時
- (2) 開始後の測定データ取得時
- (5) データ取得応答待ちによる再試行
- RM_FS3000_DataCalculate():
- (3) RM_FS3000_Read()後のデータ演算時
- 注意: OS 使用時、複数のスレッド/タスクで同時にセンサを制御する場合はユーザによるセマフォを用いた bus 制御が必要となります。セマフォの生成タイミング、ブロッキングの制御は「OS 版サンプル ソフトウェアフロー」を参考にしてください。

4.3 サンプルソフトウェア Non-OS 版メイン処理フロー

サンプルソフトウェアはドライバの開始処理を行い、その後はセンサデータ取得、測定結果演算の処理を 繰り返します。

図 4-3 FS3000 サンプルソフトウェア Non-OS 版メイン処理フロー1

図 4-4 FS3000 サンプルソフトウェア Non-OS 版メイン処理フロー2

4.4 サンプルソフトウェア OS版メイン処理フロー

OS版ではセマフォによる制御を行い、センサ制御を行う2つのスレッドを並列で動作させます。

それぞれセンサの制御はドライバの開始処理を行い、その後はセンサデータ取得、測定結果演算の処理を 繰り返します。

図 4-5 FS3000 サンプルソフトウェア OS版メイン処理フロー1

図 4-6 FS3000 サンプルソフトウェア OS 版メイン処理フロー2

図 4-7 FS3000 サンプルソフトウェア OS 版メイン処理フロー3

5. Configuration 設定

設定可能な項目と設定値は以下のとおりです。

緑色の設定値はデフォルトで選択されている項目、<mark>橙色</mark>の設定値は変更不可の項目です。 モジュール名やコールバック関数名に関して、C 言語標準に準拠した名前を指定してください。 最新版利用時、以下に示す設定項目や設定値が異なる場合が有ります。

5.1 FS3000 Sensor Control Module 設定

5.1.1 RA ファミリ

FSP Configurator の Stack タブで"**rm_fs3000**"の Stack を選択することにより、Properties タブに設定可能な項目が表示されます。

設定項目	設定値	説明
Common		
Parameter Checking	Default (BSP)	パラメータチェック処理をコードに含めるか選択できま
	Enabled	す。
	Disabled	"Disabled"の場合、コードから省略します。
		"Enabled"の場合、をコードに含めます。
Device type	FS3000-1005	センサの種別を設定します。
		"FS3000-1005"のみ設定できます。
Module g_fs3000_senso	or FS3000 on rm_fs3000	
Name	g_fs3000_sensor0	モジュール名を設定します。
Callback	fs3000_callback	ユーザコールバック関数名を設定します。
		NULL を設定した場合は、コールバック関数は使用され
		ません。

表	5-1	RA	FS3000	設定-	-覧
---	-----	----	--------	-----	----

5.1.2 RX ファミリ

Smart Configurator の Component タブで"r_fs3000_rx"コンポーネントを選択することにより、Configure 領域に設定可能な項目が表示されます。

設定項目	設定値	説明	
Configurations			
Parameter Checking	System Default	パラメータチェック処理をコードに含めるか選択で	
	Enabled	きます。	
	Disabled	"Disabled"の場合、コードから省略します。 "Enabled"の想合、コードから省略します。	
		Enabledの場合、コートに含めます。	
Number of FS3000 sensors	1	接続する FS3000 センサの数を設定します。	
	2		
Device type of FS3000	FS3000-1005	センサの種別を設定します。	
Sensors		"FS3000-1005"のみ設定できます。	
I2C Communication device No.	I2C Communication	I2C 通信デバイス番号を設定します。	
for FS3000 sensor device{x}	Device{y}		
(x = 0 or 1)	(y = 0 - 15)		
Callback function for FS3000	fs3000_user_callback{x}	ユーザコールバック関数名を設定します。	
sensor device{x}	(x = 0 or 1)	NULL を設定した場合は、コールバック関数は使用	
(x = 0 or 1)		されません。	

表 5-2 RX FS3000 設定一覧

5.1.3 RL78 ファミリ

Smart Configurator の Component タブで"**r_fs3000**"コンポーネントを選択することにより、Configure 領域に設定可能な項目が表示されます。

設定項目	設定値	説明	
Configurations			
Parameter Checking	System Default	パラメータチェック処理をコードに含めるか選択で	
	Enabled		
	Disabled	"Disabled"の場合、コードから省略します。 "Enabled"の場合、コードに含めます。	
Number of FS3000 sensors	1	接続する FS3000 センサの数を設定します。	
	2		
Device type of FS3000 Sensors	FS3000-1005	センサの種別を設定します。	
	FS3000-1015	FS3000-1005 のみ設定できます。	
I2C Communication device No. for FS3000 sensor device{x} (x = 0 or 1)	Comms{y} (y = 0 - 4)	I2C 通信デバイス番号を設定します。	
Callback function for FS3000 sensor device{x} (x = 0 or 1)	fs3000_user_callback{x} (x = 0 or 1)	ユーザコールバック関数名を設定します。 NULLを設定した場合は、コールバック関数は使用 されません。	

表 5-3 RL78 FS3000 設定一覧

5.2 I2C Communication Middleware (COMMS_I2C) 設定

5.2.1 RA ファミリ

FSP Configurator の Stack タブで"**rm_comms_i2c**"の Stack を選択することにより、Properties タブに設 定可能な項目が表示されます。

表 5-4	RA COMMS	_l2C 設定-	·覧
-------	----------	----------	----

設定項目	設定値	説明
Common		
Parameter Checking	Default (BSP)	パラメータチェック処理をコードに含めるか
	Enabled	選択できます。
	Disabled	Disabledの場合、コードから省略します。
		Enabledの場合、コードに含めます。
Module g_comms_i2c_device0 I2	C Communication Device (rm_comms	<u>i</u> 2c)
Name	g_comms_i2c_device0	モジュール名を設定します。
Semaphore Timeout	0xFFFFFFFF	RTOS プロジェクト時、semaphore のタイ ムアウト時間を設定します。
Slave Address	0x28	スレーブアドレスを設定します。
		Sensor Control module により上書きされる ため、設定不要です。
Address	7-Bit	スレーブアドレスのビット幅を設定します。
Mode		Sensor Control module により上書きされる ため、設定不要です。
Callback	rm_fs3000_comms_i2c_callback	ユーザコールバック関数名を設定します。
		Sensor Control module により上書きされる
		ため、設定不要です。
Module g_comms_i2c_bus0 I2C S	Shared Bus (rm_comms_i2c)	
Name	g_comms_i2c_bus0	I2C モジュール名を設定します。
Bus Timeout	0xFFFFFFF	I2C バスのタイムアウト時間を設定します
Semaphore for blocking	Unuse	RTOS プロジェクト時、Blocking 処理の有
	Use	効/無効を設定します。
Recursive Mutex for Bus	Unuse	RTOS プロジェクトかつ、Blocking 処理が
	Use	有効の時、再帰動作の有効/無効を設定しま す。
Channel	0	使用するチャネル番号を設定します。
		I2C ドライバが r_iic_master の場合のみ、こ
		の設定が有効です。
		他の I2C ドライバの場合、この設定は無効
Pate	Standard	じり。
Trate	East mode	こうドレードを設定します。 I2C ドライバが riic masterの場合のみ、こ
		の設定が有効です。
	Fast-mode plus	FS3000 の場合は、Standard もしくは Fast-
		mode の設定が可能です。同一バス上の他デ
		バイスが接続されている場合、そのデバイス
		の設定可能な転送レートも考慮して設定して
		他のI2Cドライバの場合、この設定は無効で
		9 o

5.2.2 RX ファミリ

Smart Configurator の Component タブで"**r_comms_i2c_rx**"コンポーネントを選択することにより、 Configure 領域に設定可能な項目が表示されます。

設定項目	設定値	説明
Configurations		
Parameter Checking	System Default	パラメータチェック処理をコードに含めるか選択でき
	Enabled	ます。
	Disabled	Disabled の場合、コードから省略します。
		Enabled の場合、コードに含めます。
Number of I2C Shared	Unused	接続可能とする I2C バス数を設定します。
Buses	1	
	2 - 16	
Number of I2C	Unused	接続可能とする I2C デバイスを設定します。
Communication Devices	1	
	2 - 16	
Blocking operation	Disabled	RTOS プロジェクト時のブロッキング動作を設定し
supporting with RTOS	Enabled	ます。
Bus lock operation	Disabled	RTOS プロジェクト時のバスロック動作を設定しま
supporting with RTOS	Enabled	す。
I2C Driver Type for I2C	RIIC	通信バスが使用する I2C バスの種別を設定します。
Shared bus{x}	SCI IIC	RIIC を使用する場合は、r riic rx, SCI IIC を使用する
(x = 0 - 15)	Not selected	場合は、r_sci_iic_rx が必要となります。
		使用しない FIT モジュールを削除すると、警告が表示
		されますが、動作に問題はありません。
Channel No. for I2C Shared	0	通信バスが使用する I2C バスのチャネル番号を設定
bus{x}		します。
(x = 0 - 15)		
Timeout for the bus lock of	0xFFFFFFF	I2C バスの I2C バスロックタイムアウト時間を設定し
I2C Shared Bus{x}		ます。
(x = 0 - 15)		
Communication Device/x	$12C$ Shared Bus{x}	通信ハスが使用する120 ハスのコンフィグレーショ
(x = 0 - 15)	(x = 0 - 15)	ンを設定します。
Slave address for I2C	0x00	通信バスに接続されるデバイスのスレーブアドレスを
Communication device{x}		設定します。
(x = 0 - 15)		FS3000の場合は、0x28に設定してください。
Address mode for I2C	7 bit address mode	スレーブアドレスモードを設定します。
Communication device{x}		FS3000の場合は、7 bit address mode に設定してく
(x = 0 - 15)		
Callback function for I2C	comms i2c user callback{x}	ユーザコールバック関数名を設定します。
Communication device{x}	(x = 0 - 15)	r fs3000 rx を使用する場合は.
(x = 0 - 15)		rm fs3000 callback{v} (v = 0 or 1)を設定します。
Timeout for the blocking bus	0xFFFFFFFF	I2C バスの I2C バスブロッキングタイムアウト時間を
of I2C Communication		設定します。
device{x}		
(x = 0 - 15)		

表 5-5 RX COMMS_I2C 設定一覧

5.2.3 RL78 ファミリ

Smart Configurator の Component タブで"r_comms_i2c"コンポーネントを選択することにより、 Configure 領域に設定可能な項目が表示されます。

設定項目	設定値	説明		
Configurations				
Parameter Checking	System Default	パラメータチェック処理をコードに含めるか選択で		
	Enabled	きます。		
	Disabled	Disabled の場合、コードから省略します。		
		Enabled の場合、コードに含めます。		
Number of I2C Shared	Unused	接続可能とする通信バス数を設定します。		
Buses	1			
	2 - 5			
Number of I2C	Unused	接続可能とする I2C デバイス数を設定します。		
communication Devices	1			
	2 - 5			
I2C Driver Type for I2C	IICA	通信バスが使用する I2C バスの種別を設定します。		
Shared Bus x (x = 0 - 4)	SAU IIC	FS3000 の場合は、IICA に設定してください。		
	Not selected			
Component name for the	Config_IIC00	通信バスが使用する I2C バスのコンポーネント名を		
I2C Bus{x}		指定します。		
(x = 0 - 4)				
I2C Shared Bus No. for I2C	I2C bus0	通信バスが使用する I2C バスのコンフィグレーショ		
Communication Device{x}	I2C bus1	ンを設定します。		
(x = 0 - 4)	I2C bus2			
	I2C bus3			
	I2C bus4			
Slave address for I2C	0x00	通信バスに接続されるデバイスのスレーブアドレス		
Communication Device{x}		を設定します。		
(x = 0 - 4)		FS3000 の場合は、0x28 に設定してください。		
Callback function for I2C	comms_i2c_user_callback{x}	ユーザコールバック関数名を設定します。		
Communication Device{x}	(x = 0 - 4)	r_fs3000 を使用する場合は、rm_fs3000_callback{y}		
(x = 0 - 4)		(y = 0 or 1)を設定します。		

表 5-6 RL78 COMMS_I2C 設定一覧

5.3 I2C ドライバ設定

5.3.1 RA ファミリ

FSP Configurator の Stack タブで"**r_iic_master**"、"**r_sci_i2c**"もしくは"**r_iica_master**"の Stack を選択 することにより、Properties タブに設定可能な項目が表示されます。

Simplified I2C using Serial Array Unit (SAU)は、クロック・ストレッチ機能を未サポートのため使用できません。

(1) r_iic_master

表 5-7 RA r_iic_master 設定一覧

設定項目	設定値	説明
Common		
Parameter Checking	Default (BSP)	パラメータチェック処理をコードに含めるか選択できます。
	Enabled	Disabled の場合、コードから省略します。
	Disabled	Enabled の場合、コードに含めます。
DTC on Transmission	Enabled	送受信に DTC を使用するか設定します。
and Reception	Disabled	
10-bit slave	Enabled	10-bit addressing をサポートするか設定します。
addressing	Disabled	COMMS_I2Cにより上書きされるため、設定不要です。
Module g_i2c_master0	I2C Master (r_iic_master)	
Name	g_i2c_master0	モジュール名を設定します。
Channel	0	使用するチャネル番号を設定します。
		COMMS_I2Cにより上書きされるため、設定は不要です。
Rate	Standard	ビットレートを設定します。
	Fast-mode	COMMS_I2Cにより上書きされるため、設定は不要です。
	Fast-mode plus	
Custom Rate (bps)	0	カスタムビットレートを設定します。
		0以外の場合に設定が有効です。Rateの設定範囲内の低ビット
		レートを設定したい場合に利用してください。
Rise Time (ns)	120	SCL の立ち上がり時間を設定します。使用するボードに合わせ
		て設定してください。
Fall Time (ns)	120	SCL の立ち下がり時間を設定します。使用するボードに合わせ
		て設定してください。
Duty Cycle (%)	50	SCL のデューティ比を設定します。
Slave Address	0x00	接続するデバイスのスレーブアドレスを設定します。
		COMMS_I2C により上書きされるため、設定は不要です。
Address Mode	7-Bit	接続するデバイスのスレーブアドレスモードを設定します。
	10-Bit	COMMS_I2C により上書きされるため、設定は不要です。
Timeout Mode	Short Mode	I2C バスのタイムアウト時間を設定します
	Long Mode	
Timeout during SCL	Enabled	SCLがタイムアウト モードで設定されている時間よりも長い時
low	Disabled	間 Low に保持されている場合にタイムアウトするか設定する。
Callback	rm_comms_i2c_callback	ユーザコールバック関数名を設定します。
		COMMS_I2C により上書きされるため、設定は不要です。
Interrupt Priority Level	Priority 0 (highest)	I2C バスドライバの割り込み優先レベルを設定します。
	Priority 1	
	Priority 2	
	Priority 3	
Pins	-	
SDA	Pxxx	ドライバが使用する端子番号が表示されます。
SCL	Pxxx	端子の設定は、Pins タブで行います。

(2) r_sci_i2c

表 5-8 RA r_sci_i2c 設定一覧

設定項目	設定値	説明
Common		
Parameter Checking	Default (BSP)	パラメータチェック処理をコードに含めるか選択できます。
	Enabled	Disabled の場合、コードから省略します。
	Disabled	Enabledの場合、コードに含めます。
DTC on Transmission	Enabled	送受信に DTC を使用するか設定します。
and Reception	Disabled	
10-bit slave	Enabled	10-bit addressing をサポートするか設定します。
addressing	Disabled	COMMS_I2Cにより上書きされるため、設定不要です。
Module g_i2c0 I2C Mas	ster (r_sci_i2c)	
Name	g_i2c0	モジュール名を設定します。
Channel	0	使用するチャネル番号を設定します。
Slave Address	0x00	接続するデバイスのスレーブアドレスを設定します。
		COMMS I2C により上書きされるため、設定不要です。
Address Mode	7-Bit	接続するデバイスのスレーブアドレスモードを設定します。
	10-bit	COMMS I2C により上書きされるため、設定不要です。
Rate	Standard	ビットレートを設定します。
	Fast-mode	FS3000の場合は、Standard もしくは Fast-mode の設定が可能
		です。同一バス上の他デバイスが接続されている場合、そのデ
		バイスの設定可能な転送レートも考慮して設定してください。
Custom Rate (bps)	0	カスタムビットレートを設定します。
		0以外の場合に設定が有効です。Rateの設定範囲内の低ビット
		レートを設定したい場合に利用してください。
SDA Output Delay	300	SDA 出力遅延時間を設定します。
(nano seconds)		
Noise filter setting	Use clock signal divided by 1 with noise filter	入力信号のノイズフィルタ使用を設定します。
	Use clock signal divided	
	Use clock signal divided	
	by 4 with noise filter	
	Use clock signal divided	
	by 8 with noise filter	
Bit Rate Modulation	Enable	Bit Rate Modulation 機能の使用を設定します。
	Disable	
Callback	rm_comms_i2c_callback	ユーザコールバック関数名を設定します。
		COMMS_I2C により上書きされるため、設定不要です。
Interrupt Priority Level	Priority 0 (highest)	I2C 割り込みの割り込み優先レベルを設定します。
	Priority 1	
	Priority 2	
	Priority 3	
RX Interrupt Priority	Priority 0 (highest)	DTC を使用した場合の受信割り込みの割り込み優先レベルを設
when DTC is	Priority 1	定します。
enabled]	Priority 2	
	Priority 3	
Dine	Disabled	
PINS	Duoy	ドニノデジはロナノ地マチロジェニナシャナナ
SDA		トフィハか使用りる「「「一番方か衣示されよう。
SUL	FXXX	「「「」」「「」」「「」」「「」」「「」」「」」「「」」「」」「」」「「」」「」」「」」「」」「」」「」」「」」「」」「」」「」」「」」「」」」「」」「」」「」」「」」」「」」」「」」」「」」」「」」」」

(3) r_iica_master

FSP v5.4.0 以降で IICA を設定する場合は、Stacks タブの"SCLA Pin"と"SDAA Pin"をピン番号のみ設定してください。

設定項目	設定値	説明
Common		
Parameter	Default (BSP)	パラメータチェック処理をコードに含めるか選択できます。
Checking	Enabled	Disabled の場合 コードから省略します。
Ũ	Disabled	Enabled の場合、コードに今めます。
10 bit alaya	Enchlad	
addressing	Enabled	10-bit addressing をサホートするか設定します。
addressing	Disabled	COMMS_120 により上書さされるため、設定不要です。
Module g_iica_ma	aster0 IICA Master (r_lica_ma	aster)
Name	g_iica_master0	ーモジュール名を設定します。
Rate	Standard	ビットレートを設定します。
	Fast-mode	IICAx の電気的特性のため、Standard に設定してください。
	Fast-mode plus	(FS3000 の場合は、Standard もしくは Fast-mode の設定が可能で
		す。)
Custom Rate	0	カスタムビットレートを設定します。
(bps)		0以外の場合に設定が有効です。Rateの設定範囲内の低ビットレー
		トを設定したい場合に利用してください。
Signal Rising	0	SCLの立ち上がり時間を設定します。使用するボードに合わせて設
Times (us)	-	定してください。
Signal Falling	0	SCLの立ち下がり時間を設定します。使用するボードに合わせて設
Times (us)	Ũ	定してください。
Duty Cycle (%)	53	として (たとし)
Duty Cycle (76)	55 Enabled	
Digital Filler	Dischlad	テンダルノイルダ機能の使用を設定しまり。
Address Mode		接続するテハイスのスレーファトレスモートを設定します。
	10-bit	COMMS_12C により上書きされるため、設定不要です。
Slave Address	0x00	接続するデバイスのスレーブアドレスを設定します。
		COMMS_I2C により上書きされるため、設定不要です。
Communication	Enabled	通信予約機能の許可を設定します。
reservation	Disabled	
Callback	rm_comms_i2c_callback	ユーザコールバック関数名を設定します。
		COMMS_I2C により上書きされるため、設定不要です。
IICA0	Priority 0 (highest)	I2C 割り込みの割り込み優先レベルを設定します。
communication	Priority 1	
interrupt priority	Priority 2	
	Priority 3	
SCLA Pin	Pxxx	ドライバが使用する端子番号を設定します。
SDAA Pin	Pxxx	Pins タブでの設定は不要です。
		a jica master() IICA Master (r. jica master)
		Settings Property Value
		Module g_iica_master0 IICA Master (r_iica_master)
		Name g_iica_master0 Rate a Standard
		Signal Rising Time (us) 0
		Signal Falling Time (us) 0 Duty Cycle (%) 53
		Digital Filter Disabled
		Address Mode 7-Bit
		Communication reservation Disabled
		Callback 🔐 m_comms_i2c_callback
		SCLA Pin P100
		SDAA Pin P101

表 5-9 RA r_iica_master 設定一覧

5.3.2 RX ファミリ

Smart Configurator の Component タブで"**r_riic_rx**"もしくは"**r_sci_iic_rx**"コンポーネントを選択することにより、Configure 領域に設定可能な項目が表示されます。

(1) r_riic_rx

衣 5-10 KX [IIIC IX 設正一頁	表	5-10	RX r	riic	rx	設定-	-覧
--------------------------	---	------	------	------	----	-----	----

設定項目	設定値	説明
Common		
Set parameter checking	System Default	パラメータチェック処理をコードに含めるか選択できます。
enable	Not	Not の場合、コードから省略します。
	Include	Include の場合、コードに含めます。
MCU supported channels	Not supported	該当チャネルに関する処理をコードに含めるか選択できま
for CH{x}	Supported	す。
(x = 0 - 2)		Not supported の場合、コードから省略します。
		Supported の場合、コードに含めます。
CH{x} RIIC bps(kbps)	400	ビットレートを設定できます。
(x = 0 - 2)		FS3000 の場合は、400kbps 以下を設定してください。同一
		バス上の他デバイスが接続されている場合、そのデバイスの
		設定可能な転送レートも考慮して設定してください。
Digital filter for CH{x}	Not	指定したチャネルのノイズフィルタの段数を選択できます。
(x = 0 - 2)	One IIC phi	Notの場合、ノイズフィルタは無効となります。
	Two IIC phi	
	Three IIC phi	
	Four IIC phi	
Setting port setting	Not include port setting	ポートを SCL, SDA 端子として使用するための設定処理を
processing	Include port setting	コードに含めるかを選択します。
		Not include port setting の場合、コードから省略します。
		Include port setting の場合、コードに含めます。
Master arbitration lost	Unused	指定したチャネルのマスタアービトレーションロスト検出機
detection function for CH{x} (x = 0 - 2)	Used	能の有効/無効を選択できます。
		マルチマスタで使用する場合は、"Used"にしてください。
		Unused の場合、無効にします。
		Used の場合、有効にします。
Address {y} format for	Not	スレーブアドレスのフォーマットを 7-bit/10-bit から選択で
CH{x}	7 bit address format	きます。
(x = 0 - 2, y = 0 - 2)	10 bit address format	FS3000 の場合は、7-bit address format に設定してくださ
		い。同一バス上に異なるアドレスフォーマットのデバイスを
		接続しないでください。
Slave Address {y} for	0x0025	指定したデバイスのスレーブアドレスを設定します。
CH{x}		COMMS_I2Cにより上書きされるため、設定不要です。
(x = 0 - 2, y = 0 - 2)		
General call address for	Unused	指定したチャネル のゼネラルコールアドレスの有効/無効が
	Used	選択できます。
(x = 0 - 2)		Unusedの場合、無効にします。
		Used の場合、有効にします。
CH{x} RXI INT Priority	Level 1	指定したチャネルの受信データフル割り込み(RXI)の優先レ
	Level 2	ベルを選択できます。
(x = 0 - 2)		
	Level 14	
	Level 15 (highest)	
CH{x} TXI INT Priority	Level 1	指定したチャネルの送信データエンプティ割り込み(TXI)の
(x = 0 = 2)	Level 2	懓先レヘルを選択できます。
(^ - 0 - 2)		
	Level 14	
	Level 15 (highest)	

RA ファミリ、RX ファミリ、RL78 ファミリ

CH{x} EEI INT Priority	Level 1	指定したチャネルの通信エラー/イベント発生割り込み(EEI)
Level	Level 2	の優先レベルを選択できます。
(x = 0 - 2)		
	Level 14	
	Level 15 (highest)	
CH{x} TEI INT Priority	Level 1	指定したチャネルの送信終了割り込み(TEI)の優先レベルを
Level	Level 2	選択できます。
(x = 0 - 2)		
	Level 14	
	Level 15 (highest)	
Timeout function for CH{x}	Unused	指定したチャネルのタイムアウト検出機能を有効にできま
(x = 0 - 2)	Used	す。
		Unused の場合、タイムアウト検出機能無効
		Used の場合、タイムアウト検出機能有効
Timeout detection time for	Long mode	指定したチャネルのタイムアウト検出時間を選択できます。
CH{x}	Short mode	Long mode の場合、ロングモードを選択。
(x = 0 - 2)		Short mode の場合、ショートモードを選択。
Count up during low period	Unused	指定したチャネルのタイムアウト検出機能有効時、SCL ラ
of timeout detection for	Used	インが Low 期間中にタイムアウト検出機能の内部カウンタ
CH{X}		のカウントアップを有効にできます。
(x = 0 - 2)		Unused の場合、カウントアップ禁止。
		Used の場合、カウントアップ有効。
Count up during high	Unused	指定したチャネルのタイムアウト検出機能有効時、SCL ラ
period of timeout detection	Used	インが High 期間中にタイムアウト検出機能の内部カウンタ
for CH{x}		のカウントアップを有効にできます。
(x = 0 - 2)		Unused の場合、カウントアップ禁止。
		Used の場合、カウントアップ有効。
Set Counter of checking	1000	API 関数のバスチェック処理時に、ソフトウェアによりタイ
bus busy		ムアウトカウンタ(バス確認回数)を設定できます。
Resources		
SCLx 端子	Checked	使用する端子を設定します。
	Unchecked	使用する端子を Checked に設定してください。
SDAx 端子	Checked	
	Unchecked	

(2) r_sci_iic_rx

表	5-11	RX r_	_sci_	_iic_	_rx	設定-	-覧
---	------	-------	-------	-------	-----	-----	----

設定項目	設定値	説明				
Configurations						
Set parameter checking	System Default	パラメータチェック処理をコードに含めるか選択できます。				
enable	Not	Not の場合、コードから省略します。				
	Include	Include の場合、コードに含めます。				
MCU supported	Not supported	該当チャネル関する処理をコードに含めるか選択できます。				
channels for CH{x}	Supported	Not supported の場合、コードから省略します。				
(x = 0 - 12)		Supported の場合、コードに含めます。				
SCI IIC bitrate (bps) for	384000	ビットレートを設定してください。				
CH{x}		FS3000 の場合は、384000bps 以下を設定してください。同一				
(x = 0 - 12)		バス上の他デバイスが接続されている場合、そのデバイスの設				
		定可能な転送レートも考慮して設定してください。				
Interrupt Priority for	Level 1	コンディション割り込み、受信割り込み、送信空割り込み、送				
CH{x}	Level 2	信完了割り込みの優先レベルを設定してください。				
(x = 0 - 12)						
	Level 14					
	Level 15 (highest)					
Digital noise filter	Disable	SSCL、SSDA 入力信号のノイズ除去機能を使用するか選択でき				
(NFEN bit) for CH{x}	Enable	ます。				
(x = 0 - 12)		Disable の場合、無効にします。				
		Enable の場合、有効にします。				
Noise Filter Setting	The clock divided by 1	デジタルノイズフィルタのサンプリングクロックを選択しま				
Register (NFCS bit) for	The clock divided by 2	す。				
(x = 0 - 12)	The clock divided by 4					
(x = 0 - 12)	The clock divided by 8					
I2C Mode Register 1	18	SSCL 端子出力の立ち下がりに対する SSDA 端子出力の遅延を				
$(\Pi \bigcup \{x\}$						
(x = 0 - 12)		1 ~ 31 の範囲で設定してください。				
Software bus busy	1000	バスビジー判定のカウント数を設定します。				
check counter		簡易 I2C の API 関数のバスチェック処理時の、タイムアウトカ				
		ウンタ(バス確認回数)を設定できます。				
Port Setting Processing	Not include port setting	ポートを SSCL、SSDA 端子として使用するための設定処理を				
	Include port setting	コードに含めるか選択できます。				
		Not include port setting の場合、コードから省略します。				
		Include port setting の場合、コードに含めます。				
Resources						
SSCLx 端子	Checked	使用する端子を設定します。				
	Unchecked	使用する端子を Checked に設定してください。				
SSDAx 端子	Checked					
	Unchecked					

5.3.3 RL78 ファミリ

Smart Configurator の Component タブで IIC 通信(マスタモード)からリソースに"IICAx"を選択すること により、Configure 領域に設定可能な項目が表示されます。

Simplified I2C using Serial Array Unit (SAU)は、クロック・ストレッチ機能を未サポートのため使用できません。

(1) IICAx

表 5-12 RL78 IICAx 設定一覧

設定項目	設定値	説明				
Configurations	- -					
クロック・モード設定	fCLK	カウント・クロックを設定します。				
	fCLK/2					
アドレス	16	自局アドレスを設定します。				
動作モード	標準	動作モードを設定します。				
	ファスト・モード	FS3000 の場合は、標準もしくはファスト・モードの設				
	ファスト・モード・プラス	定が可能です。同ーバス上の他デバイスが接続されてい る場合、そのデバイスの設定可能な転送レートも考慮し て設定してください。				
ディジタル・フィルタ・	Checked	デジタルフィルタ機能の使用を設定します。				
オン	Unchecked					
転送クロック (fSCL)	100000	ビットレートを設定します。				
		IICAx の電気的特性のため、100000bps 以下に設定して ください。				
tRとtFの手動入力設定	Checked	SDAAn, SCLAn 信号の立ち上がり時間、立ち下がり時間				
	Unchecked	を手動で設定します。				
tR	0	SDAAn, SCLAn 信号の立ち上がり時間を設定します。				
tF	0	SDAAn, SCLAn 信号の立ち下がり時間を設定します。				
通信完了割り込み優先順位	レベル 0(高優先順位)	通信完了割り込みの割り込み優先順位を設定します。				
(INTIICAx)	レベル1					
	レベル2					
	レベル 3(低優先順位)					
マスタ送信完了	Checked	マスタ送信完了によるコールバック機能を設定します。				
	Unchecked					
マスタ受信完了	Checked	マスタ受信完了によるコールバック機能を設定します。				
	Unchecked					
マスタ・エラー	Checked	通信エラーによるコールバック機能を設定します。				
	Unchecked					
マスタ送信/受信完了コー	Checked	コールバック時のストップ・コンディション生成を設定				
ルバック時に	Unchecked	します。Unchecked に設定してください。				
ストップ・コンディション を生成						

RENESAS

6. デバイス変更ガイド

サンプルプロジェクトを異なるデバイスで動作させるには、以下の手順に従ってください。 移行元デバイスのサンプルプロジェクトは、事前に Workspace にインポートしてください。

6.1 RA サンプルプロジェクト

サンプルプロジェクトを変更する場合の手順は以下の通りです。

以下のボード変更例での変更手順を説明します。なお、Pmod Type 2A/3A コネクタ使用時は Interposer Board が必要です。

 サンプルプロジェクト "FS3000_RA2E1_NonOS": PMOD1 (Type 2A/3A: SCI0)
 → EK-RA6M4 ボードの PMOD1 (Option Type 6A: IIC1)もしくは PMOD2 (Type 2A: SCI0)

6.1.1 サンプルプロジェクトのインポート

1. メニューから、インポートを選択します。

表示されたインポートウィンドウで、"Rename & Import Existing C/C++ Project into Workspace"を選択し、[Next]ボタンを押下します。

2. [Browse]ボタンを押下し、フォルダの選択ウィンドウを表示します。

インポート済みのサンプルプロジェクトから、移行元デバイスのプロジェクトのフォルダを選択し、[フォ ルダの選択]ボタンを押下します。

Select Folder			×
$\leftarrow \rightarrow \checkmark \uparrow$. Windows (C:)	> workspace > e2_studio > FS3000 ~	ප Search FS3000	م
Organize 🔻 New folder			III - ?
workspace	^ Name	Date modified	Туре
e2_studio	.metadata	10/17/2024 4:30 PM	File folder
FS3000	FS3000_RA0E1_NonOS	10/8/2024 4:51 PM	File folder
.metadata	FS3000_RA2E1_FreeRTOS	10/18/2024 9:30 AM	File folder
FS3000_RA0E1_NonOS	FS3000_RA2E1_NonOS	10/18/2024 9:30 AM	File folder
FS3000_RA2E1_FreeRTOS			
FS3000_RA2E1_NonOS			
.settings			
script			
src	~ <		>
Folder: FS3000_R	A2E1_NonOS		
		Select Folder	Cancel

3. プロジェクト名の入力および、移行元デバイスのプロジェクトを選択し、[Finish]ボタンを押下します。

🖸 Import				×
Rename & Impo	rt Project			5
Select a directory	to search for existing Eclipse projects.			
Project name: FS	S3000_RA6M4_NonOS			
🗹 Use default lo	ocation			
Location:	C:\workspace\e2_studio\FS3000\FS3000_RA6M4_N	Ni I	Browse	
	Create Directory for Project			
Choose file syster	m: default 🖂			
Import from:				
Select root dir	ectory: C:\workspace\e2_studio\FS3000\FS3000_f ~		Browse	
◯ Select archive	file: 🗸		Browse	
Proiects:				
FS3000 RA2E1	NonOS (C:\workspace\e2 studio\FS3000\FS3000 RA	2E1 N	lonOS)	
			,	
Options				
Keep build co	nfiguration output folders			
?	< Back Next > Finish		Cance	I

6.1.2 FSP Configurator の設定変更

プロジェクトツリーの Configuratorn.xml をダブルクリックし、FSP Configurator を開きます。

(1) BSP

BSP タブで Board および Device を変更します。

ルネサス製ボードに変更する場合は、Boardの設定のみ変更してください。

ルネサス製以外のボードに変更する場合は、Board を"Custom User Board (Any Device)"に変更後、 Device を使用するデバイスに変更してください。

ard Suppo	ort Package Configuration		Generate Project Conte
			🔜 Restore Defau
evice Selectio	on		
FSP version:	X.X.X	Board Details	
Board:	EK-RA2E1 🗸	Visit https://www.renesas.com/ra/ek-ra2e1 to get kit user's mar	ual. quick start quide.
Device: E	EK-RA2E1	errata, design package, example projects, etc.	
Core: E	EK-RA2L1 EK-RA4E2	✓	
RTOS:	EK-RA4M1 EK-RA4M2	~	
E	EK-RA4M3		
E	EK-RA6E2		
E	EK-RA6M1		
E	EK-RA6M2		
	EK-RAGIVIS EK-RAGM3G		
	EK-RA6M4		
Ē	EK-RA6M5		
E	EK-RA8D1		
E	EK-RA8M1		
F	FPB-RA0E1		
F	FPB-RA2E1		
F	FPB-RA2E2		
F	FPB-RA2E3		
F	FPB-RA4E1		
	FPB-RA4E2		

(2) Clocks

Clocks タブで、クロック設定を変更します。

Board を"Custom User Board (Any Device)"に変更した場合は、使用するボードに合わせてクロック設定 を変更してください。

Clocks Configuration		Generate Project Content
		🔣 Restore Defaults
XTAL 24MHz	Clock Src: PLL V V ICLK Div /1	✓ → ICLK 200MHz
	→ PCLKA Div /2	→ PCLKA 100MHz
HOCO 20MHz V	↔ PCLKB Div /4	✓ → PCLKB 50MHz
LOCO 32768Hz	→ PCLKC Div /4	✓ → PCLKC 50MHz
MOCO 8MHz	→ PCLKD Div /2	✓ → PCLKD 100MHz
SUBCLK 32768Hz	BCLK Div /2	✓ → BCLK 100MHz
>> PLL Src: XTAL >>	EBCLK Div /2	✓ → EBCLK 50MHz
PLL Div /3 ~	FCLK Div /4	✓ → FCLK 50MHz
PLL2 Div /2 v	ightarrow CLKOUT Disabled $ ightarrow ightarrow$ CLKOUT Div /1	✓ → CLKOUT 0Hz
PLL2 Mul ×20.0 V	>> UCLK Disabled >> UCLK Div /5	✓ → UCLK 0Hz
PLL2 0Hz	→ OCTASPICLK Disabled ~ → OCTASPICLK Div /1	✓ → OCTASPICLK 0Hz
Summary BSP Clocks 😣 Pins Interrupts Event Links S	tacks Components	

Board をルネサス製ボードに変更した場合は、自動的に設定が変更されます。

- (3) Pins
- (a) ボード変更

Pins タブで、使用するボードに合わせて、端子設定を変更します。

ルネサス製ボードを使用する場合は、Select Pin Configuration を"RA2E1 EK"から使用するボードに変更 することで、自動的に割り当てが行われます。

Pin Configuration			Generate Project Content
Select Pin Configuration		📑 Export to C	CSV file 🖺 Configure Pin Driver Warnings
RA2E1 EK	<u>Manage configurations</u>	🗹 Gene	erate data: g_bsp_pin_cfg
Pin Selection $\blacksquare \blacksquare \blacksquare \downarrow^a_{\mathbf{Z}}$	Pin Configuration		😲 Cycle Pin Group
Type filter text	Name	Value	Link
✓ ✓ Ports ∧			
> 🗸 b0			
> 🗸 P1			
> V P2			
> V P3			
> V P4			
> P6			
> P7			
> P8			
> 🗸 Other Pins			
V 🐼 Peripherals			
> 🗸 Analog:ADC			
> 🗸 Analog:ANALOG			
> Analog:DAC			
> Connectivity:CAN			
> Connectivity:ETHERC			
> Connectivity:IIC			
Connectivity.Sci	<		>
Connectivity:SSI			
> ✓ Connectivity:USB			
> Input:CTSU			
> 🗸 Input:ICU			
> Monitoring:CAC 🗸			
Pin Function Pin Number			
Summary BSP Clocks (9 Pins Interrupts	Event Links Stacks Components		

Select Pin Configuration のドロップダウンリストに変更したいボードが表示されない場合は、[Manage Configuration]をクリックし、表示された Manage Pin Configuration ウィンドウで、使用したいボードを選択してください。

Manage Pin Configurations		×
Multiple Pin Configuration Management		1
Modify pin configuration list or import/export external file		
RA2E1 EK (Current)	Add	
R7FA6M4AF3CFB.pincfg	Remove	:
	Rename	
	Duplicat	e
	Merge to	
	Import.	
	Export	
	OK	
	UK	

(b) I2C I/F 端子変更

EK-RA6M4 ボードの場合、(a)での割り当てでは Pmod Type 2A に対応した SPI 通信の端子設定が適用されます。

このサンプルソフトウェアでは、Pmod Type 6A を使用するため、Pmod Type 6A に対応した I2C 通信 の端子設定への変更が必要です。

EK-RA6M4 ボードでは、PMOD1 に IIC1、PMOD2 に SCI0 が割り当てられています。

PMOD1 (Option Type 6A)を使用する場合は P511 と P512 が、PMOD2 を使用する場合は P410 と P411 が I2C 通信を行う端子となるため、Select Pin Configuration の自動割り当て後、Pin Configuration にて再設 定を行います。

Pin Configuration					Generate Project Content
Select Pin Configuration		📑 Export	to CSV file	Configure	Pin Driver Warnings
RA6M4 EK	✓ Manage configurations		Generate data:	g_bsp_pin	n_cfg_6m4
Pin Selection \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare	Pin Configuration				😲 Cycle Pin Group
Type filter text	Name	Value	Lock	Link	
> ✓ Other Pins ^ ✓ ✓ Peripherals	Operation Mode	Simple I2C			
> 🗸 Analog:ADC > 🗸 Analog:ANALOG	✓ Input/Output TXD0	None			
Analog:DAC Connectivity:CAN	RXD0 SCK0	None None		\rightarrow	
Connectivity:ETHERC Connectivity:IIC	CTS0 SDA0	None P411 			
✓ ✓ Connectivity:SCI	SCL0 CTSRTS0	✓ P410 None			
SCI1 SCI2 SCI3					
SCI4 SCI5	<				>
 ✓ SCI7 SCI8 SCI9 	Module name: SCI0 Usage: When using Sim When switching	ple I2C mode, ensure port between I2C and other m	pins output ty odes, first disab	pe is n-ch o Ile.	pen drain.
Pin Function Pin Number Summary BSP Clocks Pins Interrupts Ev	rent Links Stacks Components				

デバイス変更により、"Genarate data"が無効状態になります。次ページに有効化方法を説明します。

Pin Configuration				Generate Project Content
Select Pin Configuration		Į	Export to CSV file	E Configure Pin Driver Warnings
RA6M4 EK	✓ Manage configurations		Generate dat	a:
Pin Selection 🗄 🕀 🛱	Pin Configuration			😲 Cycle Pin Group
Type filter text	Name	Value	Link	

端子設定の生成を有効にするには、[Generate data]にチェックを入れ、テキストボックスに任意の名称を 入力してください。

入力された名称は設定した端子構成に紐付けられるため、他の端子構成と重複しないよう、固有の名称と してください。

例の場合、"g_bsp_pin_cfg_6m4"です。

Pin Configuration				Generate Project Content
Select Pin Configuration		📑 Exp	port to CSV file 🖺 Configure	Pin Driver Warnings
RA6M4 EK	Manage configurations	[Generate data: g_bsp_pin	_cfg_6m4
Pin Selection $\blacksquare \boxdot \Box \downarrow^a_Z$	Pin Configuration			😲 Cycle Pin Group
Type filter text	Name	Value	Link	
 Ports Ports P0 P1 P2 P3 P4 P5 P6 P7 P8 Other Pins Peripherals Analog:ADC Analog:ADAC Analog:DAC Connectivity:CAN Connectivity:CAN 	٢			>
 Connectivity: ETHERC Connectivity:IIC Connectivity:SCI Connectivity:SPI 				
Pin Function Pin Number ummary BSP Clocks Pins Interrupts Even	nt Links Stacks Components			

(4) Stacks

Stacks タブで、各コンポーネント設定を変更します。

(a) COMMS_I2C と I2C ドライバの設定変更

使用するボードに合わせて、COMMS_I2CとI2Cドライバの設定を変更してください。

I2C I/F の端子を使用する場合は、不要な stack を削除し、新たに使用する stack を追加してください。

表 6-1 EK-RA6M4 における I2C I/F とチャネルの設定

EK-RA6M4	12C I/F	g_comms_i2c_bus0 I2C Shared Bus (rm_comms_i2c)	g_i2c_master0 I2C Master
PMOD1 Option Type 6A	IIC1	Channel: 1	Pins を確認
PMOD2 Type 2A	SCI0	Channel: 0	Pins を確認

g_comm	s_i2c_bus0 l2C Shared Bus (rm_comms_i2c)	
Settings	Property	Value
API Info	Parameter Checking	Default (BSP)
	 Module g_comms_i2c_bus0 I2C Shared Bus (rm_comms_i2c) 	
	Name	g_comms_i2c_bus0
	Bus Timeout	0xFFFFFFF
	Semaphore for Blocking (RTOS only)	Use
	Recursive Mutex for Bus (RTOS only)	Use
	Channel	1
	Rate	Standard

_____ IIC1 の場合

D	
Property	Value
✓ Common	
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	🔒 1
Rate	🔒 Standard
Custom Rate (bps)	0
Rise Time (ns)	120
Fall Time (ns)	120
Duty Cycle (%)	50
Slave Address	0x00
Address Mode	7-Bit
Timeout Mode	Short Mode
Timeout during SCL Low	Enabled
Callback	fill rm_comms_i2c_callback
Interrupt Priority Level	Priority 12
✓ Pins	
SDA1	P511
SCL1	P512

_____ IIC1 の場合

(b) 汎用 I/O ポート Driver 設定変更

"g_ioport I/O Port"の Pin Configuration Name に、使用する端子構成の名称を入力してください。

例の場合、"g_bsp_pin_cfg_6m4"です。

Stacks Configuration		Generate Project Content
Threads 🔄 New Thread 🔊 Remove 📄 HAL/Common Stacks	🗐 New Stack 🤉	Extend Stack > 😨 Remove
Gioport I/O Press P	g_fs3000_sensor0 FS3000 Flow Sensor (rm_fs3000)	
+ fs3000_delay Timer, General PWM (r_gpt)	①	1
	g_comms_i2c_device0 I2C Communication Device (rm_comms_i2c)	
	0	
< >>	g_comms_i2c_bus0 I2C Shared Bus (rm_comms_i2c)	
Objects 🐑 New Object > 😰 Remove	0	
	g_i2c0 I2C Master (r_sci_i2c)	
	1	~
Summary BSP Clocks Pins Interrupts Event Links Stacks Components		

🔲 Properti	es ×	📑 🖇 🗖 E	3
g_ioport	I/O Port (r_ioport)		
Settings	Property	Value	`
API Info	✓ Common		L
	Parameter Checking	Default (BSP)	
	 Module g_ioport I/O Port (r_ioport) 		
	Name	g_ioport	
	1st Port ELC Trigger Source	Disabled	
	2nd Port ELC Trigger Source	Disabled	
	3rd Port ELC Trigger Source	Disabled	L
	4th Port ELC Trigger Source	Disabled	L
	Pin Configuration Name	g_bsp_pin_cfg_6m4	
	✓ Pins		L
	TCK	P300	L
	TDI	P110	L
	TDO	P109	
	TMS	P108	L
	SWCLK	<unavailable></unavailable>	
	SWDIO	<unavailable></unavailable>	1
	TRACESWO	<unavailable></unavailable>	
	TCLK	<unavailable></unavailable>	
	TDATA0	<unavailable></unavailable>	
	TDATA1	<unavailable></unavailable>	
	TDATAD	summer and the second s	1
	<	>	

他 stack でエラーが表示されている場合は、表示されたエラーに従って指定の項目を変更してください。

(5) コード生成

修正完了後、[Generate Project Content]を押下して、ファイルを生成します。

プロジェクトをビルドします。

メニューから[Debug Configurations]を選択し、使用するボードに接続するエミュレータに合わせて、 Debugger の設定を変更してください。

6.1.3 ツールチェイン設定変更

GCC ARM Embedded ツールチェイン以外のツールチェインを使用する場合は、本プロジェクトから RA_FS3000.c (Non-OS)または、fs3000_sensor_thread_entry.c 及び sensor_thread_common.c、 sensor_thread_common.h (FreeRTOS)をコピーしてプロジェクトを作成してください。

6.2 RX サンプルプロジェクト

サンプルプロジェクトを変更する場合の手順は以下の通りです。

以下のボード変更例での変更手順を説明します。なお、Pmod Type 2A/3A コネクタ使用時は Interposer Board が必要です。

 サンプルプロジェクト "FS3000_RX65N_NonOS": PMOD1 (Type2A: SCI2)
 → RSK-RX231 ボードの PMOD1 (Type 2A: SCI8)

6.2.1 サンプルプロジェクトのインポート

1. メニューから、インポートを選択します。

表示されたインポートウィンドウで、"Rename & Import Existing C/C++ Project into Workspace"を選択 し、[Next]ボタンを押下します。

Import -		×
Select	\ \	
Rename and Import and Existing C/C++ Project into the workspace	2	-
Select an import wizard:		
type filter text		
V 🗁 General		^
🚇 Archive File		
😡 CMSIS Pack		
CMSIS Pack		
😭 Existing Projects into Workspace		
🚞 File System		
GNUARM-NONE/RZ(DS-5) project conversion to GCC ARM Emb	edded	
Preferences		
Projects from Folder or Archive		
😭 Rename & Import Existing C/C++ Project into Workspace		
Renesas CCRX project conversion to Renesas GCC RX		
Renesas CS+ Project for CA78K0R/CA78K0		
Renesas CS+ Project for CC-RX and CC-RL		
Renesas GitHub FreeRTOS (with IoT libraries) Project		~
? < <u>B</u> ack <u>N</u> ext > <u>F</u> inish	Canc	el

2. [Browse]ボタンを押下し、フォルダの選択ウィンドウを表示します。

インポート済みのサンプルプロジェクトから、移行元デバイスのプロジェクトのフォルダを選択し、[フォ ルダの選択]ボタンを押下します。

3. プロジェクト名の入力および、移行元デバイスのプロジェクトを選択し、[Finish]ボタンを押下します。

📴 Import									0		×
Rename & Imp Select a directory	ort Project	or existin	ng Eclip	ose proj	ects.						7
,			5							<u> </u>	-4
<u>P</u> roject name:	FS3000_RX23	31_NonO	DS								
✓ Use <u>d</u> efault	location										
Location:	C:¥User	s¥a50905	534¥e2	_studio	¥works	pace	€F\$30	00_F	Bro	owse	
	Creat	e Directo	ory for	Project							
Choose file syste	em: default	\sim									
Import from:	_										
Select root d	lirectory: C	:¥Users¥	⁽ xxxxx)	xxx¥e2_	studio	fwork	space	~	Bro	wse	
○ Select <u>a</u> rchiv	e file:							~	Bro	wse	
Projects:											
FS3000_RX65	N_NonOS ((C:¥Users ³	¥xxxxx	xxxx¥e2	_studio	¥worl	kspac	e¥ FS3	000_R)	(65N_	Non(
<											>
Options											
Keep build c	onfiguration	output	folder	s							
?	< <u>B</u>	ack		<u>N</u> ext >		E	inish		(Cancel	1

6.2.2 デバイスの変更

プロジェクトツリーでインポートしたプロジェクトを選択し、右クリックでコンテキストメニューを表示します。表示されたメニューから、"Change Device"を選択します。

2. Change Device ウィンドウで、変更したいボードもしくは、デバイスを選択し、[Next]ボタンを押下します。

Refactoring		—		×
Change Device Select the new o	e device for FS3000_RX231_NonOS			
Current Device: Current Board: E	R5F565NEDxFB InvisionRX65N			
Target Board:	RSKRX231			\sim
Target Device:	R5F52318AxFP	<u>Download add</u>	ditional bos Jnlock Dev	ices
?	< <u>B</u> ack <u>N</u> ext >	<u>F</u> inish	Cance	4

3. Warning が表示された場合、内容を確認し問題無ければ、[Next]を押下します。

Image: Sectoring Image:			Х
Change Device Review the information provided in the list below. Click 'Nex view the next item or 'Finish'.	t >' to		
Found problems		5	60
This change cannot be undone. Please make sure you base	ckup thi	s project	before
<			>
? < <u>B</u> ack <u>Next</u> > <u>F</u> inish		Cance	4

4. 変更内容が表示されるため、[Finish]ボタンを押下して、変更を実行します。

6.2.3 Smart Configurator 設定の変更

プロジェクトツリーで、デバイス変更したインポートしたプロジェクトの.scfg ファイルをダブルクリックし、Smart Configurator を表示します。

(1) Board

Board タブで、変更したボード、デバイスに変更されていることを確認します。

戀 FS300x_F	X_NonOS.scfg ⊠
Device s	election
Device sel	ection
Board: Device:	RSKRX231 (1.00) V R5F52318AxFP Download more boards

(2) Clocks

Clocks タブで、使用するボードに合わせてクロックを設定します。

🕸 FS3000_RX_NonOS.scfg % 🔍 🗖	- 🖶 * F53000_RX_NonOS.scfg 🛙 📃 🗖
Clocks configuration	Clocks configuration
VSC 13 (Planet excel 22) VSC 13 (Planet excel 22) VSC 13 (Planet excel 22) VSC 14 (Planet exc	VCC 22 01 Direct value 32 PC 22 01 Direct value 32 01 Direct 32
HOCD close CACHAL Registry - House KCD contradies after source -	CONCIL
Language data	COLOR C

(3) Components

Components タブで、使用するボードに合わせて、各 component の設定を変更します。

(a) I2C ドライバ設定変更

RSK-RX231 では、PMOD1 (Type 2A)に SCI8 が割り当てられているため、r_sci_iic_rx の MCU supported channels for CH2 を"Not supported"に、MCU supported channels for CH8 を"Supported"に変更します。

Resources の SCI8, SSCL8 Pin, SSDA8 pin をチェックします。

onends 177 🛏 🕂 🛒 🕇	Configure	(i)	Componente 🗟 🗖 🗖 🗮	Configure	
* *			components 12 (a) (a) (b)	comgare	
filter text	Configurations	value	huma fillers have	Property	Value
	# Set parameter checking enable	Include	type litter text	v lin Resources	
startup	# MCU supported channels for CH0	Not supported	V 🗁 Startup		
Generic	# MCU supported channels for CH1	Not supported	V 🗁 Generic	B SSCLODIE	
Cosp # MCU supported channels for CH2 Not supported Communications # MCU supported channels for CH3 Not supported V	Not supported	💣 r_bsp	SSCLOPHI SSCLOPHI		
	V 🗁 Drivers				
Communications	# MCU supported channels for CH4	Not supported	V 🗁 Communications	SCI 1 Dia	
r_nic_rx	# MCU supported channels for CH5	Not supported	🐮 r_niic_nx	SOLUTION SOLUTION	
r_sci_iic_rx	# MCU supported channels for CH6	Not supported	😵 r_sci_iic_nx		
Middleware	# MCU supported channels for CH7	Not supported	✓	Sector Dia	
Centernic Comparison Co	Supported	V 🗁 Generic	SSCED Fill		
	Not supported	r_comms_i2c_rx		17	
	# MCU supported channels for CH10	Not supported	🖌 r_hs300x_rx	SSCI 6 Pin	I liked
	# MCU supported channels for CH11	Not supported		SSDA6 Pin	V Used
	# MCU supported channels for CH12	Not supported		× SCI8	
	# SCI IIC bitrate (bps) for CH0	384000		SSCL8 Pin	V Used
	# SCI IIC bitrate (bps) for CH1	384000		SSDA8 Pin	Used
	# SCI IIC bitrate (bps) for CH2	384000		✓ 10 SCI9	
	# SCI IIC bitrate (bps) for CH3	384000		SSCL9 Pin	
	# SCI IIC bitrate (bps) for CH4	384000		SSDA9 Pin	
	# SCI IIC bitrate (bps) for CH5	384000		✓ III SCI12	
	# SCI IIC bitrate (bps) for CH6	384000		SSCL12 Pin	
	# SCI IIC bitrate (bps) for CH7	384000		SSDA12 Pin	
	# SCLIIC hitrate (hos) for CH8	384000 ¥			

(b) COMMS_I2C 設定変更

I2C ドライバおよびチャネルを変更した場合、設定を変更する必要があります。 SCI8 を使用する場合、r_comms_i2c_rx の Channel No. for I2C Shared Bus0 を"8"に変更します。

(4) Pins

(a) I2C I/F 端子変更

Pin function で、SCI8 端子に端子機能が割り当てられていることを確認します。

n configuration			Generate Co	de Generate Rep
lardware Resource 🛛 🕀 📮 🖧	Pin Function	n	3	🖬 👪 🔤
Type filter text	type filter	text (* = any str	ing, ? = any character)	II
TMR2 TMR3 TMR3 TMR3 TMR3 TMR3 TMR3 TMR3 TMR4 TMR3 TMR4 Scription Scliption Scliption		Function CTS## RTS# RTS# RTS# RTS# RTS# SKR8 SMIS08 S	Assignment Not assigned Control Car/MTOC3A/	 Number Not assigned
📥 Analog power supply 🗊 I/O ports 🗸 🗸				
< >	<			

RSK-RX231のボード情報は、Pmod Type 2A (Extend SPI)で使用するように割り当てられているため、 I2C で使用する場合は、Warning が表示されますが問題ありません。

また、センサボードを接続するには、Pmod Type 2A を Pmod Type 6A に変換する Interposer Board が必要となります。

(5) コード生成実行

修正完了後、[Generate Code]アイコンを押下して、コード生成を行います。

			C	-
Verview information			Generate Code enerat	e Repoi
- General Information				0
This editor allows you to modify the settings stored in config	guration file (.sc	:fg)		
Board				
Allow board and device selection				
Clocks			Application under development	
Allow clock configuration			+ Component	ate
			Middleware	
Components			Device	
Allow software component selection and configuration			driver	
			- Pins	
Pins				
Pins Allow general pin configuration and pin configuration for se	elected software	e component		
Pins Allow general pin configuration and pin configuration for se	elected software	e component		
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configure	elected software	e component		
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interrupt configuration and interrupt configur	elected software ration for select	e component ted software component		
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configur. Click here to not more information on Lice's Manual Deleas	elected software ration for select	e component ted software component		
Pins Allow general pin configuration and pin configuration for se Intercupt Allow general intercupt configuration and intercupt configur Click here to get more information on <u>User's Manual, Releas</u>	elected software ration for select	e component ted software component ation Notes Tool News		
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interrupt configuration and interrupt configur Click here to get more information on <u>User's Manual Releas</u>	elected software ration for select	e component ted software component ation Notes Tool News		
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interupt configuration and interupt configur Click here to get more information on <u>User's Manual Releas</u> • Current Configuration	elected software ration for select <u>ee Note Applica</u>	e component ted software component ation Notes Tool News		
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configur Click here to get more information on <u>User's Manual Releas</u> Current Configuration Selected board/device: RSF52318AxFP (ROM size: S12 Kbytes	elected software ration for select ie Note Applica s , RAM size: 64	e component ted software component <u>ation Notes Tool News</u> Kbytes , Pin count: 100)		
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interrupt configuration and interrupt configur Click here to get more information on <u>User's Manual Releas</u> • Current Configuration Selected board/device: R5F52318AxFP (ROM size: S12 Kbytes Generated Jocation (PROJECT_JOC'P): <u>servitance.gen</u>	elected software ation for select ie Note Applica s , RAM size: 64	e component ted software component ation Notes Tool News I Kbytes , Pin count: 100)	Edfu	
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interrupt configuration and interrupt configur- Click here to get more information on <u>User's Manual Releas</u> • Current Configuration Selected board/device: RSF2318AxFP (ROM size: S12 Kbyter Generated location (PROJECT_LOCY): <u>srcKsnc.gen</u> Selected components:	elected software ation for select ie Note Applica s , RAM size: 64	e component ted software component ation Notes Tool News Kbytes , Pin count: 100)	: Edit]	
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configuration Click here to get more information on User's Manual Releas Current Configuration Selected board/device: RSF52318AxFP (ROM size: 512 Kbytes Generated location (PROJECT_LOC'9): [srcVame_gen Selected components: Component Compo	elected software ation for select is Note Applica s , RAM size: 64 Version	e component ted software component ation Notes Tool News Kbytes, Pin count: 100	Edr]	
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configuration Click here to get more information on <u>User's Manual Release Current Configuration Selected board/device: R5F52318AxFP (ROM size: 512 Kbytes Generated Location (PROJECT_LOC'9): <u>IsrcVarm.gen Selected components: Component Beaced Support Page Selected Device Selected Component Device Selected Device Sele</u></u>	elected software ation for select te Note Applica s , RAM size: 64 Version 5.66	e component ted software component ation Notes Tool News I Kbytes, Pin count: 100) Configuration r bsofused	: Edit]	
Pins Allow general pin configuration and pin configuration for se Interupt Allow general interrupt configuration and interrupt configuration Click here to get more information on <u>User's Manual Releas</u> Current Configuration Selected board/device: RSF2318AxFP (ROM size: 512 Kbyter Generated location (PROJECT_LOCP); <u>serEsmc_gen Selected component: Component: Component FS000 Senser Middleware (r,fs100_n) FS000 Senser Middleware (r,fs100_n) </u>	elected software ation for select <u>e Note Applica</u> s , RAM size: 64 Version 5.66 1.00	e component ted software component ation Notes Tool News I Kbytes, Pin count: 100 Configuration r, bsp(used)	Edit]	
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configuration Click here to get more information on User's Manual Release • Current Configuration Selected board/device: RSF52318AxFP (ROM size: S12 Kbytes Generated location (PROJECT_LOC%): [srcVame_gen Selected component: Component © Board Support Packages (r_bsp) © IN Communication Driver Interface Middleware (r_com.	elected software ation for select s. RAM size: 64 Version 5.66 1.00	e component ted software component ation Notes Tool News Kbytes, Pin count 100) Configuration r, bsp(used) r, comm, i2c, pi(u	Edit]	
Pins Allow general pin configuration and pin configuration for se Interrupt Allow general interrupt configuration and interrupt configuration Click here to get more information on <u>User's Manual Release current Configuration Generated location (RPO)ECT_LOC%: <u>artVanc_gen</u> Selected board/device: RSF52318AxFP (ROM size: 512 Kbyter Generated location (RPO)ECT_LOC%: <u>artVanc_gen</u> Selected component: Component Component FS3000 Sensor Middleware (r_fs3000_rs) CF33000 Sensor Middleware (r_fs3000_rs) CF310C Milth Matter Lic Driver (Intriace Middleware (r_co) C MIC Multi Matter Lic Driver (Intriace)</u>	elected software ation for select is , RAM size: 64 Version 5.66 1.00 2.46	e component ted software component ation Notes Tool News Kbytes, Pin count 100 Configuration r, brafoursetion r, brafoursetion r, brafoursetion r, comms, Izz, en(u r, ruit, en, ruised)	cdit]	

プロジェクトをビルドします。

メニューから[Debug Configurations]を選択し、使用するボードに接続するエミュレータに合わせて、 Debugger の設定を変更してください。

6.2.4 ツールチェイン設定変更

CC-RX ツールチェイン以外のツールチェインを使用する場合は、本プロジェクトから RX_FS3000.c (Non-OS)または、main.c と fs3000_sensor_thread_entry.c (FreeRTOS)をコピーしてプロジェクトを作成し てください。

6.3 RL78 サンプルプロジェクト

サンプルプロジェクトを変更する場合の手順は以下の通りです。

以下のボード変更例での変更手順を説明します。

- サンプルプロジェクト "FS3000_RL78G23_NonOS": PMOD2 (Type 6A: IICA1)
 → RL78/G22 Fast Prototyping Board の Grove (IICA0)
 - なお、Groveの VDD を 3.3V にするために、J17 を 2-3 に設定してください。
 - また、以下のようにジャンパ線で接続してください。

RL78/G22						
Fast Prototyping Board (Grove)						
Din	Eurotion	IICA I/F,				
ГШ	Function	Power Supply				
#1	SCL	SCLA0				
#2	SDA	SDAA0				
#3	VCC	3.3V				
#4	GND	GND	-			
			-			

Renesas Pmod Type 6A Sensor Board						
Pin	Function					
#1	IRQ					
#2	RESET					
#3	IIC_SCL					
#4	IIC_SDA					
#5	GND					
#6	VCC					
#7	BUSY#					
#8	ENABLE					
#9	POWER_ON					
#10	GPIO					

6.3.1 サンプルプロジェクトのインポート

サンプルプロジェクトのインポートは以下の手順で行います。

1. メニューから、[Import]を選択します。

表示されたインポートウィンドウで、"Rename & Import Existing C/C++ Project into Workspace"を選択し、[Next]ボタンを押下します。

🕲 Import	_		×
Select Rename and Import and Existing C/C++ Project into the workspace		Ľ	1
<u>S</u> elect an import wizard:			
type filter text			
 ✓ Construction ✓ Construction ✓ CMSIS Pack ✓ Existing Projects into Workspace ○ File System ✓ GNUARM-NONE/RZ(DS-5) project conversion to GCC AR ○ Preferences ○ Projects from Folder or Archive ○ Rename & Import Existing C/C++ Project into Workspace ☑ Preferences ○ Projects from Folder or Archive ○ Renesas CSRV project conversion to Renesas GCC RX ○ Renesas CS+ Project for CA78K0/CA78K0 ○ Renesas GitHub FreeRTOS (with IoT libraries) Project 	M Emb	edded	~
(?) < Back Next > Einish		Cance	el

- 2. [Browse]ボタンを押下し、フォルダの選択ウィンドウを表示します。
- インポート済みのサンプルプロジェクトから、移行元デバイスのプロジェクトのフォルダを選択し、[Select Folder]ボタンを押下します。

Select Folder					×	
$\leftarrow \rightarrow$ \checkmark \uparrow \frown Windows (C:) \Rightarrow	→ work	space > e2_studio > FS3000	✓ O Search FS3000		9	
Organize 👻 New folder				== -	?	
workspace	^	Name	Date modified	Туре		
e2_studio		.metadata	10/8/2024 4:39 PM	File folder		
		FS3000_RL78G23_NonOS	10/8/2024 4:44 PM	File folder		
.metadata						
FS3000_RL78G23_NonOS	~ -	¢ line line line line line line line line			>	
Folder: FS3000_RL78G23_NonOS						
			Select Folder	Cancel		

3. プロジェクト名の入力および、移行元デバイスのプロジェクトを選択し、[Finish]ボタンを押下します。

Import	_		Х
Rename & Import Proje Select a directory to search	ct n for existing Eclipse projects.		
Project name: FS3000_RL	78G22_NonOS		
Use default location			
Location: C:\w	orkspace\e2_studio\FS3000\FS3000_RL78G22_♪ eate Directory for Project	Browse	
Choose file system: defa	ult \sim		
Import from:			
• Select root directory:	C:\workspace\e2_studio\FS3000\FS3000_F ~	Browse	
○ Select archive file:	×	Browse	
Projects: FS3000_RL78G23_NonC	DS (C:\workspace\e2_studio\FS3000\FS3000_RL7	78G23_NonOS	
Keep build configurati	on output folders		
?	Back Next > Finish	Cancel	I

6.3.2 デバイスの変更

プロジェクトツリーでインポートしたプロジェクトを選択し、右クリックでコンテキストメニューを表示します。表示されたメニューから、"Change Device"を選択します。

Project Explorer				
ES3000 R178G22 NonO	E [Hards	wareDebugl		
FS3000_RL78G23_NonOS	, filarai	New	>	
		Go Into		
		Open in New Window		
		Show In	Alt, Chiff, W/ N	
		Show In	AIC+ SHITC+ W /	
		Сору	Ctrl+C	
	Ē	Paste	Ctrl+V	
	×	Delete	Delete	
		Source	>	
		Move		
		Rename	F2	
	2	Import		
	<u>~</u>	Export		
		Build Project		
		Clean Project		
	ഭ	Refresh	F5	
		Close Project		
		Close Unrelated Project		
		Puild Targets		
		Index	()	
		Ruild Configurations	()	
		-		
		Source	>	
	0	Run As	>	
	枠	Debug As	>	
		Team	>	
		Compare With	>	
		Restore from Local History		
		MISRA-C	>	
	1	C/C++ Project Settings	Ctrl+Alt+P	
	14.7	Renesas C/C++ Project Settings	>	Change Device
	*	Run C/C++ Code Analysis		Change Toolchain Version
		System Explorer		Save build settings report
	<u> (%)</u>	Command Prompt		
		Validate		
		Configure	>	
		Source	>	
		Properties	Alta Enter	

2. Change Device ウィンドウで、変更したいボードもしくは、デバイスを選択し、[Next]ボタンを押下します。

Refactoring						×
Change Devic Select the new	hange Device Select the new device for FS3000_RL78G22_NonOS					
Current Device: Custom	R7F100GSNxFB					
Target Board:	RL78G22_FastPrototy	pingBoard				\sim
				Download a	dditional bo	ards
Target Device:	R7F102GGExFB					
					Unlock Dev	ices
0		c De els	Marcha	Finish	Contract	
\bigcirc		< Back	ivext >	FINISN	Cance	21

3. Warning が表示された場合、内容を確認し問題無ければ、[Next]を押下します。

Refactoring			—		×
Change Device Review the information provided in	the list below. Clic	k 'Next >' to view th	ne next item or 'Fini	ish'. 🖹	
Found problems					0 û
This change cannot be undone. P	lease make sure yo	ou backup this proje	ect before continuir	ng.	
	No context infor	mation available			
(?)	< Back	Next >	Finish	Canc	el

4. 変更内容が表示されるため、[Finish]ボタンを押下して、変更を実行します。

Befactoring				×
Change Device				
The following changes to 4 files are necessary to perform the refactoring.				
Changes to be performed		Ŷ		7 -
V 🖓 😓 Change Device for FS3000_RL78G22_NonOS				
V 🖉 🚵 Launch Configurations				
RL/8G22_NonOS HardwareDebug				
Project Files				
🗸 🎂 Project nes				
No preview available				
? < Back Next > Finish	n	(Cance	
			_	

6.3.3 Smart Configurator 設定の変更

プロジェクトツリーで、デバイス変更したインポートしたプロジェクトの.scfg ファイルをダブルクリックし、Smart Configurator を表示します。

(1) Board

Board タブで、変更したボード、デバイスに変更されていることを確認します。

ボード、デバイスの変更が反映されていない場合、[…]ボタンを押下し、Change Device ウィンドウで再 度デバイス変更を行ってください。

🔅 FS3000_R	L78G23_NonOS.scfg ×
Device s	election
Device se	lection
Board:	RL78G22_FastPrototypingBoard
Device:	R7F102GGExFB
	Download more boards

(2) Clocks

Clocks タブで、使用するボードに合わせてクロックを設定します。

	siscig A						
ocks configurati	on					Generate	Code Generate R
	3						
Operation mode:	ligh speed main r	mode 4.0(\)0 5.50(\)		•			
	ngri speca man i	100C 4.0(V)- 5.5(V)					
a tara a ar							
Frequency:	p oscillator	• (MHz)					
fHOCO start setting:	Normal	((((())))))					fIHP
(These is setting)	involuted birt	and an akin and the second					32
the times of release	from STOP more	de and of transitions to					fMAIN
SNOOZE mode.)				ŏ_		•	32
				/			fCLK
							iour.
							32000
							32000
Middle-speed on-	chip oscillator						32000 fIMP
Middle-speed on-o	thip oscillator	• (MHz)					32000 fIMP
Middle-speed on-o	chip oscillator	(MHz)	Divider				32000 flMP 🕒
Middle-speed on-o Frequency:	thip oscillator	• (MHz)	Divider				32000 fiMP () -
Middle-speed on-o Frequency: X1 oscillator Operation mode:	4 X1 oscillation	• (MHz)	Divider				
Middle-speed on-o Frequency: X1 oscillator Operation mode: Frequency:	A A X1 oscillation 5	 (MHz) (MHz) 	Divider				32000 flMP
Middle-speed on-o Frequency: X1 oscillator Operation mode: Frequency: Stable time:	4 X1 oscillation 5 2^18/fx	 (MHz) (MHz) (MHz) 52428.8(µs) 	Divider x1 *				
Middle-speed on- Frequency: X1 oscillator Operation mode: Frequency: Stable time:	4 X1 oscillation 5 2^18/fx	 (MHz) (MHz) (MHz) \$2428.8(µs) 	Divider				6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Middle-speed on- Frequency: X1 oscillator Operation mode: Frequency: Stable time: Low-speed on-chip os	hip oscillator 4 X1 oscillation 5 2^18/fx cillator	 (MHz) (MHz) (MHz) 52428.8(µz) 	Divider		•		132000 132000 1117 € 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.

(3) Components

Components タブで、使用するボードに合わせて、各 component の設定を変更します。

(a) I2C ドライバ設定変更

SAU は FS3000 センサが要求するクロック・ストレッチ機能を未サポートのため使用できません。 I2C ドライバ設定の変更は以下の手順で行います。

1. RL78/G22 では、IICA として使用可能なリソースが IICA0 のみのため、既存の Config_IICA1 コンポー ネントを削除します。

2. ソフトウェアコンポーネントの選択から、IIC 通信(マスタモード)を選択し、リソースに IICA0 を指定し てください。

New Component				Х	New Component		-		×
Software Component Selection					Add new configuratio	n for selected component			-
Select component from those availab	ole in list		1		, ,			1	
Category All				~	IIC Communication (M	laster mode)			
Exection All					Configuration name:	Config_IICA0			
An				~	Resource:	IICA0			~
Filter									
Components	Short Name	Туре	Version	^					
H A/D Converter		Code Generator	1.5.0						
Board Support Packages v1.62	2 r_bsp	RL78 Software I	1.62						
H Clock Output /Buzzer Output Co	ontroller	Code Generator	1.4.1						
🖶 Data Transfer Controller		Code Generator	1.3.1						
# Delay Counter		Code Generator	1.4.1						
H Divider Function		Code Generator	1.4.2						
Event Link Controller		Code Generator	1.2.0						
External Event Counter		Code Generator	1.4.1						
FS1015 Sensor Middleware	r_fs1015	RL78 Software I	x.xx						
# FS2012 Sensor Middleware	r_fs2012	RL78 Software I	x.xx						
# FS3000 Sensor Middleware	r_fs3000	RL78 Software I	x.xx						
HS300x Sensor Middleware	r_hs300x	RL78 Software I	x.xx						
HS400x Sensor Middleware	r_hs400x	RL78 Software I	x.xx						
HIC Communication (Master mod	de)	Code Generator	1.5.1						
IIC Communication (Slave mode)	Code Generator	1.4.1						
HIC Communication Driver Interf	ace Mid r_comms_i2c	RL78 Software I	x.xx						
HINDUT Pulse Interval/Period Mea	surement	Code Generator	1.4.3						
HInput Signal High-/Low-Level W	Vidth Me	Code Generator	1.4.2	~					
Show only latest version				•					
Description									
This is a clocked communication fu using two lines: serial clock (SCL) at Download RL78 Software Integration Configure general settings	nction (Master mode) to commu nd serial data (SDA). <u>- System modules</u>	nicate with two or more	devices b	y ^					
?	< Back Next >	Finish	Cance	el	?	< Back Next > Fi	nish	Canc	el

3. Config_IICA0 で Clock mode setting を"fCLK/2"に変更し、Generated stop condition in master transmission/reception end callback function のチェックを外します。

rasooo_kcroozs_Nonos.scrg ×			
oftware component configurati	on		🐻 📄 Generate Code 🛛 Generate Report
Components 🚵 🛃 📮 🖽	Configure		
type filter text ✓ Startup ✓ Startup ✓ Startup ✓ Drivers ✓ Drivers ✓ Config_TAU0_0 ✓ Communications Startup ✓ Communications ✓ Config_ICA0 ✓ Seneric Seneric Seneric Seneric Striddleware ✓ Generic Striddleware ✓ Generic Striddleware ✓ r_r63000	Clock mode setting Clock mode setting Local address setting Address Operation mode setting © Standard Digital filter on Transfer clock (fSCL) tR and tF setting Set tR and tF manually tF	fCLK Image: Constraint of the second secon	(Clock frequency: 32000 kHz) O Fast mode plus (Actual value: 99688.474)
	Interrupt setting Communication end interrupt priority (INTIICA0) Callback function setting Master transmission end Callback function enhanced feature setting	Level 3 (low)	Master error

(b) COMMS_I2C 設定変更

設定が妥当か見直してください。I2C ドライバを変更した場合、見直しが必要です。

r_comms_i2c で Shared Bus0 を"Config_IICA0"に変更します。

ftware component configurat	ion	Generate Code Generate Re			
mponents $\ge \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ $					
🕫 🛃 🐞 🖜	Property	Value			
	# Parameter Checking	System Default			
> 🔁 Startup	# Number of I2C Shared Buses	1			
V 🔁 Generic	# Number of I2C communication Devices	1			
er_psp	# I2C Driver Type for I2C Shared Bus0	IICA			
Drivers	# Component name for the I2C Shared Bus0	Config_IICA0			
Carefin TALIO O	# Driver Type for I2C Shared Bus1	Not selected			
Config_IAUU_U	# Component name for the I2C Shared Bus1	Config_IIC00			
Communications	# Driver Type for I2C Shared Bus2	Not selected Config_IIC00			
Config_IICAU	# Component name for the I2C Shared Bus2				
	# Driver Type for I2C Shared Bus3	Not selected			
	# Component name for the I2C Shared Bus3	Config_IIC00			
0 + 6-2000	# Driver Type for I2C Shared Bus4	Not selected			
* r_ts3000	# Component name for the I2C bus4	Config_IIC00			
	# I2C Shared Bus No. for I2C Communication Device0	I2C bus0			
	# Slave address for I2C Communication Device0	0x28			
	# Callback function for I2C Communication Device0	rm_fs3000_callback0			
	# I2C Shared Bus No. for I2C Communication Device1	I2C bus0			
	# Slave address for I2C Communication Device1	0x00			
	# Callback function for I2C Communication Device1	comms_i2c_user_callback1			
	# I2C Shared Bus No. for I2C Communication Device2	I2C bus0			

(4) Pins

(a) I2C I/F 端子変更

Pins タブの IICA を選択し、Pin function で IICA 端子に端子機能が割り当てられていることを確認します。

🛱 *FS3000_RL78G23_NonOS.scfg × 🗧 🗖							
Pin configuration					Gen	erate Code Ge	inerate Report
Hardware Resource 🗉 🗎 🕹	Pin Functio	n				2 🖬 🛙	1 2 2
	type filter	text (* = any str	ing, ? = any chara	cter)		All	~
TAU07 Cock TAU07 Cock Cock PCLBUZ0 PCLBUZ0 PCLBUZ1 A/D Converter PCB Serial Array Unit	Enabled	Function SCLA0 SDAA0	PIOR PIOR2 PIOR2	Assignment P60/SCLA0 P61/SDAA0	; 6 6	^v in Number 1 2	Direction IO IO
 ✓ SAU0 ✓ SAU00 ✓ SAU01 ✓ SAU02 ✓ SAU02 ✓ SAU3 ✓ SAU10 ✓ SAU10 ✓ SAU11 							
Interrupt Function Interrupt Interr							
Power Supply Voltage Regulator Pin Function Pin Number	<						>
Overview Board Clocks System Components Pir	s Interrupt						

(5) コード生成とビルドの実行

[Generate Code]アイコンを押下して、コード生成を行います。

🔅 *FS3000_RL78G23_NonOS.scfg 🗙							
Overview information			Generate Code Generate Report				
Overview			^				
Get an <u>overview</u> of the features provided by Sn	nart Configurat	tor.					
Videos		Application Code					
Introduction to Smart Configurator		Application couc	SL				
Browse related videos		Software Components					
			2 (1)				
👝 What's New		Drivers					
Check out <u>what's new</u> in the latest release.							
See all <u>Release Notes</u> .		Device Drivers					
Product Documentation and FAG)		at				
API manual	All manual MICO Hardware						
Application Notes							
Tool news							
FAQ : Smart Configurator							
- Current Configuration							
Selected board/device: R/F102GGExFB (ROM size: 64 KB, RAF	/I size: 4 KB, Pi	n count: 48)					
Generated location (PROJECT_LOC\): src\smc_gen Edit							
Selected components:							
Component	Version	Configuration					
Board Support Packages v1.70 (r_bsp)	1.70	r_bsp(used)					
FS3000 Sensor Middleware (r_fs3000)	1.01	r_fs3000(used)					
 IIC Communication (Master mode) 	1.6.0	Config_IICA0(IICA0: used)					
IIC Communication Driver Interface Middleware (r_co	1.11	r_comms_i2c(used)					
C Interval limer	1.5.0	Contig_IAU0_0(IAU0_0: used)	~				
Overview Board Clocks System Components Pins Interrup	nt						

「6.3.4 生成コードの変更」を実施後、プロジェクトをビルドします。

メニューから[Debug Configurations]を選択し、使用するボードに接続するエミュレータに合わせて、 Debugger の設定を変更してください。

6.3.4 生成コードの変更

Config_IICA0_user.c を開き、以下のコードを追加します。

r_comms_i2c_if.h のインクルード定義

コールバック関数に rm_comms_i2c_bus0_callback()関数を追加してください。

送受信完了コールバックは、引数を false に、エラーコールバックは、引数を true に設定してください。

```
* Function Name: r_Config_IICA0_callback_master_sendend
* Description : This function is a callback function when IICA0 finishes master
transmission.
* Arguments : None
* Return Value : None
static void r_Config_IICA0_callback_master_sendend(void)
/* Start user code for r Config IICAO callback master sendend. Do not edit comment
generated here */
 rm comms i2c bus0 callback(false);
/* End user code. Do not edit comment generated here */
}
* Function Name: r Config IICAO callback master receiveend
* Description : This function is a callback function when IICAO finishes master
reception.
* Arguments
          : None
* Return Value : None
                *****
*****
static void r Config IICA0 callback master receiveend(void)
/* Start user code for r_Config_IICA0_callback_master_receiveend. Do not edit comment
generated here */
 rm_comms_i2c_bus0_callback(false);
/* End user code. Do not edit comment generated here */
}
* Function Name: r Config IICAO callback master error
* Description : This function is a callback function when IICAO master error occurs.
* Arguments : flag
             status flag
* Return Value : None
             * * *
static void r Config IICA0 callback master error(MD STATUS flag)
{
  /* Start user code for r_Config_IICA0_callback_master_error. Do not edit comment
generated here */
 rm comms i2c bus0 callback(true);
  /* End user code. Do not edit comment generated here */
```


6.3.5 ツールチェイン設定変更

LLVM ツールチェインに変更する場合は、サンプルプロジェクト"FS3000_RL78G23_NonOS_LLVM"を使用してください。

CC-RL または、LLVM ツールチェイン以外のツールチェインを使用する場合は、本プロジェクトから FS3000_RL78G23_NonOS.c と RL78_FS3000.c をコピーしてプロジェクトを作成してください。

また、LLVM ツールチェインを使用した際、セクション配置に起因するビルドエラーが発生する可能性があります。その場合、リンカスクリプトを変更する必要があります。

以下に、例としてサンプルプロジェクト"FS3000_RL78G23_NonOS_LLVM"で実施している linker_script.ld ファイルの変更を記載します。

1. .text セクションを.frodata セクション以降に配置しています。

2. .rodata セクションの配置アドレスをミラー領域の先頭アドレスに固定しています。

3. .data セクションを.ocd_traceram セクション以降のアドレスに配置しています。

6.3.6 LLVM ビルド時の注意

LLVM プロジェクトをビルド時、以下の Warning が発生します。

これらの Warning は、COMMS_I2C で使用される"slave_address"と"bytes"を 32-bit 型で扱っているため 発生しています。

"slave_address"は 7-bit データ、"bytes"は 16-bit データのため、変換によるロスは発生しません。

そのため Warning が表示されますが、動作に問題はありません。

Problems X	7	000	- 1	3
0 errors, 5 warnings, 0 others				
Description	Resou	urce		
Vanings (5 items)				
😘 implicit conversion loses integer precision: 'const uint32_t' (aka 'const unsigned long') to 'uint16_t' (aka 'unsigned short') [-Wimplicit-int-conversion]	rm_cc	omms	_i2c_d	rive
😘 implicit conversion loses integer precision: 'const uint32_t' (aka 'const unsigned long') to 'uint16_t' (aka 'unsigned short') [-Wimplicit-int-conversion]	rm_cc	omms	_i2c_d	rive
😘 implicit conversion loses integer precision: 'uint32_t' (aka 'unsigned long') to 'uint8_t' (aka 'unsigned char') [-Wimplicit-int-conversion]	rm_cc	omms	_i2c_d	rive
😘 implicit conversion loses integer precision: 'uint32_t' (aka 'unsigned long') to 'uint8_t' (aka 'unsigned char') [-Wimplicit-int-conversion]	rm_cc	omms	i2c_d	rive
😘 implicit conversion loses integer precision: 'uint32_t' (aka 'unsigned long') to 'uint8_t' (aka 'unsigned char') [-Wimplicit-int-conversion]	rm_cc	omms	_i2c_d	rive
<				>
				-

6.4 割り込み信号回路に関する注意

FS3000から出力される割り込み要求信号はありません。

また、FS3000 Sensor Pmod Board にプルアップ抵抗回路と割り込み信号回路が無いため、Renesas sensor Pmod ボードのデイジ・チェーン接続時の注意はありません。

6.5 RESET 信号回路に関する注意

FS3000 に RESET 入力端子はありません。

また、FS3000 Sensor Pmod Board にプルアップ抵抗回路が無いため、Renesas sensor Pmod ボードの デイジ・チェーン接続時の注意はありません。

6.6 Renesas Pmod Sensor Board のデイジ・チェーン接続時のプルアップ抵抗処理

デイジ・チェーン接続時のプルアップ抵抗の推奨処理を以下に示します。また、他ボードのプルアップ処理を無効にしてください。

多数の Sensor Board のプルアップ抵抗を同時に有効にした場合、正常に動作しない場合があります。

Pmod Sensor Board Type 6A 信号名	プルアップ抵抗の推奨処理
#1: IRQ# (注 1)	プルアップ抵抗回路付ボードで、最も MCU ボードに近いボードのみ有効
#2: RESET# (注 1)	プルアップ抵抗回路付ボードで、最も MCU ボードに近いボードのみ有効
#3: SCL	最も MCU ボードに近いボードのみ有効
#4: SDA	最も MCU ボードに近いボードのみ有効
#7: BUSY# (注 1, 2)	プルアップ抵抗回路付ボードで、最も MCU ボードに近いボードのみ有効

表 6-2 デイジ・チェーン接続時のプルアップ抵抗を有効化する対象ボード

注1:プルアップ抵抗回路無しのボードがあります。

注2:割り込み要求信号として利用時に処理してください。

7. 風速データの確認方法

リアルタイムの風速データは、以下の手順に従って確認することができます。

1. Debug を実行後、Expressions ウィンドウを開いてください。

Expressions ウィンドウは[Window]→[Show View]→[Expressions]から開くことができます。

🖬 workspace P3500 - NA P3500 Benthelafver/RA P35001 - e' shallo File Falt Source Refector Novigate Search Perject RemoverViews Russ	Window Help							- 0 X	
🔏 🔯 💼 🔯 Debug 🗸 🗸 🕅 RA, IS3000, illare/Metal Debug	New Window	. 1	8 - 5 - 8 1 2 2 2 4 -	/ * * * * * * * * * * * * * * * * * * *	1 No - 18 (10 111)	17 📽 🕹 👁 (종) - 80	- 0.07 01 - 0 - 1		
	Arrentore	- 1					Q	The Cit C/C++ to Datag	
🕼 Debug 27 📄 👘 🛔 🖤 🛔 🖤	Septembrie			Terupic R mainic		00+1/h	10. Pt 12 . 12 . f.	5. Pe. TL "D	
V 7 RA, FS000, Bare Metal Debug, Rat (Renesas GDB Hardware Debuoging)	Show View	- 2 -	Breakpoints Alt+Shift+Q.B		~		1.000	250 0 0 10 11 1	
v 🔐 RA, FS3000, Barel/fetal.eff [1] (cover 0)	Perspective	> 1	Console Att+Shitt+Q, C			Automation Tunn	Mars Address		
🗸 🧬 Thread #1.1 (single care) (cores of (Suspended): Breakpoint)	Nevigation		p Debag			Add on	Patrice Patrices		
start_demo() at FA_F53000.c139.0k278		- 1	B Debagger Console			Y MILLING			
hal_entry() at hal_entry.c 15 0x330	Pretenetices	0	Debag Shell						
manu at mars.cls 0x54e	18	- 6	Debug Sources						
Reneral SUB server (Hod)	10		E Disasanikiy						
	20		EmorLos Alt+Shift+Q.L	ect "/					
	22	1.	Tarrutables						
	23		- Internet						
	24	1	2 Martin						
	20		1 Menory	The second se					
	27		Memory Sconser	gs_demo_callback_status;					
	28	11	Modules	gs_demo_output_count;					
	29	2	Outline Alt+Shift+Q, 0	BrToppoTrans					
	32	8.	6 Peripherals						
	32	18	Problems Alt+Shift+Q, X						
	34		Progress						
	35	16	Protect Exclorer	1000000					
	.36		Benister	prever_1;					
	18		- finale						
	. 39 8666603	s (a ajuns						
	40		M temptates						
	42 88866035		 Variables Alt+Shitt+Q, V 						
	43		か スマート・ブランダー						
	44	e 6	Q X7-1-722P#						
	45 0000028	. 6	Othera Att+Shift+Q, Q						
	40		CALL DEND SECURICE 2 :						
	4.9		(
	50		/* Clear status */	THE - OTHER CHARGE STATIS WITH					
	51 0000028	0	gs_deeo_calleace_sta	Hus - Deno_Cretemen_Status_willij	~				
		1						- K	
	Console 11	III Regi	ters 🕕 Debug Shell 🐒 Problems 📿 De	bugger Console 🕘 XR-1-7577- 🚺 Memory	= # 2011년 · · · · · · · · · · · · · · · · · ·				
	FA PSIOD Each	inta: Deta	ig Part (Nerseau trait Hardware Delargging)						
	9-0-01400125	- 06							
	9-591報約165	- 0K							
	多72/G一种国际			Contract of the second second and the second s					
	Option Function	on bala	at writing to address Building 100	with data for according to the test of tes					
	9750-1987		ce) a read to address mutueature						
	A-POTP-71-	MON	2PU2ex34605508U#T.						
	パードウエア・ブレー	味白枝	アドレス8x278に設定します。						
(> <							, ,	
SSD:Secure - 0x0000278 () His ns	6	50 SWT 30					1 10 + 2 0	
								1	

2. Expressions 内の Add new expression をクリックして、"gs_fs3000_data"を追加してください。

									Q	1 23 El OC++ (\$ Debe
Debug 11 🕒 👘 🕴 🕯 **	C: RA_F53000.c 31	(BA_F53000_BareMetal) FSP Configuration	🗟 startupic	R main.e	· · · ·	(in-Varia., % Bres., 26 Mod	L. 🙆 Proj	1 Dept. 22	· (10). 5	Pel. [] IO R.,
FALFS3000_BareMetal Debug_Flat (Renesas GD8 Hardinare Debugging)	11	} demo_sequence_t;			^			志	海日日	三 茶 健 第 15 日
(a) PA (Stock Antonian (1) (Store B) (B) (Stock Antonian (1) (Store B) (B) (Stock Antonian (1) (Store B) (B) (Store B) (Store B) (B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store B) (Store	10 10 10 10 10 10 10 10 10 10	<pre>> pyme run = _im cites cites_ run _ cites cite cites run _ run _ run _ cites cite run _ run _ run _ cites run _ run _ run _ run _ cites run _ run - run _ run _ run _ run</pre>	roject */ id); t gs.demo_ gs_demo_ gs_demo_ sequence_I;	થોંગ્રિયો કેમ્પ્રેલન તોકનું આવે: સ્થિતુ		lipptino" (Φ - g.,dom,c.yhpt.cov) (Φ - g.,dom,c.ho Φ . Add.or expended	hpe votatio antă. votatie float	Value 0 0	Address 0x20000 0x303800	
	42 00000209 43 45 46 0000028e 49 50 51 0000028e 51 0000028e 51 0000028e 51 0000028e	<pre>E_112000_sectore_globs_setup(); 0 while (1) 6 soutch (securice) 7 (Chern status ?) 8 gl_dem_csllubs_stru 9 gl_dem_csllubs_tru 9 (</pre>	tus - aeno_c	NLEACH_STATUS_NUT1	ž	¢			,	
	■ Console 云 単 RA, FSYND, Barribler ラーマル・相関はOE - ジンロート開始 Option Function のたいロート制で ハードウェア・フレークガ ハードウェア・フレークガ ハードウェア・フレークガ ハードウェア・フレークガ	Hoppins () Decay Stell []: Frederic () de d Debug Ret (Denses GDB Handware Debugging) Select, writing to address endoledade () Delect, writing to address (Moldedade () Delect, writing to address (Moldedade () Delect, writing to address (Moldedade () Delect, Writing () delected () Delect () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected () Delected ()	ugger Console eith data ff	کی ۲۲-۱۰ ۲۶۶۶۰ () Manay				1 H % %	5 × 5	2 d Q • 🗂 • "

3. 追加した変数を右クリックすると、Enable Real-time Refresh を選択することができます。

4. Debug をスタートすると、リアルタイムの値を確認することができます。

The file term Math Rape Level Field Randows Review () () () () () () () () () () () () ()	🕼 werkspace F\$3000 - RA F\$5000 Ban/Metal/sec/RA F\$3000z - e ² studio	- a >
	File Rill Science Refactor Navigate Smech Project Remeas Views Run 1	
<pre>D Dec 11</pre>		
<pre>0 Prove: 1: Control Contro Control Control Control Control Control Control Contro</pre>		Q, [2] (Q, CC++ G Ebbag) (0) F9 Configuration
option Facilita Salati, et ling is a solares abblaciade site data preferenterenterenterenterenterenterenter	9 Enko II	B. M. 1990. 11 Bit M. 1990. M. 1990. 11 Bit Montes B. Markenson B. Ma

改訂記録

		改訂内容		
Rev.	発行日	ページ	ポイント	
Rev.1.00	Jun 30, 2022	-	初版リリース	
Rev.1.01	Mar 3, 2023	-	更新:RL78 の動作環境	
Rev.1.02	Mar 29, 2023	-	更新:RA、RX、RL78、RZ の動作確認環境	
			更新:サンプルソフトウェアメインフロー	
			更新:デバイス変更ガイド	
Rev.1.03	Sep 7, 2023	-	更新:デバイス変更ガイド	
			削除:RE01 の項目	
Rev.1.04	Dec.10.24	-	追加:FS3000-1015 関連	
			追加:用語/略語	
			更新:RA、RL78 の動作確認環境	
			更新:FS3000 センサ仕様	
			更新:サンプルソフトウェア仕様	
			更新:Configuration 設定	
			更新:デバイス変更ガイド	

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_{IL}(Max.)からV_{IH}(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_{IL}(Max.)からV_{IH}(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

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/

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。