

RZ/T1 グループ

R01AN3561JJ0110

Encoder I/F BiSS-C application package

Rev.1.10

2018.04.02

要旨

本書は、RZ/T1 Encoder I/F BiSS-C のアプリケーション・パッケージについて記載します。

本アプリケーション・パッケージを使用する時は、「RZ/T1 Encoder I/F Configuration Library」のリリースパッケージを入手してください。

BiSS C モードに関する技術的詳細については、iC-Haus 社にお問い合わせの上、プロトコル仕様書(BiSS C Protocol Description)やエンコーダの仕様書を入手してください。

動作確認デバイス

RZ/T1 CPU ボード (RTK7910022C00000BR)

過去バージョンからの変更履歴

Ver.	Date	内容	備考
1.1	2018.4	RZ/T1 BiSS-C サンプルプログラムをアップデート (1) ch1 用の ID マクロ定義追加 (2) ch1 用レジスタ定義変更 (3) SCIFA サンプルプログラム追加 RZ/T1 グループ BiSS インタフェース (BiSS) ユーザーズマニュアルをアップデート	
1.0	2017.1	RZ/T1 グループ BiSS インタフェース (BiSS) ユーザーズマニュアルをアップデート。	
0.9	2016.12	RZ/T1 BiSS-C サンプルプログラムをアップデート。 (4) 割り込み処理内のエラー処理を変更 (5) KPIT GCC 版を追加 (6) モジュールストップ解除動作の安定性を向上 (7) 割り込み処理動作の安定性を向上 RZ/T1 グループ BiSS インタフェース (BiSS) ユーザーズマニュアルをアップデート。 RZ/T1 グループ BiSS-C サンプルプログラム アプリケーションノートを更新。	
0.8	2015.8	新規	

目次

1. パッケージ内容	3
1.1 ソフトウェア	3
・ソースコード	3
・コンフィグレーション・データ	3
1.2 ドキュメント	3
2. ファイル構成	4
3. BiSS サンプルプログラムについて	5
3.1 ソフトウェア情報	5
3.1.1 ベース OS	5
3.1.2 メモリサイズ	5
3.2 ハードウェア情報	6
3.2.1 デバイス	6
3.2.2 ターゲットボード	6
3.3 各開発環境における動作手順	7
3.3.1 サンプルプログラムの実行前の準備	7
3.3.2 EWARM : IAR 社製	7
3.3.3 DS-5 : ARM 社製	9
3.3.4 e2 studio : RENESAS 社製	12
4. 制限事項	14
5. 注意事項	14
5.1 処理時間	14
5.2 検証済みエンコーダー一覧	15

1. パッケージ内容

本パッケージには次のものが含まれています。

本パッケージに含まれている Configuration Data やサンプルプログラムは、1 軸にのみ対応しています。2 軸で使用する場合は、RZ/T1 グループ Encoder I/F 2ch Tool (R01AN4306) を入手し、Configuration Data とサンプルプログラムを変更して使用してください。

1.1 ソフトウェア

・ソースコード

No.	名称	版数
1	RZ/T1 BiSS-C サンプルドライバコード一式	1.2

・コンフィグレーション・データ

No.	名称	版数
1	RZ/T1 Encoder I/F Configuration Data (BiSS)	1.0

1.2 ドキュメント

No.	文書名	版数	ファイル名
1	RZ/T1 Encoder I/F BiSS-C sample program リリースノート	1.10	(日) r01an3561jj0110-rzt1.pdf (本書) (英) r01an3561ej0110-rzt1.pdf
2	RZ/T1 グループ BiSS インタフェース (BiSS) ユーザーズマニュアル	1.50	(日) r01uh0597jj0150_rzt1_biss.pdf (英) r01uh0597ej0150_rzt1_biss.pdf
3	RZ/T1 グループ BiSS-C サンプルプログラム アプリケーションノート	1.20	(日) r01an2792jj0120_rzt1_biss-c.pdf (英) r01an2792ej0120_rzt1_biss-c.pdf

2. ファイル構成

本パッケージのファイル構成と内容物の詳細を以下に示します。

```

Top
├──r01an3561ej0110-rzt1.pdf
├──r01an3561jj0110-rzt1.pdf
├──workspace
│   ├──Software
│   │   ├──armcc
│   │   │   └──RZ_T1_biss.zip : RZ/T1 BiSS-C サンプルドライバコードー式(DS-5)
│   │   ├──iccarm
│   │   │   └──RZ_T1_biss.zip : RZ/T1 BiSS-C サンプルドライバコードー式(IAR)
│   │   └──kpitgcc
│   │       └──RZ_T1_biss.zip : RZ/T1 BiSS-C サンプルドライバコードー式(e2 studio)
│   └──Documentation
│       ├──r01an2792ej0120_rzt1_biss-c.pdf
│       ├──r01an2792jj0120_rzt1_biss-c.pdf
│       ├──r01uh0597ej0150_rzt1_biss.pdf
│       └──r01uh0597jj0150_rzt1_biss.pdf

```

RZ_T1_biss.zip のファイル構成を以下に示します。

Top folder	
inc	
iodefne.h	RZ/T1 レジスタ定義ファイル
iodefne_biss.h	BiSS-C レジスタ定義ファイル
r_biss_rzt1_dat.h	biss.dat 用ヘッダファイル
r_biss_rzt1_if.h	BiSS 共通ドライバヘッダファイル
r_bissc_rzt1_if.h	BiSS-C ドライバヘッダファイル
lib	
ecl	
r_biss_rzt1.dat	Multi-Protocol Encoder IF(BiSS-C mode) Configuration data
src	
common	
初期設定などの共通ソース	
drv	
scifa_uart	
SCIFA サンプルプログラム	
biss	
r_biss_rzt1_config.h	BiSS 共通ドライバファイル
r_biss_rzt1_private.h	BiSS 共通ドライバファイル
r_biss_rzt1.c	BiSS 共通ドライバファイル
bissc	
r_bissc_rzt1_config.h	BiSS-C ドライバファイル
r_bissc_rzt1_private.h	BiSS-C ドライバファイル
r_bissc_rzt1.c	BiSS-C ドライバファイル
sample	
main.c	サンプルプログラム
r_biss_rzt1_dat.s	Configuration data 用リンカ設定ファイル ※1
siorw.c	SCIFA サンプルプログラム
siochar.c	SCIFA サンプルプログラム

※1 : DS-5/e2 studio 用ファイル
 DS-5 : r_biss_rzt1_dat.s
 e2 studio : biss_rzt1_dat.asm

3. BiSS サンプルプログラムについて

BiSS サンプルプログラム一式を使用するために必要な情報を記載します。

3.1 ソフトウェア情報

3.1.1 ベース OS

OS レス

3.1.2 メモリサイズ

領域名		開発環境別メモリサイズ			
		IAR [bytes]	DS-5 [bytes]	e2 studio [bytes]	
BiSS-C サンプルドライバ	コード領域	3076	4492	7912	
	データ領域（初期値あり）	8	46	8	
	データ領域（初期値なし）	104	66	112	
	定数領域	170	172	172	
	スタック サイズ	R_BISS_Open 関数	60	68	104
		R_BISS_Close 関数	36	44	80
R_BISS_Control 関数		64	88	120	
R_BISS_GetVersion 関数		0	16	0	
	bissc0_rx_int_isr 関数	144+※	88+※	128+※	
BiSS-C Configuration data	コード領域	0	0	0	
	データ領域（初期値あり）	0	0	0	
	データ領域（初期値なし）	0	0	0	
	定数領域	20684	20684	20684	
サンプルプログラム	コード領域	2084	2908	4496	
	データ領域（初期値あり）	32	29	16	
	データ領域（初期値なし）	367	356	372	
	定数領域	1100	41	1089	

※ R_BISS_Control 関数に登録するユーザー定義コールバック関数のうち最大のスタックサイズ

3.2 ハードウェア情報

3.2.1 デバイス

RZ/T1

3.2.2 ターゲットボード

(1) ボード名

RZ/T1 CPU ボード(RTK7910022C00000BR)

(2) CPU ボードの設定

ターゲットボードの設定は以下の通り。

SW4-1: ON

SW4-2: シリアルフラッシュを使用する場合は ON、NOR を使用する場合は OFF

SW4-3: ON

SW4-4: ON

SW4-5: ON

SW4-7: OFF

JP2: 2-3 ショート

JP7: 1-2 ショート

3.3 各開発環境における動作手順

3.3.1 サンプルプログラムの実行前の準備

本サンプルプログラムは、PC との通信動作を行います。ホスト PC のターミナルソフトの設定は「RZ/T1 グループ FIFO 内蔵シリアルコミュニケーションインタフェース (SCIFA) アプリケーションノート」の「6.1.2 使用準備」を参照してください。

3.3.2 EWARM : IAR 社製

➤ ビルド環境

IAR Embedded Workbench for ARM v7.80.2

➤ 実行環境

I-jet

➤ サンプルプログラムのビルド手順

サンプルプログラムのビルド手順は以下の通り。

1. 展開したソースファイルを任意の場所にコピー
2. “RZ/T1 Encoder I/F Configuration Library” (IAR EWARM 版)の以下のファイルを各フォルダへコピー

lib¥ecl¥r_ecl_rzt1.a

inc¥r_ecl_rzt1_if.h

3. EWARM を起動
4. [ファイル]メニュー→[開く]→[ワークスペース]を選択
5. 展開したソースファイルの RZ_T1_biss_serial_nor¥RZ_T1_bissc_****_boot.eww を開く

Nor/Serial	プロジェクトファイル
Nor 版	RZ_T1_bissc_nor_boot.eww
Serial Flash 版	RZ_T1_bissc_serial_boot.eww

6. [プロジェクト]メニュー→[すべてを再ビルド]を選択

次のファイルが生成される。

RZ_T1_biss_serial_nor¥Debug¥Exe¥RZ_T1_bissc_****_boot.out

Nor/Serial	プロジェクトファイル
Nor 版	RZ_T1_bissc_nor_boot.out
Serial Flash 版	RZ_T1_bissc_serial_boot.out

➤ サンプルプログラムの実行手順

「ビルド手順」を実行後、ターゲットボード、デバッガを正しく接続し、以下の操作を行う。

1. [プロジェクト]メニュー→[ダウンロードしてデバッグ]を選択
2. [デバッグ]メニュー→[実行]を選択

▶ サンプルプログラムの実行結果

サンプルプログラムを実行し、ターミナルソフトのウィンドウにコマンドを入力してください。

コマンドについては、**RZ/T1** グループ **BiSS-C** サンプルプログラムアプリケーションノートに記載の 7.11.7 コンソールコマンドを参照してください。

3.3.3 DS-5 : ARM 社製

➤ ビルド環境

ARM Development Studio 5 (DS-5) Version 5.25.0

ARM Compiler 5.06 update 3

➤ 実行環境

ULINK2 (v2.01)

➤ サンプルプログラムのビルド手順

サンプルプログラムのビルド手順は以下の通り。

1. 展開したソースファイルを任意の場所にコピー
2. “RZ/T1 Encoder I/F Configuration Library” (ARM DS-5 版)の以下のファイルを、各フォルダへコピー

lib¥ecl¥r_ecl_rzt1.a

inc¥r_ecl_rzt1_if.h

3. DS-5 を起動
4. [ウィンドウ]メニュー→[ビューの表示] →[プロジェクト・エクスプローラー]を選択
5. [プロジェクト・エクスプローラー]ビューを右クリックし、ポップアップメニューの[インポート]を選択
6. 「インポート」ダイアログの「一般」 → 「既存プロジェクトをワークスペースへ」 を選択し、「次へ」 ボタンをクリック
7. 「インポートダイアログ」 の「参照」 をクリック
8. 「フォルダの参照」 ダイアログで、ソースファイルのコピー先のトップを選択し、「OK」 をクリック
9. 「インポート」ダイアログの「プロジェクトをワークスペースにコピー」 のチェックを付ける
10. 「インポート」ダイアログの「終了」 をクリック
11. [プロジェクト]メニュー→[すべてビルド]を選択

次のファイルが生成される。

Debug¥RZ_T_nor_sample.axf

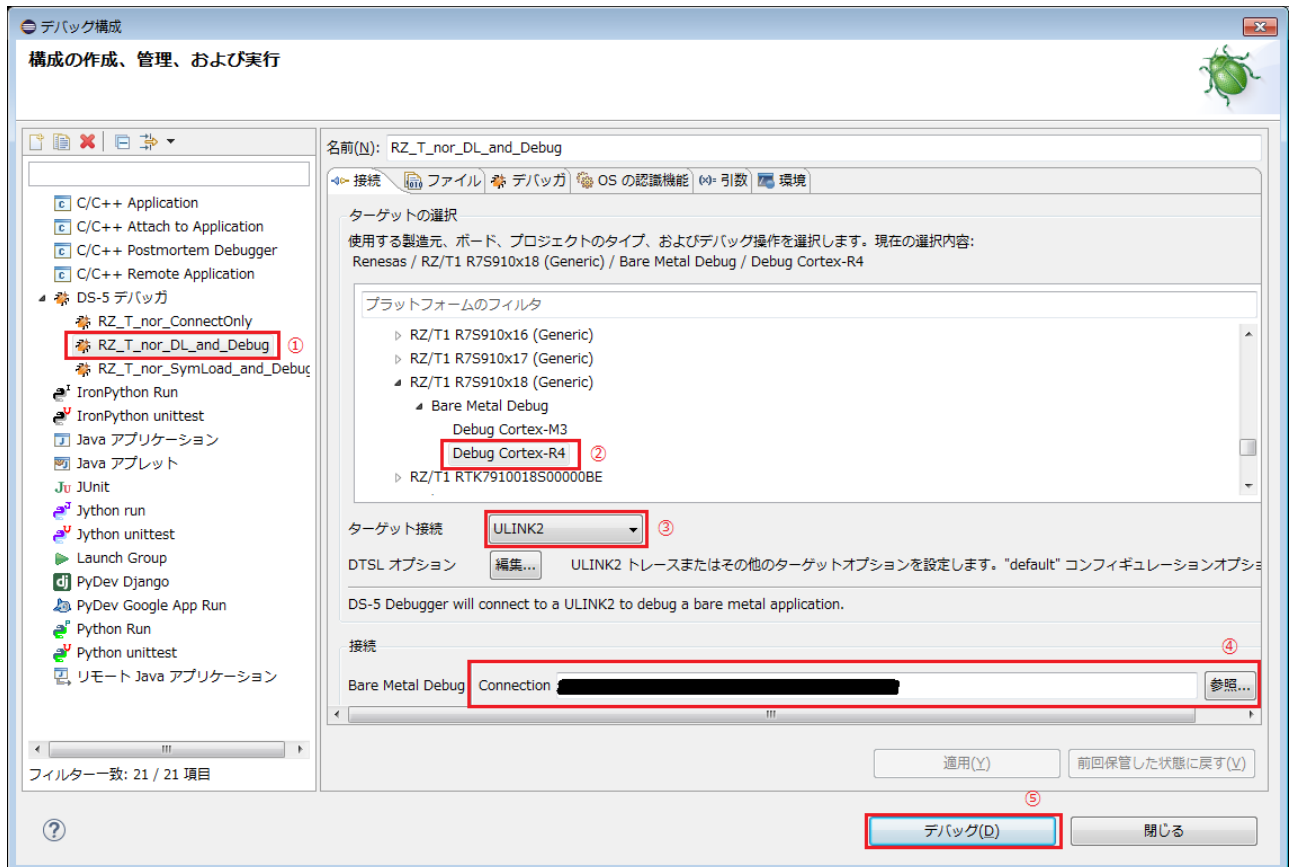
(シリアルフラッシュの場合は、「RZ_T_sflash_sample.axf」)

▶ サンプルプログラムの実行手順

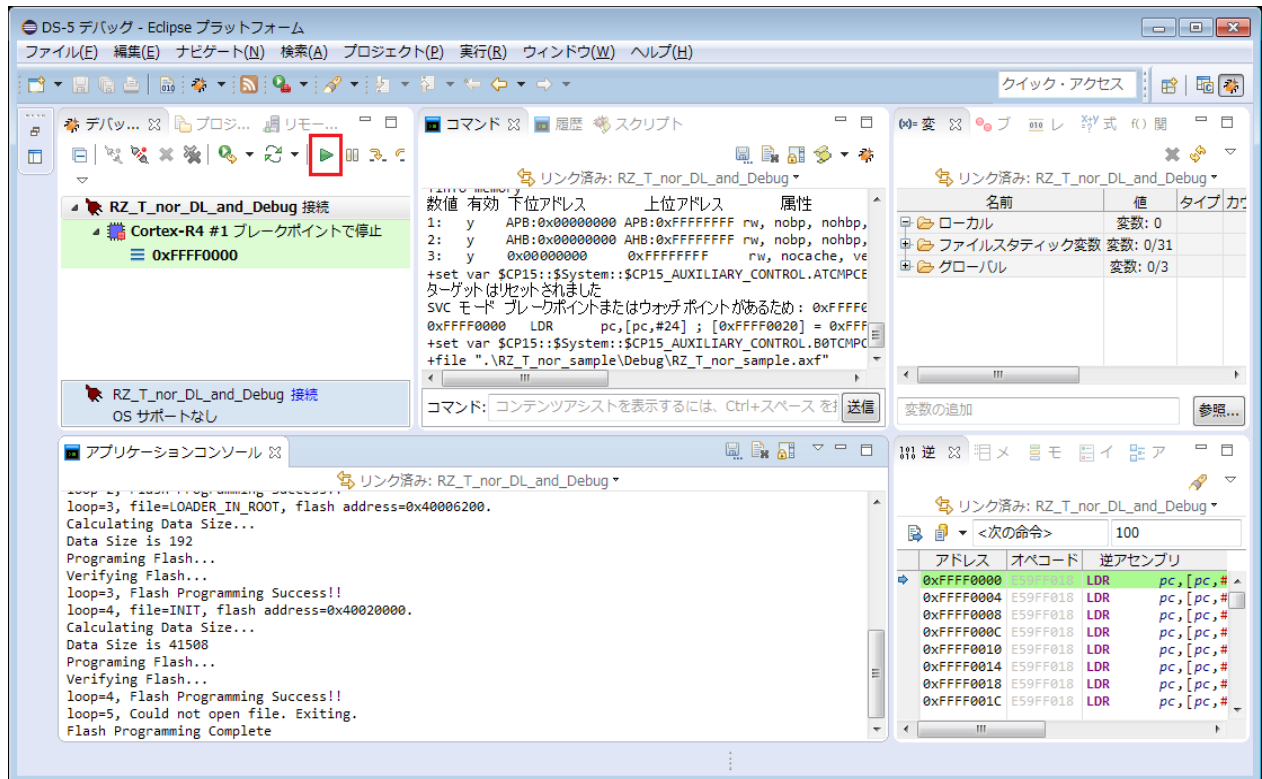
「ビルド手順」を実行後、ターゲットボード、デバッガを正しく接続し、以下の操作を行う。

1. [実行]→[デバッグの構成]でデバッグ構成を開き、“RZ_T_nor_DL_and_Debug”の接続設定を選択します。（シリアルフラッシュの場合は、“RZ_T_sflash_DL_and_Debug”）
ターゲットの選択で “RZ/T1 R7S910x18 (Generic)” の Debug Cortex-R4 を選択します。

接続タブよりターゲット接続の ULINK2 を選択し、[参照] を選択し、検出されたターゲット接続を選択後、[デバッグ] を選択しデバッグを開始します。



2. フラッシュ書き込みが完了すると、アプリケーションコンソール画面に「Flash Programming Complete」が表示され、デバッグを開始できます。



➤ サンプルプログラムの実行結果

サンプルプログラムを実行し、ターミナルソフトのウィンドウにコマンドを入力してください。

コマンドについては、RZ/T1 グループ BiSS-C サンプルプログラムアプリケーションノートに記載の 7.11.7 コンソールコマンドを参照してください。

3.3.4 e2 studio : RENESAS 社製

➤ ビルド環境

RENESAS e2 studio 5.2.0.020

KPIT GNUARM-NONE-EABI Toolchain v16.01

➤ 実行環境

J-Link BASE

➤ サンプルプログラムのビルド手順

サンプルプログラムのビルド手順は以下の通り。

1. 展開したソースファイルを任意の場所にコピー
2. “RZ/T1 Encoder I/F Configuration Library” (KPIT GCC 版)の以下のファイルを、各フォルダへコピー

lib¥r_ecl_rzt1.a

inc¥r_ecl_rzt1_if.h

3. e2studio を起動
4. [ウィンドウ]メニュー→[ビューの表示] →[プロジェクト・エクスプローラー]を選択
5. [プロジェクト・エクスプローラー]ビューを右クリックし、ポップアップメニューの[インポート]を選択
6. 「インポート」ダイアログの「一般」 → 「既存プロジェクトをワークスペースへ」 を選択し、「次へ」 ボタンをクリック
7. 「インポートダイアログ」 の「参照」 をクリック
8. 「フォルダの参照」 ダイアログで、ソースファイルのコピー先のトップを選択し、「OK」 をクリック
9. 「インポート」 ダイアログの「プロジェクトをワークスペースにコピー」 のチェックを付ける
10. 「インポート」 ダイアログの「終了」 をクリック
11. [プロジェクト]メニュー→[すべてビルド]を選択

次のファイルが生成される。

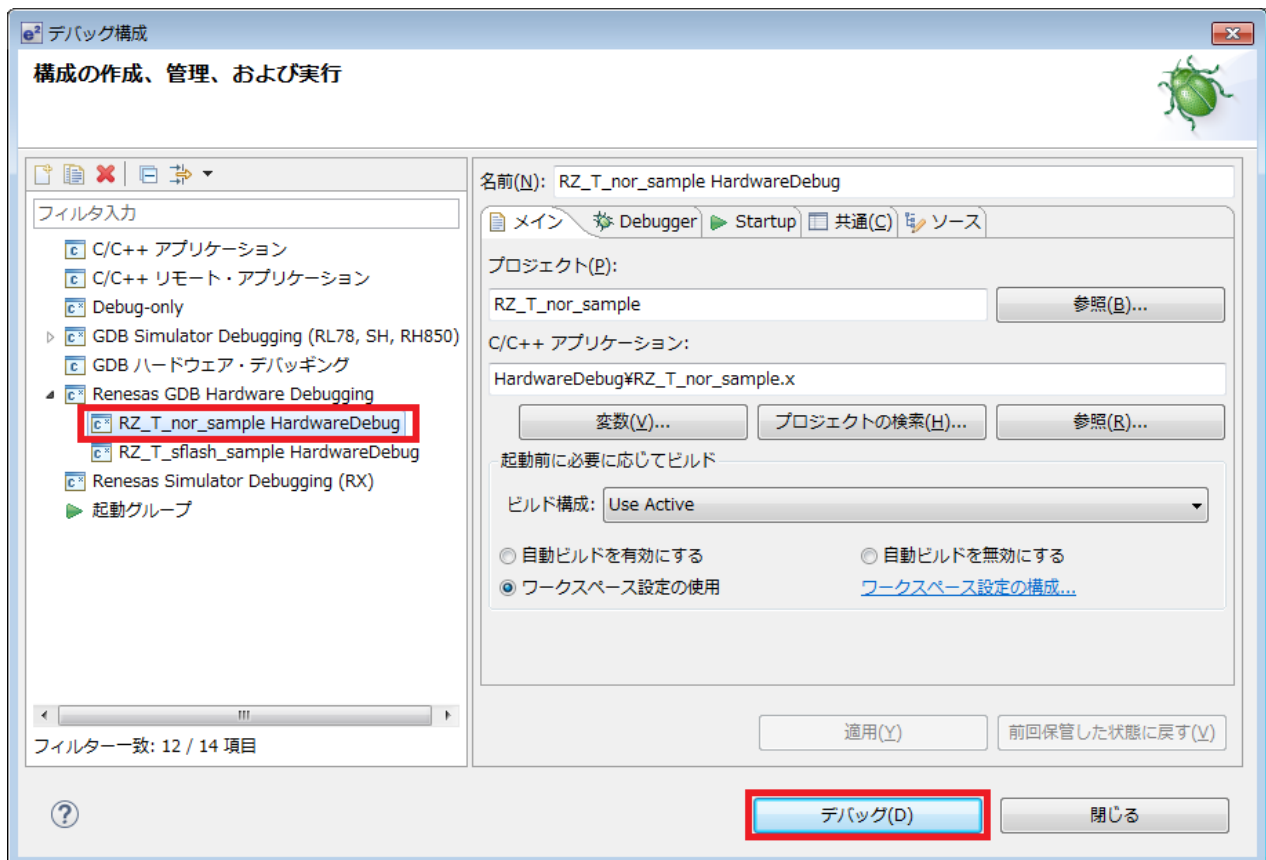
HardwareDebug¥RZ_T_nor_sample.x

(シリアルフラッシュの場合は、「RZ_T_sflash_sample.x」)

▶ サンプルプログラムの実行手順

「ビルド手順」を実行後、ターゲットボード、デバッガを正しく接続し、以下の操作を行う。

1. [プロジェクト] → [すべてビルド] を実行後、[実行] → [デバッグの構成] を選択。
2. 以下の画面の [RZ_T_nor_sample HardwareDebug] を選択、[デバッグ]をクリックするとフラッシュメモリへダウンロードを開始。
(シリアルフラッシュの場合は、「RZ_T_sflash_sample HardwareDebug」)



3. [実行] → [再開]をクリックするとサンプルプログラムの実行を開始。

▶ サンプルプログラムの実行結果

サンプルプログラムを実行し、ターミナルソフトのウィンドウにコマンドを入力してください。コマンドについては、RZ/T1 グループ BiSS-C サンプルプログラムアプリケーションノートの 5.11.7 コンソールコマンドを参照してください。

4. 制限事項

特になし。

5. 注意事項

5.1 処理時間

Encoder I/F BiSS-C サンプルプログラムでは、制御ループにおけるユーザーが使用可能な時間は以下の様になっています。ご使用の環境で問題がないか十分評価を行ってください。

制御周期を 62.5us とした場合の例を以下に示します。62.5us のうち、BiSS-C サンプル処理の使用時間は約 7.7us(13%)となり、ユーザーが使用可能な時間は約 54.8us (87%)となります。

処理		時間		占有率
BiSS-C サンプル処理 ※2	送信レジスタ設定時間	約 2.7us	約 7.7us	13%
	割り込み処理時間	約 5us		
ユーザー使用可能時間		約 54.8us ※1		87%

※1：ユーザー使用可能時間のうち、エンコーダとの通信時間については、「RZ/T1 グループ BiSS インタフェース (BiSS) ユーザーズマニュアル」の 6.1 AC 特性を参照してください。

※2：初期設定時間は除きます。

5.2 検証済みエンコーダー一覧

実機での検証済みエンコーダを以下に示します。

送信ク ロック周 波数	機能	検証済みエンコーダ				
		Danaher 社製 (HENGSTLER) AD36	RENISHAW 社製 RTLA-S	Kuebler 社製 F3663	Lika 社製 AM36	WACHENDO RFF 社製 WDGF 58M
10MHz	位置情報取得	-	○	-		-
	レジスタアクセス	-		-		-
8.33MHz	位置情報取得	○	○	○	○	○
	レジスタアクセス	○		○	○※1	○
4MHz	位置情報取得	○	○	○	○	○
	レジスタアクセス	○		○	○※1	○
2.5MHz	位置情報取得	○	○	○	○	○
	レジスタアクセス	○		○	○※1	○
1MHz	位置情報取得	○	-	-	-	-
	レジスタアクセス	○		-	-	-
400kHz	位置情報取得	○	-	-	-	-
	レジスタアクセス	○		-	-	-
299.4kHz	位置情報取得	○	○	-	-	-
	レジスタアクセス	○		-	-	-
200kHz	位置情報取得	○		-	○	-
	レジスタアクセス	○		-	○※1	-
100kHz	位置情報取得	○		-		-
	レジスタアクセス	○		-		-
80.12kHz	位置情報取得	○		○		○
	レジスタアクセス	○		○		○

○	: 検証済み
○※1	: 連続レジスタアクセスは未検証
-	: 未検証
	: エンコーダ非対応