RZ/T1 Group

Encoder I/F FA-CODER application package

R01AN3558EJ0130
Rev.1.30
April 2, 2018

Summary

This document explains about RZ/T1 Encoder I/F FA-CODER application package.

To use this application package, please obtain release package of "RZ/T1 Encoder I/F Configuration Library".

If you require detailed specifications of FA-CODER protocol and Tamagawa-Seiki encoder, please contact Tamagawa-Seiki Co to obtain them.

Device that FA-CODER functionality is checked

RZ/T1 CPU Board (RTK7910018C00000BE)

Version History

Ver.	Date	Content	Note
1.30	April 2018	Updated sample program to Ver.1.3. (1) Added ID macro definition for ch1. (2) Added the SCIFA sample program. Update the RZ/T1 Group FA-CODER Interface (FAC) User's Manual.	
1.20	February 2017	Updated sample program. (1) Improved the stability of the module stop release operation.	
1.1	December 2016	Updated sample program. (1) Changed interrupt handler	
1.0	October 2016	Updated sample program to Ver.1.1. (1) Updated the processing when EEPROM access reception busy state is 1. Updated Configuration Data to Ver.1.3. (1) Updated the error processing of EEPROM access. Update the RZ/T1 Group FA-CODER Interface (FAC) User's Manual. Update the RZ/T1 Group FA-CODER Sample Program Application Note.	
0.7	October 2015	Newly created	

Table of contents

1. Contents of package

Contents of this package are described in this chapter.

Configuration data and sample programs included in this package support only 1 channel of Encoder I/F. In order to use 2 channels of Encoder I/F, obtain the RZ / T1 group Encoder I / F 2ch Tool (R01AN4306) and change the Configuration Data and sample program.

1.1 Software

・Source code

No.	Title	Version
1	A set of RZ/T1 FA-CODER sample driver code	1.3

・Configuration data

No.	Title	Version
1	RZ/T1 Encoder I/F Configuration Data(FA-CODER)	1.3

1.2 Document

No.	Document name	Ver.	File name
1	RZ/T1 Encoder I/F FA-CODER application package release note	1.30	(English) r01an3558ej0130-rzt1.pdf (this document) (Japanese) r01an3558jj0130-rzt1.pdf
2	RZ/T1 Group FA-CODER Interface (FAC)User's Manual	1.20	(English) r01uh0621ej0120_rzt1_fa-coder.pdf (Japanese) r01uh0621jj0120_rzt1_fa-coder.pdf
3	RZ/T1 Group FA-CODER Sample Program Application Note	1.10	(English) r01an2949ej0110_rzt1_fa-coder.pdf (Japanese) r01an2949jj0110_rzt1_fa-coder.pdf

2. File Structures

File structures and contents of this package are described below.

```
Top
 ├──r01an3558ej0130-rzt1.pdf
 ├──r01an3558jj0130-rzt1.pdf
 └──workspace
     ├──Software
     │   ├──armcc
     │   │   └──RZ_T1_fac.zip：A set of RZ/T1 FA-CODER sample driver code (DS-5)
     │   ├──iccarm
     │   │   └──RZ_T1_fac.zip：A set of RZ/T1 FA-CODER sample driver code (IAR)
     │   └──kpitgcc
     │       └──RZ_T1_fac.zip：A set of RZ/T1 FA-CODER sample driver code (e2 studio)
     └──Documentation
         ├──r01an2949ej0110_rzt1_fa-coder.pdf
         ├──r01an2949jj0110_rzt1_fa-coder.pdf
         ├──r01uh0621ej0130_rzt1_fa-coder.pdf
         └──r01uh0621jj0130_rzt1_fa-coder.pdf
```

The file structures of "RZ_T1_fac.zip" are shown in Figure 2.1 File Structure.

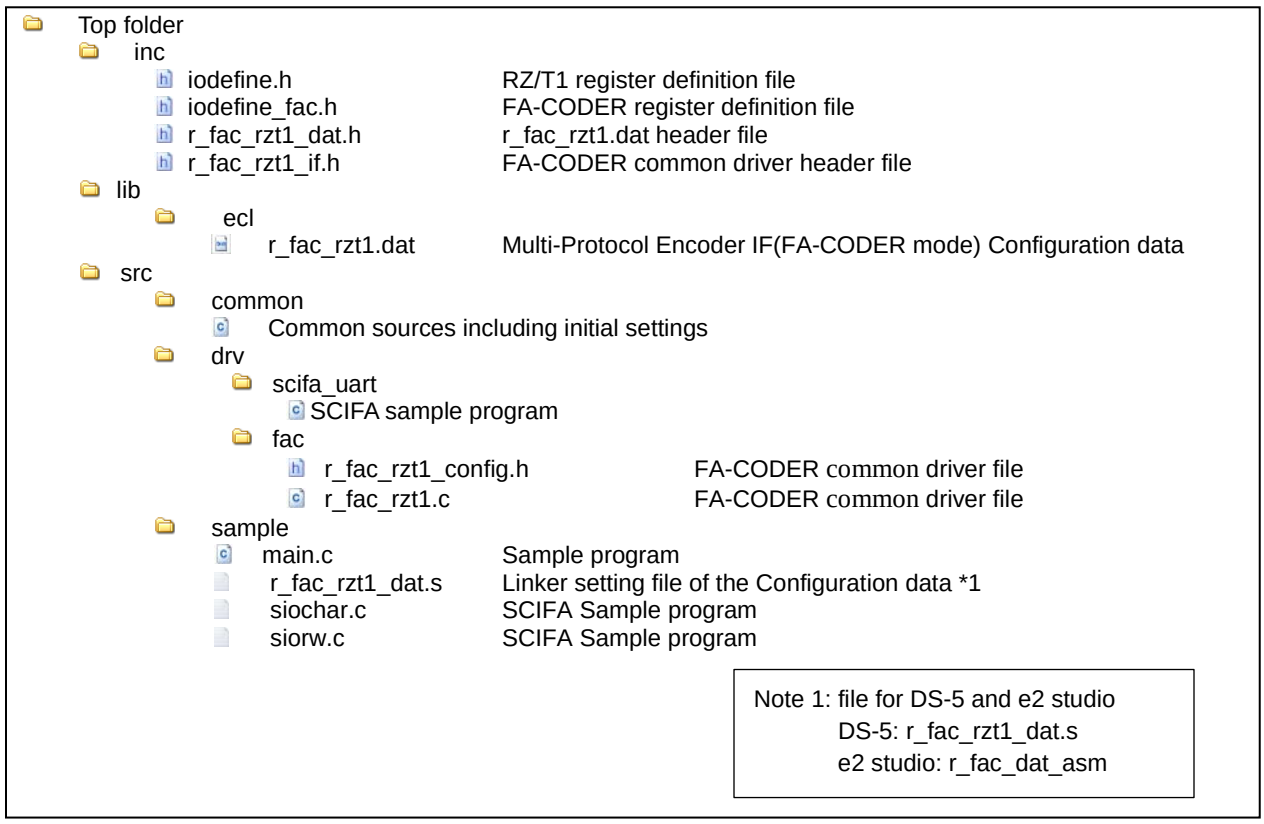


Figure 2.1 File Structure

3. Information about FA-CODER sample program

This chapter describes information to use a set of FA-CODER sample driver.

3.1 Software information

3.1.1 Operating System

This software is independent from operating system.

3.1.2 Memory footprint

Section name			Memory Size		
			IAR [bytes]	DS-5 [bytes]	e2 studio [bytes]
FA-CODER sample driver	Code		1352	2020	3904
	Data (with initial value)		0	28	0
	Data (without initial value)		78	52	80
	Constant Data		40	40	44
	Stack size of function	R_FAC_Open	44	44	72
		R_FAC_Close	20	28	20
		R_FAC_Control	20	48	72
		R_FAC_GetVersion	0	0	0
		fac0_int_isr	112+n *1	56+n *1	64+n *1
FA-CODER Configuration data	Code		0	0	0
	Data (with initial value)		0	0	0
	Data (without initial value)		0	0	0
	Constant Data		9836	9836	9836
Sample program	Code		3368	4556	6928
	Data (with initial value)		80	55	40
	Data (without initial value)		268	256	273
	Constant Data		1780	129	1766

Note 1. "n" is the Maximum stack size of user defined callback functions that are registered to R_FAC_Control function.

3.2 Hardware information

3.2.1 Device

RZ/T1

3.2.2 Target Board

(1) Board name
RZ/T1 CPU Board (RTK7910018C00000BE)

(2) Settings of CPU Board

SW4-1: ON

SW4-2: ON in case of serial flash memory is used, OFF in case of NOR flash memory is used

SW4-3: ON

SW4-4: ON

SW4-5: ON

SW4-6: OFF

JP2: 2-3 Connect

JP7: 1-2 Connect

3.3 Procedure on Development Environments

3.3.1 Preparation for the execution of the sample program

This sample program communicates with the PC. And for setting the PC, please refer to 6.1.2 Preparations of "RZ/T1 Group FIFO Integrated Serial Communication Interface (SCIFA) Application Note".

3.3.2 EWARM from IAR systems

- Build environment

  IAR Embedded Workbench for ARM v7.80.2

- Execution environment

  I-jet

- How to build sample program

  1. Extract files from RZ_T1_fac.zip and copy the files to arbitrary holder
  2. Copy the following files of "RZ/T1 Encoder I/F Configuration Library" (for IAR EWARM) to each folder

     lib\ecl\r_ecl_rzt1.a

     inc\r_ecl_rzt1_if.h
  3. Launch EWARM
  4. Select [File]menu -> [Open] -> [Workspace]
  5. Open RZ_T1_fac\RZ_T1_fac_****_boot.eww

NOR version	RZ_T1_fac_nor_boot.eww
Serial Flash version	RZ_T1_fac_serial_boot.eww

  6. Select [Project]menu -> [Rebuild all]

     Following file is generated.

     RZ_T1_fac\Debug\Exe\RZ_T1_fac_****_boot.out

NOR version	RZ_T1_fac_nor_boot.out
Serial Flash version	RZ_T1_fac_serial_boot.out

- How to execute sample program

  After executing "How to build sample program", connect the target board and the debugger properly, and execute the following operations.

  1. Select [Project] menu-> [Download and Debug]
  2. Select [Debug] menu-> [Go]

- Execution result of sample program

  After executing a sample program, input the command to "Terminal I/O" window.
  Please refer to RZ/T1 Group A-format Sample Program Application Note about the command.

3.3.3 DS-5 from ARM

- Build environment

  ARM Development Studio 5 (DS-5) Version 5.25.0

  ARM Compiler 5.06 update 3

- Execution environment

  ULINK2 (v2.01)

- How to build sample program

  1. Extract files from RZ_T1_fac.zip and copy the files to arbitrary holder
  2. Copy the following files of "RZ/T1 Encoder I/F Configuration Library" (for ARM DS-5) to each folder

     lib\ecl\r_ecl_rzt1.a

     inc\r_ecl_rzt1_if.h
  3. Launch DS-5
  4. Select [Window]menu -> [Show View] -> [Project Explorer]
  5. Click right button on [Project Explorer]view and then select [Import] of popup menu
  6. Select [General] -> [Existing Projects into Workspace] of [Import] dialog and then click [Next] button
  7. Click [Browse…] of [Import] dialog
  8. Select holder (the arbitrary holder of procedure 1 above) in [Browse For Folder] dialog and then click [OK].
  9. Select [Copy projects into workspace] of [Import] dialog
  10. Click [Finish] of [Import] dialog
  11. Select [Project] menu -> [Build All]

      Following file is generated.

      Debug\RZ_T_nor_sample.axf

      (In case of serial flash, use the "RZ_T_sflash_sample.axf" instead of the "RZ_T_nor_sample.axf")

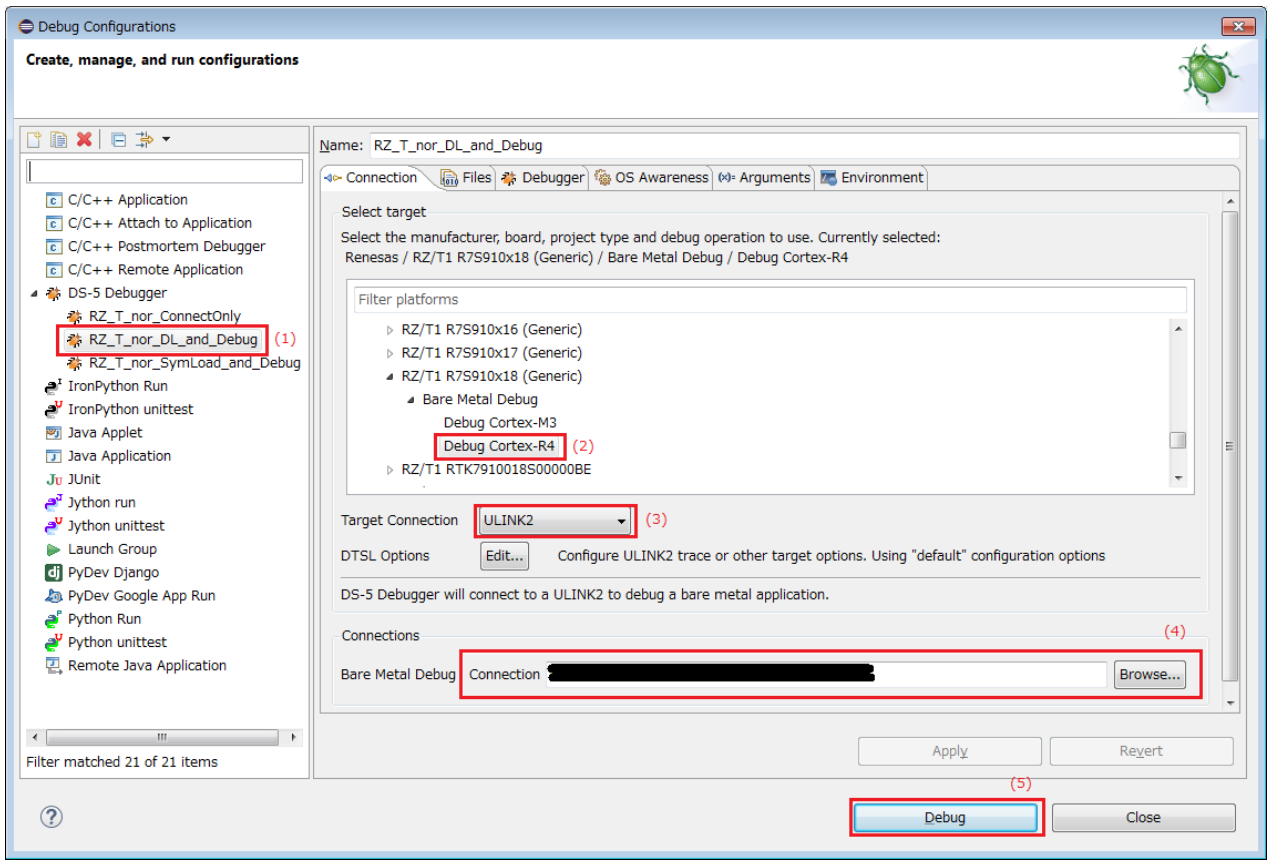
➢ How to execute sample program

After executing "How to build sample program", connect the target board and the debugger properly, and execute the following operations.

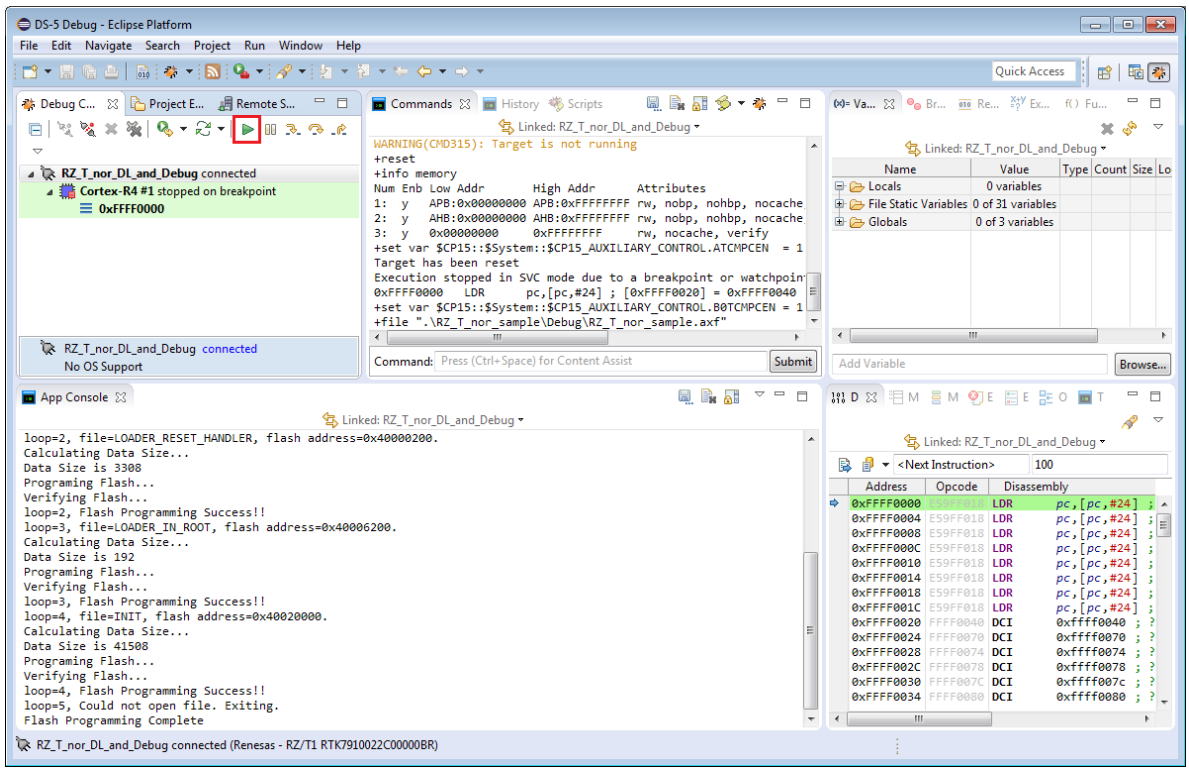
1. Open the debug configuration from the [Run] -> [Debug Configurations...], select the configuration window for "RZ_T_nor_DL_and_Debug". (In case of serial flash, use the "RZ_T_sflash_DL_and_Debug" instead of the "RZ_T_nor_DL_and_Debug")

   Select "Debug Cortex-R4" of "RZ/T1 R7S910x18 (Generic)" in [Select target].

   Select the ULINK2 of [Target Connection] in [Connection] tab, click on [Browse] and select the target connection from the list in the window. Click on [Debug] in the debug configurations window and start debugging.



2. On completion of writing to the flash memory by the script, the message "Flash Programming Complete" appears in the application console window. Debugging can then start.



➢ Execution result of sample program

After executing a sample program, input the command to "Terminal I/O" window.

Please refer to “RZ/T1 Group FA-CODER Sample Program Application Note” about the command.

3.3.4 e2 studio from RENESAS

- Build environment

  RENESAS e2 studio 5.2.0.020

  KPIT GNUARM-NONE-EABI Toolchain v16.01

- Execution environment

  J-Link BASE

- How to build sample program

  1. Extract files from RZ_T1_fac.zip and copy the files to arbitrary holder
  2. Copy the following files of "RZ/T1 Encoder I/F Configuration Library" (for KPIT GCC) to each folder

     lib\ecl\r_ecl_rzt1.a

     inc\r_ecl_rzt1_if.h
  3. Launch the e2studio
  4. Select [Window]menu -> [Show View] -> [Project Explorer]
  5. Click right button on [Project Explorer]view and then select [Import] of popup menu
  6. Select [General] -> [Existing Projects into Workspace] of [Import] dialog and then click [Next] button
  7. Click [Browse...] of [Import] dialog
  8. Select holder (the arbitrary holder of procedure 1 above) in [Browse For Folder] dialog and then click [OK].
  9. Select [Copy projects into workspace] of [Import] dialog
  10. Click [Finish] of [Import] dialog
  11. Select [Project] menu -> [Build All]

      Following file is generated.

      HardwareDebug\ RZ_T_nor_sample.x

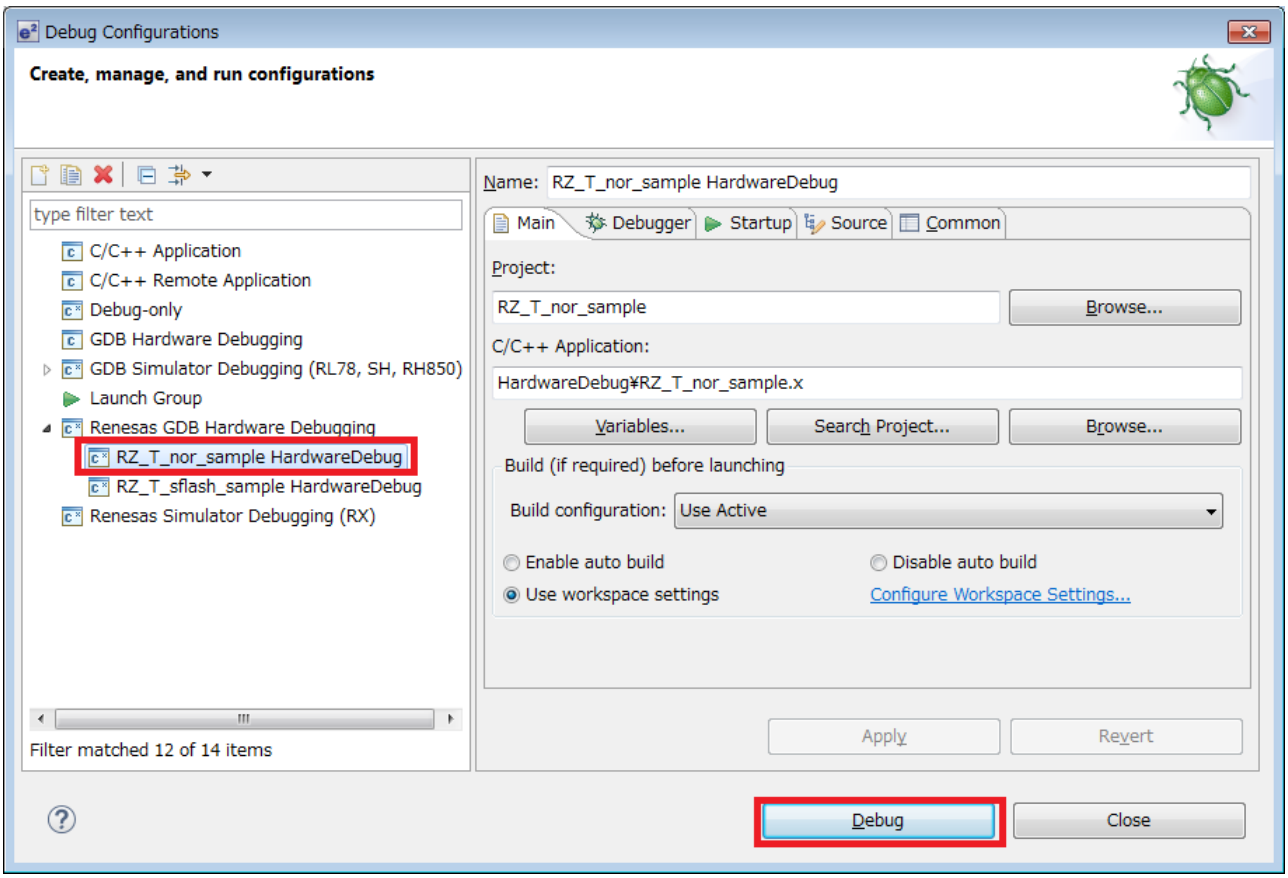
      (In case of serial flash, use the "RZ_T_sflash_sample.x" instead of the "RZ_T_nor_sample.x")

➢ How to execute sample program

After executing "How to build sample program", connect the target board and the debugger properly, and execute the following operations.

1. Select [Run] from the [Project] menu and then select [Debug Configurations].
2. Select the [RZ_T_nor_sample_HardwareDebug] in the following screen. Click the [Debug] and start the download to flash memory.

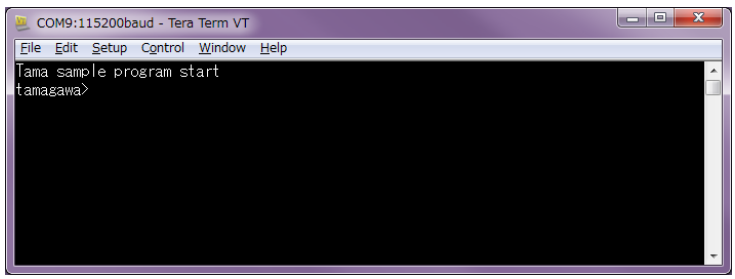
   (In case of serial flash, use the [RZ_T_sflash_sample_HardwareDebug] instead of the [RZ_T_nor_sample_HardwareDebug])



3. Click the [Resume] from the [Run] to start execution of the sample program.

➢ Execution result of sample program

After executing a sample program, input the command to "Terminal I/O" window.
Please refer to RZ/T1 Group FA-CODER Sample Program Application Note about the command.



4. Restriction

None.

5. Note

Available time for user processing of Encoder I/F FA-CODER sample program in a control loop is as follows.

Please confirm that there are no problems in your environment.

The example of the case that the control cycle is 62.5us is indicated below.

The time used by the sample program is about 8.1 us (13%) of 62.5us, and available time for user processing is about 54.4 us (87%).

Processing		Time		Occupancy rate
FA-CODER sample processing *2	Time setting registers for transmission	about 4.6 us	about 8.1us	13%
	Interrupt time	about 3.5 us		
Available time for user processing		about 54.4 us *1		87%

Note 1. Communication time with the encoder (when the number of data fields is three) is 35us of available time for user processing. For more information, refer to the "RZ/T1 Group FA-CODER Interface (FAC) User's Manual".

Note2. Initial setting time is not included.