RH850/U2A-EVA

Using OTS

Introduction

The RH850/U2A-EVA series offers an on-chip temperature sensor. This can be used to operate the chip in an optimal temperature range or in an approved temperature range. This document provides a general introduction to on-chip temperature sensor (OTS) functions. It also includes OTS configuration procedures and the sample software.

Target Device

This document describes the on-chip temperature sensor sample software for the RH850/U2A-EVA series.

In this document the RH850/U2A16 device R7F702300BFABA-C is employed to implement the example application. However, the concepts described in this document can be applied to all other members of the RH850/U2A-EVA series.

The RH850/U2A-EVA series has following variants:

RH850/U2A-EVA	FBGA-516	R7F702Z19AEDBG
		R7F702Z19BFDBG
RH850/U2A16	FBGA-516	R7F702300EBBG-C
		R7F702300AEBBC-C
		R7F702300BEBBC-C
	FBGA-373	R7F702300EBBB-C
		R7F702300AEBBB-C
		R7F702300BEBBB-C
	FBGA-292	R7F702300EABA-C
		R7F702300AFABA-C
		R7F702300BFABA-C
RH850/U2A8	FBGA-373	R7F702301EBBA-C
		R7F702301AEBBA-C
		R7F702301BEBBA-C
	FBGA-292	R7F702301EABG-C
		R7F702301AFABG-C
		R7F702301BFABG-C
RH850/U2A6	FBGA-292	R7F702302FABB-C
	HLQFP-176	R7F702302FAFK-C
	FBGA-156	R7F702302FABD-C
	HLQFP-144	R7F702302FAFM-C

Disclaimer

Renesas Electronics does not warrant the information included in this document. You are fully responsible for incorporation of these circuits, software, and information in the design of your equipment and system. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.

Contents

Introduction 1

Target Device 1

Disclaimer 1

Contents 2

Table of Tables 4

Table of Figures 5

1. Background 6

2. Reference Documents 7

2.1 RH850/U2A-EVA Group User's Manual: Hardware 7

2.2 RH850/U2A16 Starter Kit 7

3. Definitions and Abbreviations 8

4. Software and Hardware Tools 9

4.1 Development Tools 9

4.2 Emulator E2 9

4.3 RH850/U2A-EVA Evaluation Boards 9

4.3.1 Starter Kit 9

5. General information on OTS 10

5.1 OTS Features 10

5.2 OTS Interrupt & Error conditions 10

6. SW procedure and module setup for OTS applications 12

6.1 OTS Configuration 12

6.1.1 OTS Setup 12

6.1.2 Use of temperature ranges 12

6.1.3 Use of temperature deviations 12

6.2 Evaluation & Processing 13

6.3 Register description 15

7. OTS Calculation Tool 17

7.1 Sheet: MainView 17

7.2 Sheet: DetailedView 17

8. Description of Sample Software 19

8.1 Application description and appearance 19

8.1.1 User LEDs 19

8.1.2 LED Ring 19
8.1.3 Software procedure 19
8.2 Port and clock setup for TAUD, MSPI and UART 19
8.2.1 Port Setup - External Input / Output 19
8.2.2 Port Write Protection 20
8.2.3 Clock Setup for TAUD, MSPI and RLIN/UART Module 20
8.3 Sample Software 21
8.3.1 main_pe0.c 21
8.3.2 main_pe2.c 21
8.3.3 main_pe3.c 21
8.3.4 r_mspi.c and r_mspi.h 22
8.3.5 r_led.c and r_led.h 22
8.3.6 r_tau.c and r_tau.h 22
8.3.7 r_port.c and r_port.h 23
8.3.8 r_rlin.c and r_rlin.h 23
8.3.9 r_uart.c and r_uart.h 23
8.3.10 r_ots.c and r_ots.h 23

Revision History 25

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products 1

Notice 1

Table of Tables

Table 5-1 Interrupt list incl. function and number 10
Table 5-2 Conditions for OTABE and INTOTULI generation 11
Table 6-1 Register description for OTS 15
Table 8-1 Functions of User LED4 and LED5 19
Table 8-2 Configured temperature range 19
Table 8-3 External Input / Output and the Corresponding Pin Function of TAUD, MSPI and UART 20
Table 8-4 Alternative Peripheral Function Selection 20

Table of Figures

Figure 5-1 Interrupt & Error behavior at temperature change 10
Figure 7-1 Excel tool MainView sheet 17
Figure 7-2 Excel tool DetailedView sheet 18

1. Background

As much more complicated automotive applications and corresponding functional safety are required in the future, RH850/U2A-EVA devices provide an on-chip temperature sensor to monitor and control the operating temperatures of an application.

This application note provides a general description of the OTS, configuration and operating procedures, and a software example. This software example also accesses other peripherals such as the Timer Array Unit D (TAUD), the Multichannel Serial Peripheral Interface (MSPI), and the LIN/UART Interface (RLIN3).

Section 2 lists the reference documents. For detailed information or specific use cases, please use this application note in combination with the references.

Section 3 contains a list of abbreviations used in this document.

Section 4 contains a description of the hardware and software environment of the sample software for the application with on-chip temperature sensor.

Section 5 provides general information about the OTS.

Section 6 describes the SW procedures of the OTS that are used for configuration and usage. This sections also contains associated register descriptions.

Section 7 describes the corresponding excel calculation tool.

Section 8 explains the sample software by describing its appearance, the peripherals used, and the associated software.

2. Reference Documents

This Section contains information about the referenced documents.

2.1 RH850/U2A-EVA Group User’s Manual: Hardware

The Hardware User’s Manual provides information about the functional and electrical behavior of the device.

At the release time of this document the following manual version is available:

- RH850/U2A-EVA Group User’s Manual: Hardware (R01UH0864EJ0140, Rev.1.40)

It’s recommended to use the corresponding document or other later versions.

2.2 RH850/U2A16 Starter Kit

The Starter Kit User’s Manual provides information about the RH850/U2A16 Starter Kit and peripheral circuits.

At the release time of this document the following manual version is available:

- RH850/U2A16 Starter Kit User's Manual: Hardware (R20UT4861ED0300, Rev.3.00)

It’s recommended to use the corresponding document or other later versions.

3. Definitions and Abbreviations

For the purposes of this publication, the following abbreviations for common terms apply:

- **MSPI3:** Multichannel Serial Peripheral Interface 3
- **TAUD0:** Timer Array Unit D 0
  - **TAUD0C0:** Timer Array Unit D 0 Channel 0
  - **TAUD0C2:** Timer Array Unit D 0 Channel 2
- **RLIN3:** Renesas Local Interconnect Network 3
  - **RLIN39:** Renesas Local Interconnect Network 3 Channel 9
  - **UART:** Universal Asynchronous Receiver Transmitter
- **PE0:** Processor Element 0
- **PE1:** Processor Element 1
- **PE2:** Processor Element 2
- **PE3:** Processor Element 3
- **OTS:** On-Chip Temperature Sensor
- **CRAM:** Cluster RAM
  - **CRAM#1:** Cluster RAM in Cluster #1

4. Software and Hardware Tools

This section contains the information about tools which are used in this document to implement the OTS application.

4.1 Development Tools

The compiler used for the sample code is Green Hills Multi.

The required version of the GHS compiler for U2A-EVA devices is V7.1.6/2019.1.5 or later.

The EXEC, 850eserv2 and device files are also required before compiling the application code. For further information, please refer to the *Getting started software package.*

4.2 Emulator E2

The Renesas E2 Emulator is required for the debugging.

4.3 RH850/U2A-EVA Evaluation Boards

The following evaluation board is used in this document to implement the OTS application.

4.3.1 Starter Kit Board

The Starter Kit Board Version 3, Y-ASK-RH850-U2A16 carries the target device. The SW in this document is performed on the RH850/U2A16 device R7F702300BFABA-C within this this Starter Kit Board. The software uses the User LEDs and the USB interface of the starter kit.

5. General information on OTS

This section provides general information on understanding of the OTS.

5.1 OTS Features

The RH850/U2A-EVA group offers an on-chip temperature sensor. This can be used to monitor operating temperatures and keep them within an optimal or allowed range.

The on-chip temperature sensor provides one channel. The clock supply for operation and register access is the CLK_LSB.

In addition, the OTS has a data register and two different measurement modes. The single measurement mode performs a single measurement. The continuous mode performs continuous measurements.

The OTS provides three interrupts which can signal an exceeding of the limit values (1), the end of a measurement (2) and an exceeding of the maximum temperature change (3). The OTS features the following interrupts:

Table 5-1 Interrupt list incl. function and number

Number	Interrupt	Function	Interrupt no
1	INTOSTOTE	Maskable OTS error interrupt (EI Level)	EIINT657
2	INTOTSOTI	Maskable OTS end interrupt (EI Level)	EIINT658
3	INTOTSOTULI	Maskable OTS upper/lower limit interrupt (EI Level)	EIINT659

The OTS is also connected to the Error Control Module and can therefore also use or trigger functions of the ECM. For more information about the ECM, please refer to the Hardware User's Manual R01UH0864EJxxxx *Section 45 'Error Control Module (ECM)'* or Application Note R01AN6094EDxxxx *'Using CRG, PBG and ECM'*.

- OTABE as OTS abnormal temperature error with ECM error number 76

5.2 OTS Interrupt & Error conditions

Figure 5-1 contains the interrupt and error behavior of the OTS at given temperature changes.

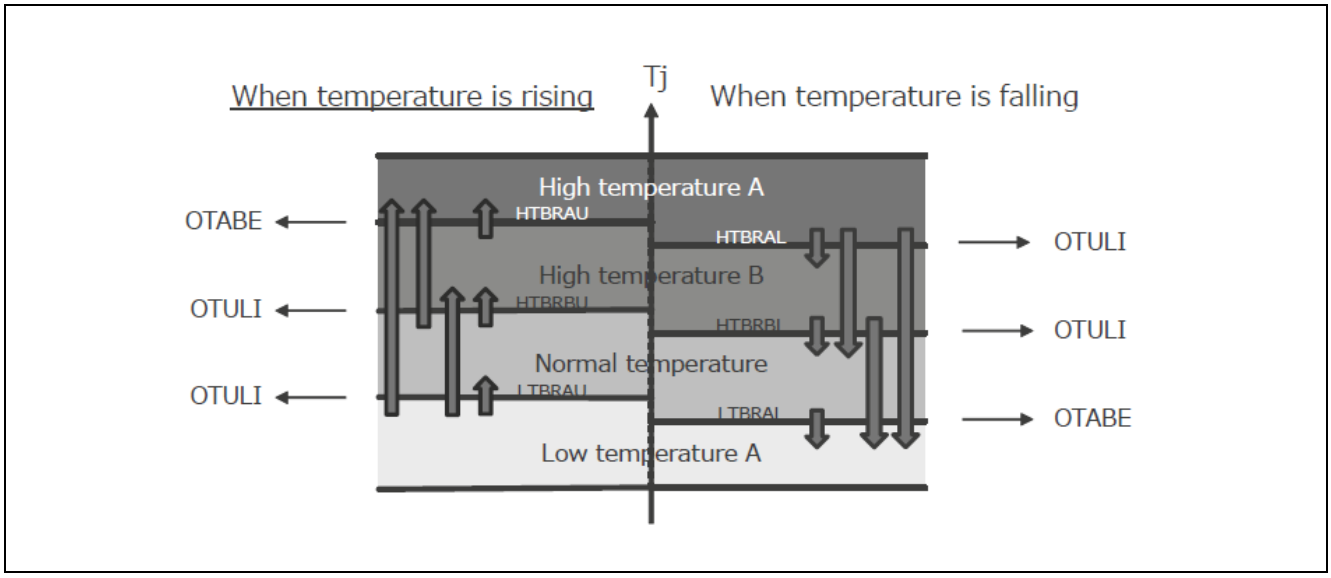


Figure 5-1 Interrupt & Error behavior at temperature change

Table 5-2 Conditions for OTABE and INTOTULI generation

		Subsequent status state			
		High temp. A	**High temp. B**	**Normal temp.**	**Low temp. A**
Previous status condition	**High temp. A**	-	INTOTSOTULI	INTOTSOTULI	OTABE
	High temp. B	OTABE	-	INTOTSOTULI	OTABE
	Normal temp.	OTABE	INTOTSOTULI	-	OTABE
	Low temp. A	OTABE	INTOTSOTULI	INTOTSOTULI	-

Table 5-2 lists the temperature changes and the generated interrupts and errors.

The INTOTSOTULI interrupt occurs when a configured temperature limit is exceeded with rising or falling measured temperature. The interrupt only occurs when a status change to another temperature range takes place. See Table 5-2 and Figure 5-1 for more details about the configuration.

The INTOTSOTE interrupt occurs when the configured temperature difference limit of the diagnostic function is exceeded.

For more details on the configuration of the OTS and the associated functions, please refer to Section 6.1 ‘OTS Configuration’.

6. SW procedure and module setup for OTS applications

This section describes the software procedures, functional configurations, and calculation descriptions.

6.1 OTS Configuration

This section deals with basic and functional configuration.

6.1.1 OTS Setup

The setup process is as follows:

To avoid malfunctions, the OTS should be set in idle mode.

1. To check this, the OTEND bit in OTS0OTENDCR should be set to 1
2. The OTACT bit in OTS0OTFR should be checked for an operation (0 in the idle state).

If the sensor is idle, the configuration should take place via the OTS0OTCR register. The register can configure the operations mode to one or continuous measurements, enable the diagnostic mode, enable the temperature alarm error, enable the temperature change interrupt, and enable the temperature sensor error interrupt.

Provided that the corresponding functions have been activated before, the corresponding modes should be configured now (see section 6.1.2 ‘Use of temperature ranges’ & 6.1.3 ‘Use of temperature deviations’). If there are no further functions, this step can be skipped.

To avoid the occurrence of unwanted interrupts the interrupt flags should be cleared by writing the value 05_H to the OTFCR register.

6.1.2 Use of temperature ranges

The configuration should be done as described in section 6.1.1 ‘OTS Setup’.

In addition, the configuration of the border registers is required. Writing to the OTS0HTBRmn registers configures the various borders.

The size of the values of the registers must be configured as follows to avoid unexpected behavior:

OTS0HTBRAU > OTS0HTBRAL > OTS0HTBRBU > OTS0HTBRBL > OTS0LTBRAU > OTS0LTBRAL

$$Temperature_H = \frac{-B + \sqrt{\left(B^2 - 4 * A * \left(C - (273.15 + value)\right)\right)}}{2 * A} * 2^{15}$$

Legend/Key:

- A/B/C: Coefficient A/B/C
- Value: Temperature value [°C]
- 273,15: Zero-point correction to 0°C
- 2^{15}: Decimal point correction

For more details about the configurable temperature ranges and the corresponding interrupt behavior, see also Figure 5-1.

6.1.3 Use of temperature deviations

The configuration should be done as described in section 6.1.1 ‘OTS Setup’.

It should be ensured that the OTS is configured in continuous mode and diagnostic mode. For this purpose, it should be ensured that the bits OTEE (bit 4), SDE (bit 1) and OTMD (bit 0) in the OTS0OTCR register are each set to 1.

In addition, the configuration of the differential register is required. The configuration of the maximum temperature deviation is done by the OTS0TDLR register.

The calculation of the difference value is like the calculation of the temperature values. In this case, the temperature calculation is based on the absolute zero point since temperature differences are calculated in Kelvin.

$$Temperaturedifference_H = \frac{-B + \sqrt{\left(B^2 - 4 * A * (C - value)\right)}}{2 * A} * 2^{15}$$

Legend/Key:

- A/B/C: Coefficient A/B/C
- Value: Temperature difference value [K]
- 2^{15}: Decimal point correction

The INTOSTOTE occurs when the configured temperature difference is exceeded. The temperature difference is exceeded as soon as the amount of the difference between the previous measurement and the latest measurement is exceeded. Condition:

- OTS0TDLR < | present temperature value – previous temperature value |

Otherwise, no sensor error interrupt is generated. Condition:

- OTS0TDLR ≥ | present temperature value – previous temperature value |

6.2 Evaluation & Processing

For the temperature calculation, the coefficients A, B, C and the temperature value X are used before the correction. The coefficients A, B, C must be read out once and can be stored for later conversions. To obtain an actual value, the temperature value should be read out after each measurement.

Calculation formula:

$$T[K] = A * X^2 + B * X + C$$
$$T[°C] = A * X^2 + B * X + C - 273.15_D$$

- Temperature in Kelvin: T[K]
- Temperature in degree Celsius: T[°C]
- Coefficient A/B/C: A/B/C OTS0COEFFRn (n = A/B/C)
- Temperature coefficient X: X OTS0OTDR

The following steps must be performed for the calculation:

1. Conversion from hexadecimal to decimal.
2. Decimal point correction:
    a. Coefficients A/B/C: $\frac{n_{A|B|C}}{2^4}$
    b. Temperature Coefficient X: $\frac{X}{2^{15}}$
3. Insert into the calculation formula for the temperature value in the unit Kelvin.

4. Subtract 273.15 Kelvin to get a value in degrees Celsius.

Example calculation:

- Example values:
  - Coefficient A: OTS0COEFFRA = $1F65_H$ = 8037_D
  - Coefficient B: OTS0COEFFRB = $2D42_H$ = 11586_D
  - Coefficient C: OTS0COEFFRC = 0070_H = 112_D
  - Temperature Coefficient X: OTS0OTDR = $2B25_H$ = 11045_D
- Decimal point correction:
  - A = $\frac{8037_D}{2^4} = 502{,}3125$
  - B = $\frac{11586_D}{2^4} = 724{,}125$
  - C = $\frac{112_D}{2^4} = 7$
  - X = $\frac{11045_D}{2^{15}} = 0{,}337067$
- Calculation:

$$T[K] = 502.3125_D * 0.337067_D{}^2 + 724.125_D * 0.337067_D + 7_D - 273.15_D$$
$$\boldsymbol{T[K] \approx 308.15K}$$
$$T[°C] = T[K] - 273.15$$
$$\boldsymbol{T[°C] \approx 35°C}$$

The Excel tool can also be used for the temperature calculation. For more details refer to section 7 ‘OTS Calculation Tool’.

6.3 Register description

Table 6-1 Register description for OTS

Register	Function	Function (Bit-specific)	Bit Name	Bit Position
OTS00OTSTCR	Start temperature measurements	1: Start measurement	OTST	0
OTS0OTENDCR	Stop temperature measurements	1: Stop measurement	OTEND	0
OTS0OTCR	Configure temperature sensor features and modes	Temperature sensor error 1: Enable, 0: Disable	OTEE	4
		Temperature rise/drop interrupt 1: Enable, 0: Disable	OTULIE	3
		Temperature alarm error 1: Enable, 0: Disable	OTABEE	2
		Diagnosis mode 1: Enable, 0: Disable	SDE	1
		Temperature measurement mode: 1: Continuous measurement 0: Single measurement	OTMD	0
OTS0OTFCR	Clear temperature sensor flags	Diagnosis Error flag: 1: Clear flag	SDERC	2
		Temperature measurement end flag: 1: Clear flag	OTFC	0
OTS0OTFR	Read temperature sensor flags	Diagnosis error flag: 1: Diagnosis error occurred 0: No diagnosis error occurred	SDER	2
		Temperature sensor status: 1: Ongoing operation 0: Sensor on idle	OTACT	1
		Temperature measurement end flag: 1: Measurement completed 0: Measurement not completed	OTF	0
OTS0OTSTR	Indicator for the temperature status	11: Low temperature A 10: High temperature A 01: High temperature B 00: Normal temperature	TSTAT	1,0
OTS0OTDR	Includes the conversion result	Bit 15: Sign bit (always 0) The decimal point is located between bit 14 & 15	OTD	15 to 0
OTS0HTBRm[1]n[2]	Configure the High temperature limits	Bit 15: Sign bit (always 0) Specifies the high temperature limit m[1]n[2]	HTBm[1]n[2]	15 to 0
OTS0LTBRAn[2]	Configure the low temperature limits	Bit 15: Sign bit (always 0) Specifies low temperature border n[2]	LTBAn[2]	15 to 0
OTS0TDLR	Configure the allowed temperature difference between the measurements	Bit 15: Sign bit (always 0) The decimal point is located between bit 14 & 15	TDL	15 to 0

OTS0COEFFRn[3]	Reading in the temperature coefficients for the calculation of the temperature values	Bit 15: Sign bit (always 0) Value of the coefficient with decimal point between bit 3 & 4	COEFF[3]	15 to 0
OTS0SDIAGCTL	Configure the ADC input for self-diagnosis	11: Prohibited 10: Pull-up 01: Pull-down 00: No pull-up/-down (normal Operation)	ADCIN	1,0

Notes: 1. Index: m=A or m=B
A: Limit register A (used for definition of high temperature A)
B: Limit register B (used for definition of high temperature B)
2. n=U or n=L
U: Upper limit value with increasing temperature
L: Lower limit value with decreasing temperature.
3. n=A, B, C
A, B, C: Temperature coefficient A, B, C

For the detailed information of the registers which are mentioned in this section refer to the RH850/U2A-EVA Hardware User's Manual *R01UH0864EJxxxx Section 12.3 'Temperature Sensor (OTS)'*.

7. OTS Calculation Tool

This section describes the Excel calculation tool.

The Excel calculation tool can be found in the attachment file: *r01an6575ed0100-rh850u2a-CalculationTool.xlsx*

7.1 Sheet: MainView

The sheet “MainView” serves as a starting point for the temperature calculation from the OTS measured values. Figure 7-1 contains a screenshot of an example view of the MainView of the tool.

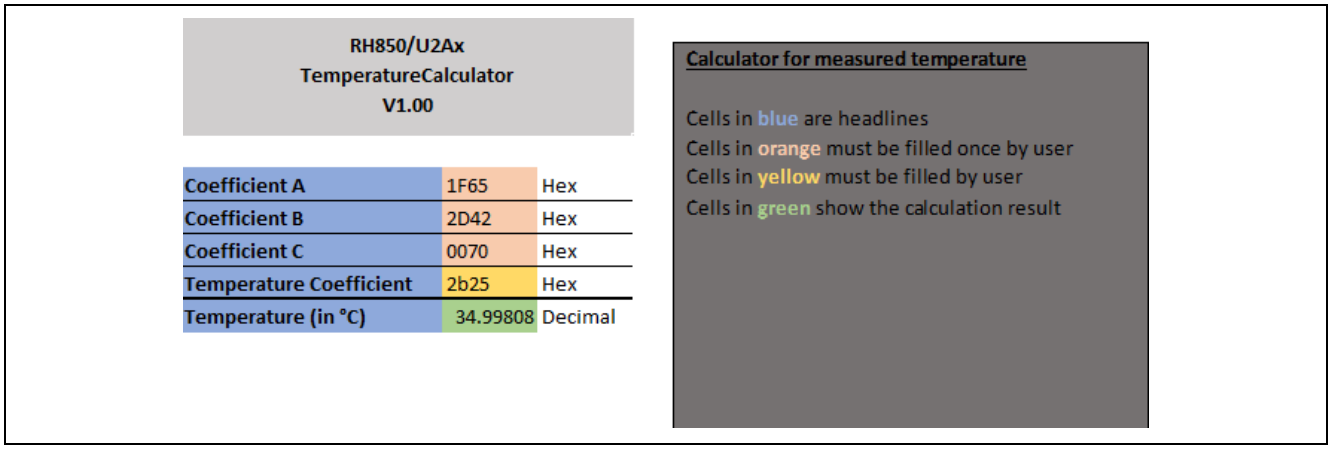


Figure 7-1 Excel tool MainView sheet

The sheet has information about the tool in the light gray box. The dark gray box on the right side gives information about the different cell colors. The calculation can be performed with the spreadsheet.

The input values are intended for the direct input of the hexadecimal register values. The output value is made in decimal format and output in degrees Celsius.

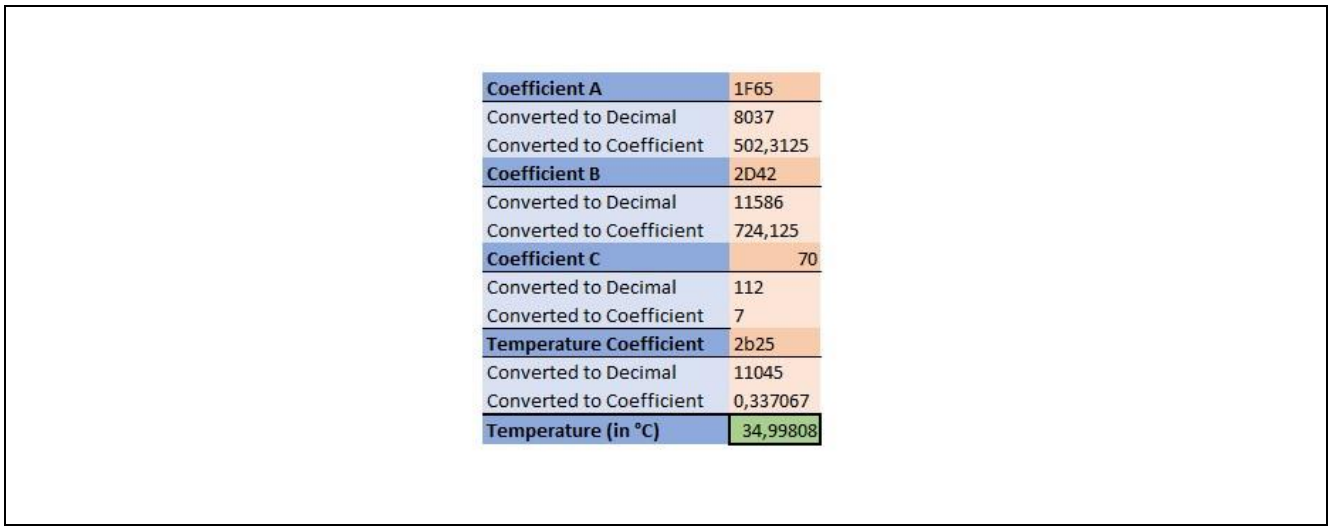
The form is to be used as follows:

- The orange fields must be filled in once with the coefficient values at the beginning of use. Since these values are set at the factory, the value must be read only once.
- The yellow fields must be re-entered after each measurement to get back an actual value.
- The green field represents the converted result in degrees Celsius.
- The blue fields indicate what must be entered in the associated field or what the field outputs.

7.2 Sheet: DetailedView

The sheet “DeatiledView” provides a more detailed view of the calculation and performs the calculations of the Mainview. Corresponding calculation formulas can be viewed in the cells of the DetailedView. For more details on the calculations, see also section 6.2 ‘Evaluation & Processing’.

Figure 7-2 contains a screenshot of an example view of the DetailedView of the tool.



Coefficient A	1F65
Converted to Decimal	8037
Converted to Coefficient	502,3125
Coefficient B	2D42
Converted to Decimal	11586
Converted to Coefficient	724,125
Coefficient C	70
Converted to Decimal	112
Converted to Coefficient	7
Temperature Coefficient	2b25
Converted to Decimal	11045
Converted to Coefficient	0,337067
Temperature (in °C)	34,99808

Figure 7-2 Excel tool DetailedView sheet

The color coding of the cells is the same as that of the cells in the MainView sheet.

8. Description of Sample Software

This section contains information about the behavior and appearance of the sample software application. It also contains information about the port and clock settings of the TAUD, MSPI and RLIN used. At the end of this section the C files and header files used are described.

8.1 Application description and appearance

This section contains information about the behavior and appearance of the application.

8.1.1 User LEDs

The user LEDs indicate whether the application is started and in which rhythm the OTS measurements are triggered.

Table 8-1 Functions of User LED4 and LED5

LED	Function
LED4	Controlled by TAUD0 channel 0 interrupt routine and flashes every second while the application is running. At each status change of the LED a temperature measurement is performed (~ every second).
LED5	Controlled by TAUD0 channel 2 output and flashes every half second.

8.1.2 LED Ring

The LED ring provides a short startup animation of a few seconds, with the upper and lower halves of the LEDs flashing alternately. 5 status changes are executed until the LEDs are switched off and the application starts.

8.1.3 Software procedure

The LED ring provides a short startup animation of a few seconds. The SW prints the same texts in debug and UART terminal. At the beginning the application displays a start text. The temperature coefficients are output afterwards. After that an interval timer starts with the time limit of 0.5 seconds. After these 0.5 seconds a temperature measurement is started and after the measurement is finished a corresponding text with temperature value is calculated and output. This procedure is repeated until the application is stopped.

The example SW uses temperature range functions and outputs corresponding texts when the range is changed. Table 8-2 lists the configured temperature ranges.

Table 8-2 Configured temperature range

Border register	Value
OTS0HTBRAU	104°C
OTS0HTBRAL	103°C
OTS0HTBRBU	44.5°C
OTS0HTBRBL	44°C
OTS0LTBRAU	1°C
OTS0LTBRAL	0°C

8.2 Port and clock setup for TAUD, MSPI and UART

This section describes the connection settings for the TAUD, the RLIN/UART and the MSPI used in the Guard application.

8.2.1 Port Setup - External Input / Output

Table 8-3 lists the external input / output pins of the TAUD, MSPI and RLIN/UART modules on the RH850/U2A16 device, which are used for this Guard application.

Table 8-3 External Input / Output and the Corresponding Pin Function of TAUD, MSPI and UART

Signal Name	Description	Corresponding Pin	Port Number	Port Function	I/O Direction
TAUD0TTOUT2	Channel 2 output	P2	2	Alternative function 6*1	Output
MSPI3SO	Output serial data signal	P22	0	Alternative function 4*1	Output
MSPI3SI	Input serial data signal		1		Input
MSPI3SC	Serial clock signal		4		Input / Output
P22_2	Latch Enable for LED module		2	Port Mode	Output
RLIN39RX	Output serial data signal	P6	2	Alternative function 4*1	Input
RLIN39TX	Input serial data signal		3		Output
P10_10	Output Enable for LED module	P10	10	Port Mode	Output

Notes: 1. The corresponding pins operate in alternative mode software I/O control, the pins operate as alternative functions, and the input / output direction is selected by setting PMn.PMn_m bit to 1 / 0. To configure the operation mode, the PMCn, PFCn, PFCEn and PFACEn registers should be configured. Table 8-4 provides an overview of the register settings.
For detailed information, please refer to the HW User's Manual *R01UH0864EJxxxx Section 2 'Pin Functions'*.

Table 8-4 Alternative Peripheral Function Selection

Alternative-Function	Bit Name			
	PMCn_m	PFCAEn_m	PFCEn_m	PFCn_m
Alternative function 1	1	0	0	0
Alternative function 2				1
Alternative function 3			1	0
Alternative function 4				1
Alternative function 5		1	0	0
Alternative function 6				1
Alternative function 7			1	0
Alternative function 8				1

8.2.2 Port Write Protection

Before the port registers described above can be written to, appropriate write processes must be enabled by the port key code protection register PKCPROT and by setting the port specific bit in the port write enable register PWE.

For details, please refer to the RH850/U2A-EVA Hardware User's Manual *R01UH0864EJxxxx Section 2 'Pin Functions'*.

8.2.3 Clock Setup for TAUD, MSPI and RLIN/UART Module

The supply clock for the MSPI, RLIN/UART and TAUD module is CLK_HSB with a maximum frequency of 80 MHz.

For detailed information on the clock supply and the corresponding settings, refer to the RH850/U2A-EVA Hardware User Guide R01UH0864EJxxxx or the RH850/U2A-EVA Application Note of Clock Supply R01AN4593EDxxxx_RH850_U2A.

To enable the clock connections to the peripherals, the MSR_TAUD.BIT.MS_TAUD_0, MSR_MSPI.BIT.MS_MSPI_3 and MSR_RLIN.BIT.MS_RLIN3_9 bits should be set to 0.

Before writing to this register, write protection must be disabled by setting A5A5A501H in the key code protection register MSRKCPROT. If the write protection is required in further accesses, please write A5A5A500H to the MSRKCPROT register after the clock setup.

8.3 Sample Software

The corresponding software example of this document can be found in the attachment file: *r01an6575ed0100_rh850_u2a_sw.7z*.

It is based on the software tools and hardware setups described in section 4.

The target device of the software is the RH850/U2A16 device R7F702300BFABA-C, the device files in the package DF-RH850U2A-EE_E130b.zip are used for the sample SW. Please notice that the latest device files are always recommended for the SW development.

The example SW is executed by core 0, 2 and 3. Core 2 and core 3 serve as a stress test to increase the measured temperatures. For this purpose, they write to the CRAM. Core 0 executes the main tasks. Core 0 configures the various hardware macros and components for their intended use. Core 0 also executes the interrupt routines.

8.3.1 main_pe0.c

The location of this file is in folder:
… \core0\src

- Main function: *int main(int argc, char *argv[])*
    - Enable required interrupts
    - Set port modes
    - Configure and start TAUD and MSPI
    - Start LED Ring boot animation
    - Set up user LEDs
    - Configure OTS
    - Setup UART
    - Print welcome message
- Print text on UART and Debug console *void PrintText(char* TextArray)*
- Generate time buffer *delay(int value)*
- Generates string for UART/console temperature output *void generate_string(void)*

8.3.2 main_pe2.c

The location of this file is in folder:
…\ core2\src

- Main function: *int main(int argc, char *argv[])*
    - Welcome message on debug terminal
    - Writes the cluster RAM #1 full (increases temperature)

8.3.3 main_pe3.c

The location of this file is in folder:
…\ core3\src

- Main function: *int main(int argc, char *argv[])*
    - Welcome message on debug terminal
    - Writes the cluster RAM #1 full (increases temperature)

8.3.4 r_mspi.c and r_mspi.h

The location of this file is in folder:
… \shared\additional_sources\peripherals\mspi

r_mspi.c contains the configuration for the basic operations for the LED driver operations of the Guard application:

- MSPI initialization & associated port setup: *R_MSPI_Init()*
- Enable MSPI, MSPI interrupts, Configure P10_10: *R_MSPI30_Start()*
- Send bitmask to LED driver and accept it as LED mask: *R_MSPI30_Send(uint16_t* data)*

r_mspi.h is the header file of r_mspi.c, in which the variables and functions are defined.

8.3.5 r_led.c and r_led.h

The location of this file is in folder:
…\ shared\additional_sources\modules\led

r_led.c contains the configuration for the basic operations for the LED driver operation of the Guard application based on the MSPI functions:

- Initialize ports for LED driver latch and output enable: *R_LED_RingGpioInit()*
- Initialize the LED driver and output: *R_LED_RingInit()*
- Turn off all LEDs: *R_LED_RingOutputDisable()*
- Turn on last LED Bitmask: *R_LED_RingOutputEnable()*
- Update LED Bitmask: *R_LED_RingUpdate(uint16_t* LedBitMask)*

r_led.h is the header file of r_led.c, in which the LED functions are defined.

8.3.6 r_tau.c and r_tau.h

The location of this file is in folder:
…\ shared\additional_sources\peripherals\tau

- Configuration and initialization of TAUD0C0, TAUD0C2: *R_TAUD_Init()*
    - TAUD0C0: ~1 sec interval timer
    - TAUD0C2: ~0.5 sec interval timer with real time output
- Start trigger TAUD0C0: *R_TAUDC0_Start()*
- Start trigger TAUD0C2: *R_TAUDC2_Start()*
- Stop trigger TAUD0C0: *R_TAUDC0_Stop()*
- Stop trigger TAUD0C2: *R_TAUDC2_Stop()*
- Switch TAUD0C0 interrupt routine: *void R_TAUD0_AssignChannel0ISR(uint8_t ISR_select)*
- TAUD0C0 ISR for boot up animation: *void R_Taud0_BootISR(void)*
    - Execute LED Ring animation
- TAUD0C0 ISR for continuous operation: *void R_Taud0_RegularISR(void)*
    - Execute user LED toggle and start OTS

- TAUD0C0 interrupt routine: void *taud0c0_interrupt(void)*
    - Execute selected ISR

r_tau.h is the header file of r_tau.c, where the functions for configuration and control of the TAUD are defined.

8.3.7 r_port.c and r_port.h

The location of this file is in folder:
…\ shared\additional_sources\peripherals\port

- Sets Port Protection (functional safety):

  void PORT_SetProtection(t_PORT_GROUP portNr, t_PORT_PROTECTION onOff)

- Configures alternative port function:

  void PORT_SetAF(t_PORT_GROUP portNr, t_PORT_PIN pinNr, t_PORT_AF afNr, t_PORT_DIRECTION dir)

- Configures port driver strength:

  void PORT_SetDriveStrength(t_PORT_GROUP portNr, t_PORT_PIN pinNr, t_PORT_DRIVE_LEVEL driveLevel)

r_port.h is the header file of r_port.c, in which the port functions are defined.

8.3.8 r_rlin.c and r_rlin.h

The location of this file is in folder:
…\ shared\additional_sources\peripherals\rlin

- Configuration of RLIN as UART: *void R_RLIN39_UartInit(void)*
- Send a UART string via RLIN: *void R_RLIN39_UartSendString(char_t send_string[])*
- Obtaining the transmission status of the RLIN39: *uint8_t R_RLIN39_GetStatus(void)*
- Transmission end interrupt *void EIINT453(void)*
    - Starts transmitting the next character if available

r_rlin.h is the header file of r_rlin.c, in which the configuration and control functions are defined.

8.3.9 r_uart.c and r_uart.h

The location of this file is in folder:
…\ shared\additional_sources\peripherals\uart

- Send a UART string: *void R_UART_SendString(char_t string[])*
    - Send a UART string as soon as RLIN39 is ready
- Reset the UART port configurations: *void R_UART_Deinit(void)*
- Configuration of UART: *void R_UART_Init(void)*
    - Configuration of ports and RLIN

r_uart.h is the header file of r_uart.c, in which the configuration and control functions of the UART are defined.

8.3.10 r_ots.c and r_ots.h

The location of this file is in folder:
…\ shared\additional_sources\peripherals\ots

- Configuration and Initialization Function of OTS0: *void R_OTS_Init(void)*

  - Single measurement
  - Diagnosis enabled
  - INTOTSOTULI enabled
  - INTOTSOTI enabled
  - INTOTSOTE enabled
  - OTABE enabled
- Configuration of temperature ranges: *void R_OTS_SetLimits(uint8_t limiter, double value)*
  - uint8_t limiter: Border register to be configured
  - double value: Limit value in degree Celsius
- Update coefficients/load into memory: *void R_OTS_UpdateCoefficients(void)*
- Get the OTS temperature register value: *uint16_t R_OTS_GetValue(void)*
- Start a temperature measurement: *void R_OTS_Start(void)*
- Stop ongoing temperature measurement: *void R_OTS_Stop(void)*
- Get measurement result converted to degree Celsius: *float R_OTS_GetConvertedResult(void)*
- Get coefficients A-C or X: *float R_OTS_GetParameter(uint8_t val)*
- Upper/Lower Limit exceed interrupt: *void INTOTSOTULI(void)*
  - Corresponding terminal output
- OTS error interrupt: *void INTOTSOTE(void)*
  - Corresponding terminal output
- OTS end interrupt: *void INTOTSOTI(void)*
  - Corresponding terminal output with converted result in degrees Celsius

r_ots.h is the header file of r_ots.c, in which the configuration and control functions of the OTS are defined.

Revision History

Rev.	Date	Description	
		Section	Summary
1.00	Jan.02.2024		First release

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

© 2021 Renesas Electronics Corporation. All rights reserved.