

# Renesas RA Family

## Azure RTOS TraceX for Azure RTOS ThreadX Debugging

### Introduction

Azure RTOS ThreadX is an RTOS from Microsoft Corporation which is based on a high-performance embedded kernel.

Azure RTOS TraceX is a Windows-based analysis tool. It provides embedded developers with a graphical view of real-time system events and enables them to visualize and better understand the behavior of their real-time systems.

This application note provides procedures to check Azure RTOS ThreadX thread and object states (referred to as resources) during the development of applications in e<sup>2</sup> studio. The procedure for starting Azure RTOS TraceX is also explained.

### Target Device

RA6M3 MCU Group (R7FA6M3AH)

### Operating Environment

Target Board	EK-RA6M3
IDE	e <sup>2</sup> studio version 2021-04 and FSP v3.0.0
Trace Tool	Microsoft Azure RTOS TraceX v6.1.6.0
OS	Microsoft Azure RTOS ThreadX v6.1.6
Toolchains	GNU Arm Embedded Toolchain: 9-2020-q2-update (GNU ARM Embedded 9.3.1.20200408)

Note: Please download and install tools from the following URL in advance.

- Quick Start Guide for e<sup>2</sup> studio for RA download site:  
[Quick Start Guide for e<sup>2</sup> studio for RA](#)
- FSP with e<sup>2</sup> studio installer download site:  
<https://github.com/renesas/fsp/releases>
- Microsoft Azure RTOS ThreadX user guide site:  
[About the Azure RTOS ThreadX Guide | Microsoft Docs](#)
- Microsoft Azure RTOS TraceX user guide site:  
[What is Azure RTOS TraceX? | Microsoft Docs](#)
- Microsoft Azure RTOS TraceX download site:  
[Get Azure RTOS TraceX - Microsoft Store](#)

**Contents**

1. Install FSP with e <sup>2</sup> studio.....	3
2. Install Azure RTOS TraceX.....	3
3. Creating a project in e <sup>2</sup> studio .....	3
4. FSP Configuration .....	6
4.1 Blinky Thread Settings .....	6
4.2 Build the project.....	8
5. Connect PC and EK-RA6M3 Board .....	9
5.1 Debugging setting.....	10
6. Using the RTOS Resource View.....	10
6.1 Displaying the RTOS Resources View.....	10
6.2 Context menu .....	11
6.3 Stack setting.....	12
6.3.1 Enable load stack data and set stack threshold.....	12
6.4 Tab menu .....	13
7. Start debugging a project with Azure RTOS TraceX .....	14
7.1 Launch debugger on e <sup>2</sup> studio .....	14
7.2 Launch Azure RTOS TraceX.....	15
7.3 Various analysis modes .....	16
Revision History.....	19

### 1. Install FSP with e<sup>2</sup> studio

All you have to do is install FSP and Azure RTOS TraceX because FSP includes Azure RTOS ThreadX. First of all, refer to “2.1 Installing the FSP with e<sup>2</sup> studio Installer” in the *Renesas e<sup>2</sup> studio 2021-04 or higher User’s Manual: Quick Start Guide*.

### 2. Install Azure RTOS TraceX

Refer to “Chapter 2 - Installation and use of Azure RTOS TraceX” of the *Azure RTOS TraceX for Azure RTOS User Guide*.

### 3. Creating a project in e<sup>2</sup> studio

A project generation wizard is available in e<sup>2</sup> studio to generate an RA project with a project name and the associated device and board, including drivers. Launch e<sup>2</sup> studio and choose a workspace folder in the e<sup>2</sup> studio Launcher. To create a new RA project, follow these steps:

1. Select **File** menu > **New** > **Renesas C/C++ Project** > **Renesas RA**.
2. Select the **Renesas RA: Renesas RA C/C++ Project** template. Click **Next** to continue.

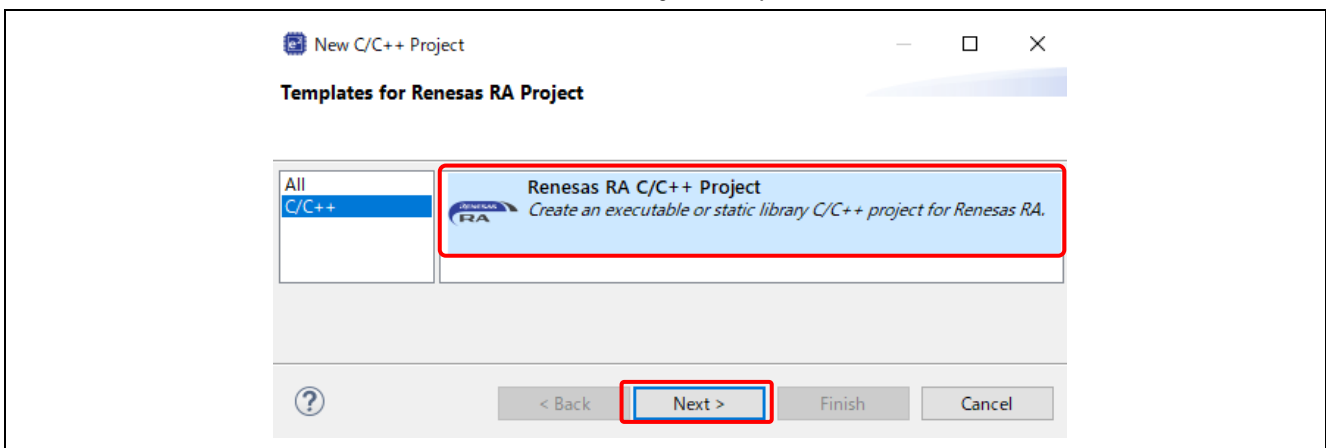


Figure 1. Template Selection

3. In the next dialog box, enter a project name and click **Next**.

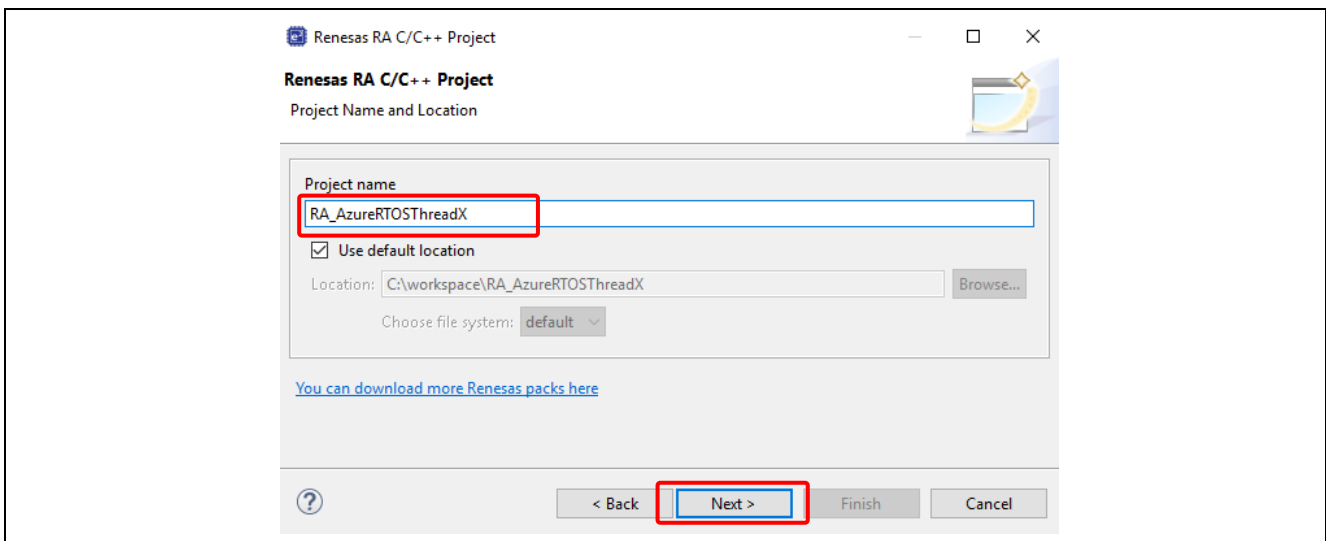


Figure 2. Project Name and Location

4. In the device selection dialog, enter device and tool information as follows.
  - FSP version: **3.0.0**
  - Board: **EK-RA6M3**
  - Device: Auto selected
  - Language: **C**
  - Toolchain version: Latest GNU Arm Embedded Toolchain approved for use with Renesas RA. (for example, GCC ARM Embedded 9.3.1.20200408)
  - Debugger: **J-Link ARM**
  - Click **Next** to continue

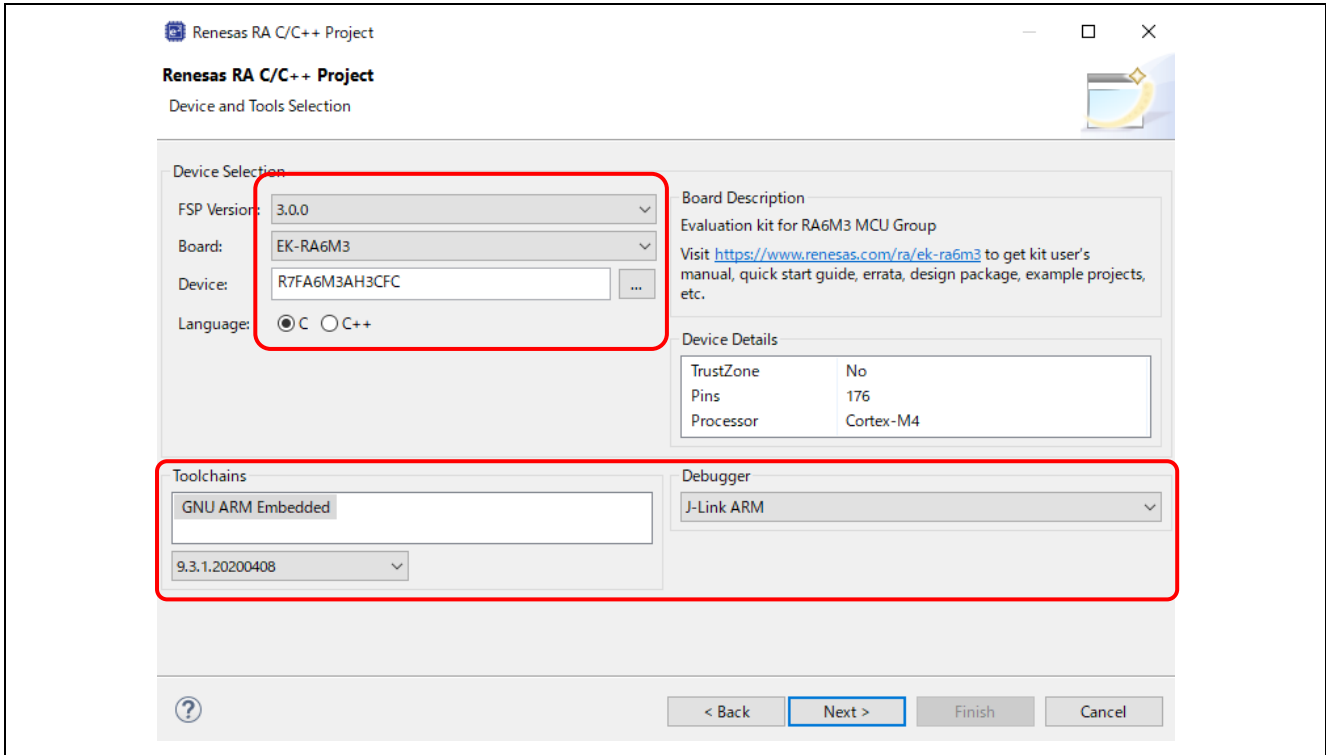


Figure 3. Create New Project for EK-RA6M3

- Build Artifact Selection: **Executable**.  
RTOS Selection: **Azure RTOS ThreadX**

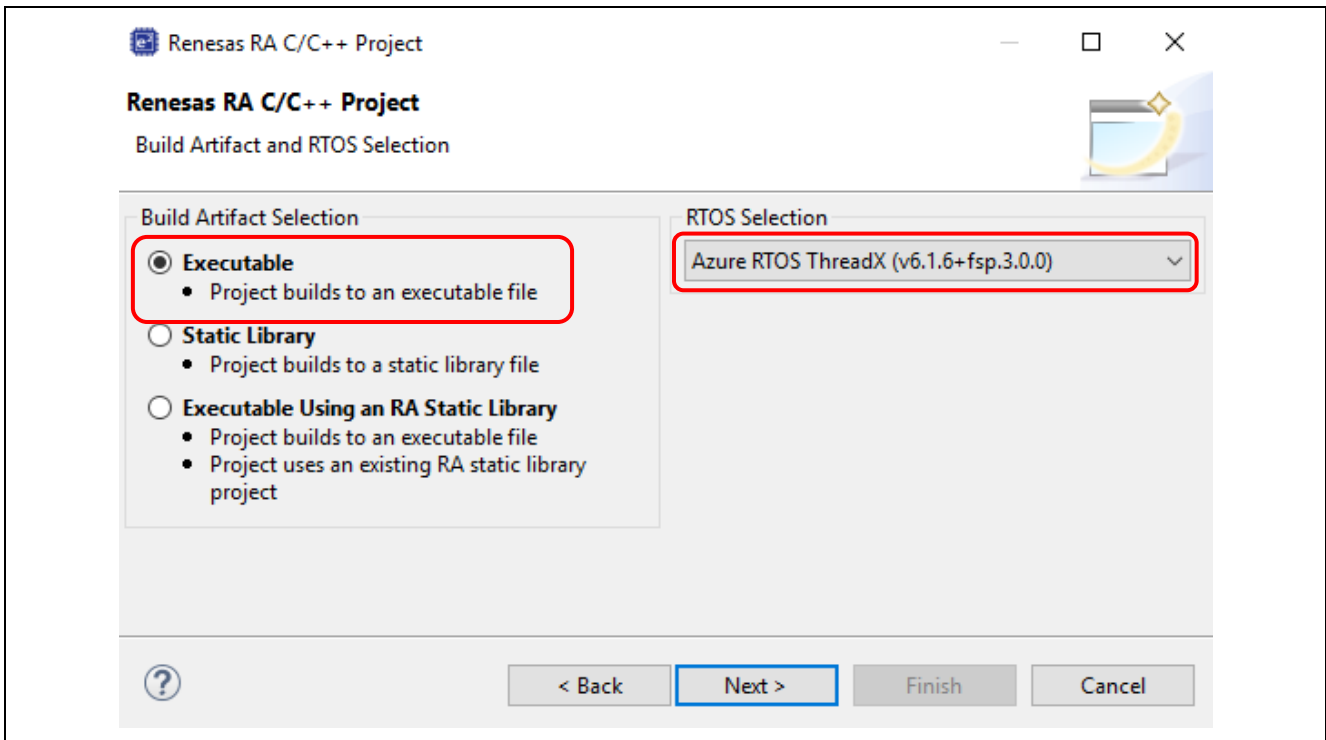


Figure 4. Build Artifact and RTOS Selection

- In the project template dialog, select **Azure RTOS ThreadX – Blinky** and click **Finish**.

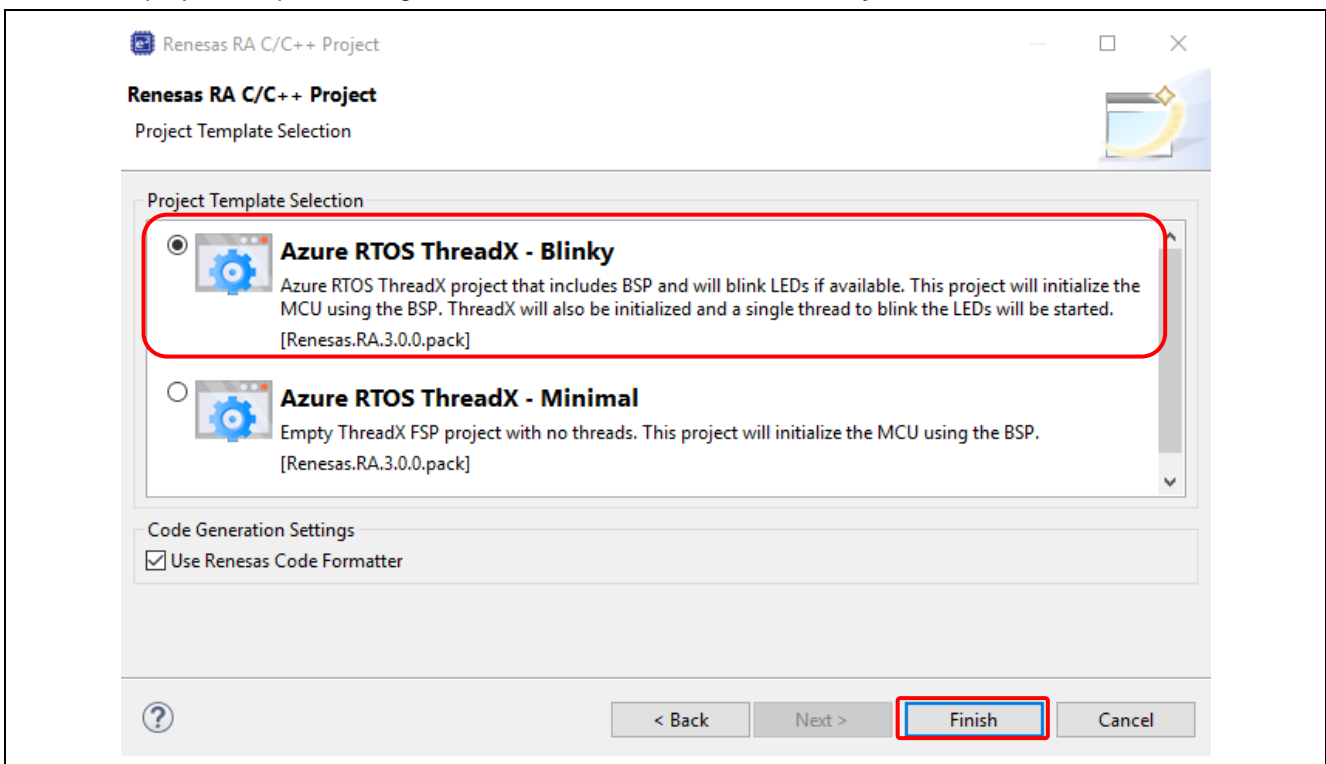


Figure 5. Project Template Selection

- Once this is complete, e<sup>2</sup> studio creates a new project with the **FSP Configuration** perspective open and ready for project configuration.

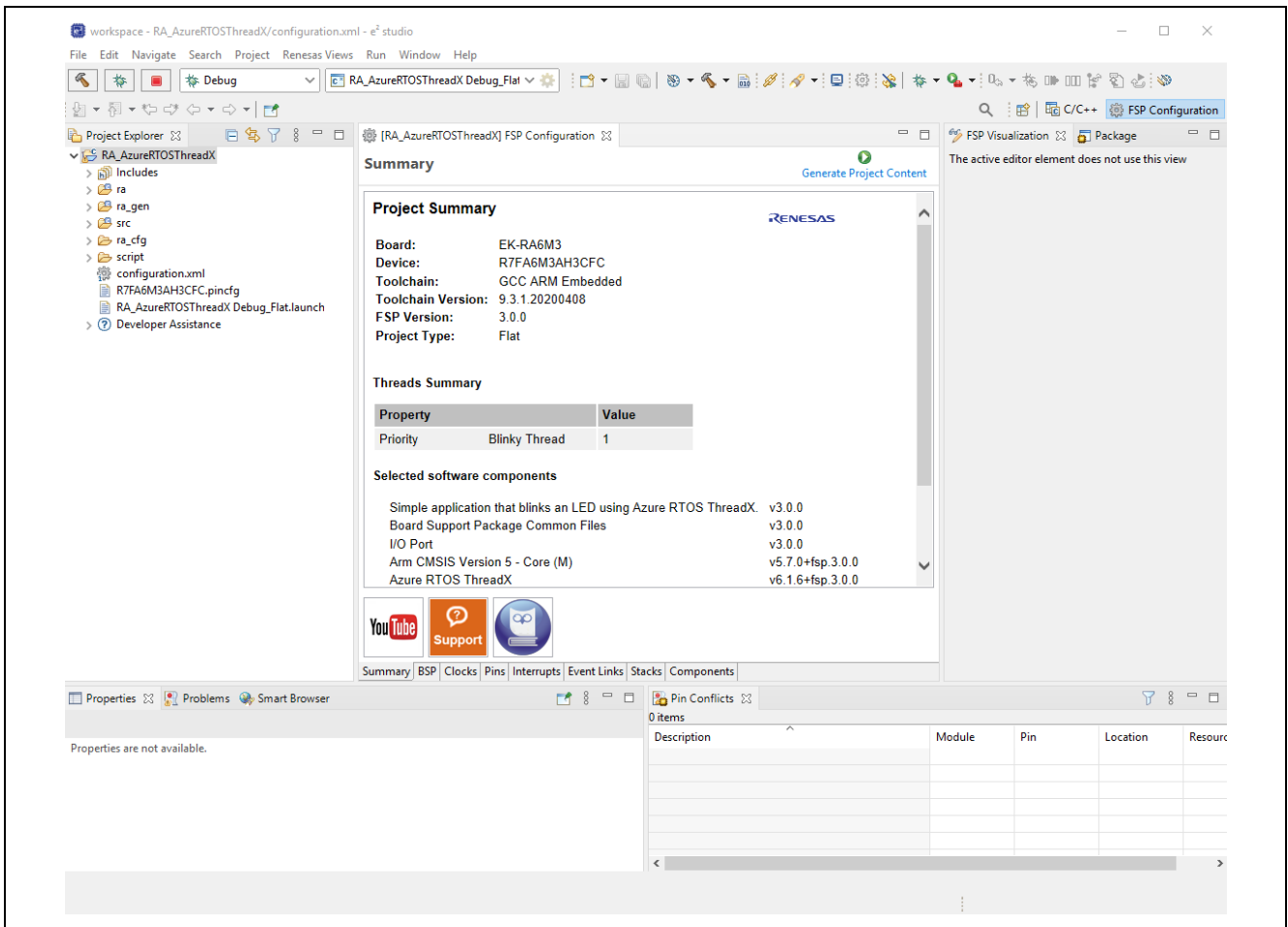


Figure 6. New Project for EK-RA6M3

## 4. FSP Configuration

### 4.1 Blinky Thread Settings

The default setting for trace buffers is within the green border.

Change the **Properties** on Blinky Thread as shown in Figure 7 below.

The definition name in "Thread properties" below is linked to the definition name of "Azure RTOS ThreadX".

**Table 1** Linked to the definition name

Properties	Change the Value	Definition name of "Azure RTOS ThreadX"
Event Trace	Enabled (Required)	TX_ENABLE_EVENT_TRACE
Block Pool Performance Info	(arbitrary)	TX_BLOCK_POOL_ENABLE_PERFORMANCE_INFO
Byte Pool Performance Info	(arbitrary)	TX_BYTE_POOL_ENABLE_PERFORMANCE_INFO
Event Flags Performance Info	(arbitrary)	TX_EVENT_FLAGS_ENABLE_PERFORMANCE_INF O
Mutex Performance Info	(arbitrary)	TX_MUTEX_ENABLE_PERFORMANCE_INFO
Queue Performance Info	(arbitrary)	TX_QUEUE_ENABLE_PERFORMANCE_INFO
Semaphore Performance Info	(arbitrary)	TX_SEMAPHORE_ENABLE_PERFORMANCE_INFO
Thread Performance Info	(arbitrary)	TX_THREAD_ENABLE_PERFORMANCE_INFO
Timer Performance Info	(arbitrary)	TX_TIMER_ENABLE_PERFORMANCE_INFO

Note: For more information see "Detailed Configuration Options" in "Chapter 2 - Installation and Use of Azure RTOS ThreadX" in the *Microsoft Azure RTOS ThreadX user guide site* as listed in the Operating Environment section.

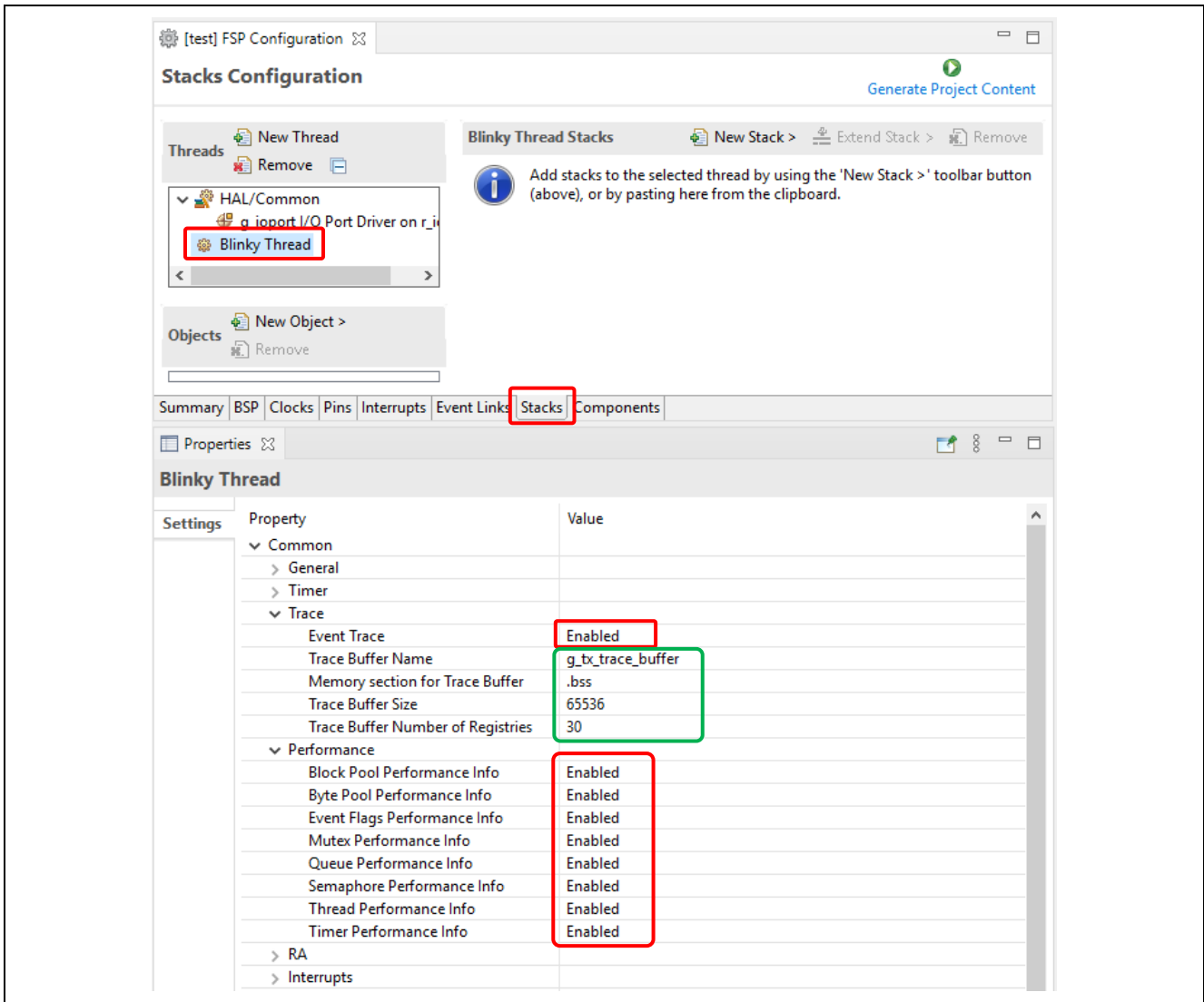


Figure 7. Blinky Thread Properties

To generate Project Content

- Click on the [Generate Project Content](#) button to generate the source files.

## 4.2 Build the project

Right-click on the project and select **Build Project**. Confirm there is no error.



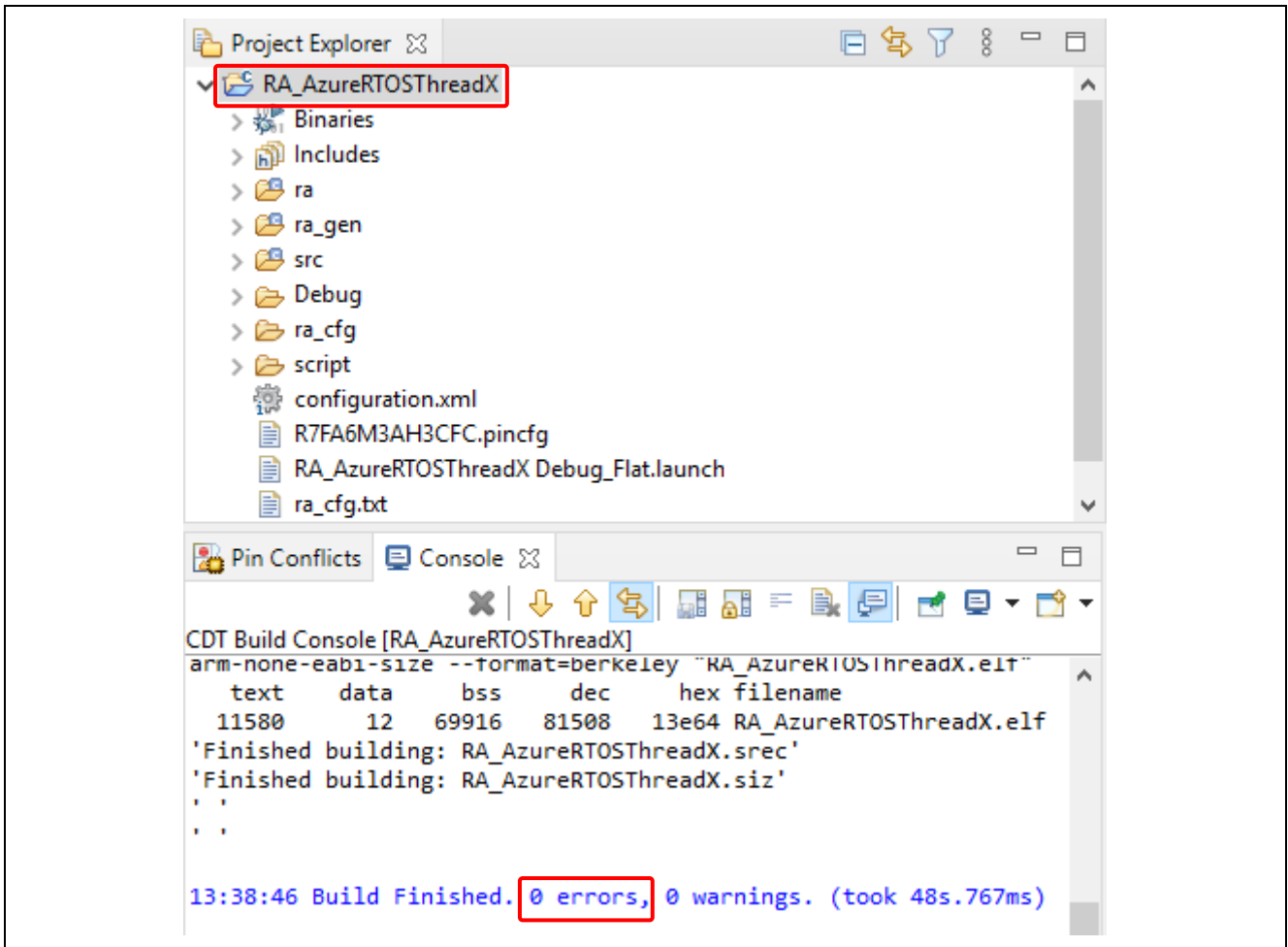


Figure 8. Build Project

### 5. Connect PC and EK-RA6M3 Board

Figure 9 below shows the connection between the host PC and the EK-RA6M3 board.

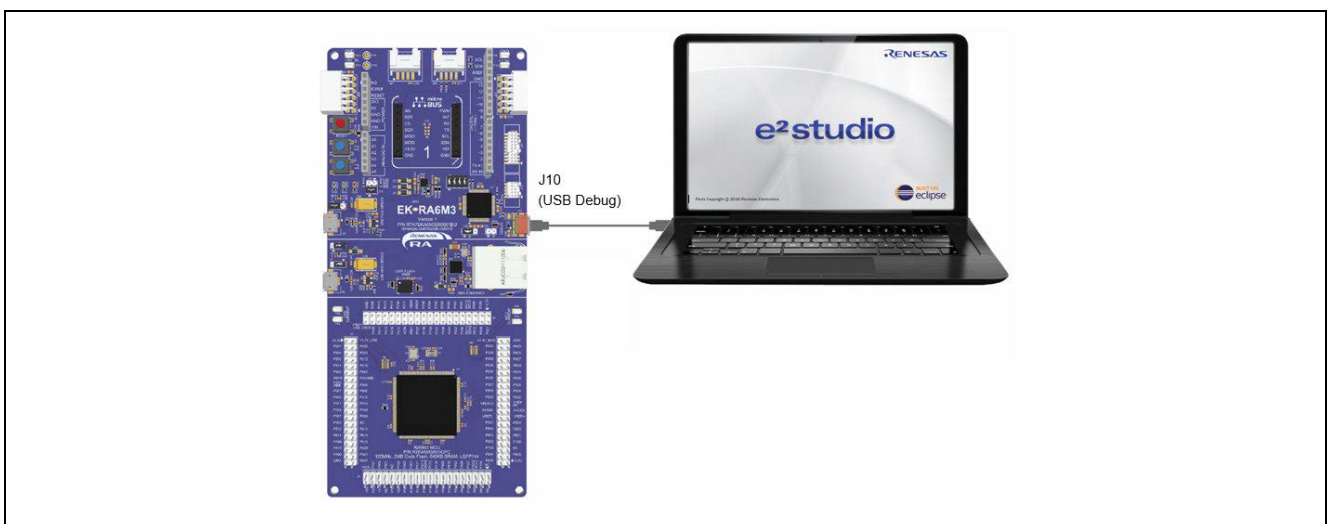


Figure 9. EK-RA6M3 Board Connection

### 5.1 Debugging Setting

The hardware settings are as follows:

**Table 2 Jumper Connection Summary for Different Debug Modes**

Debug Modes	J8	J9	J29
Debug on-board	Jumper on pins 1-2	Open	Jumpers on pins 1-2, 3-4, 5-6, 7-8

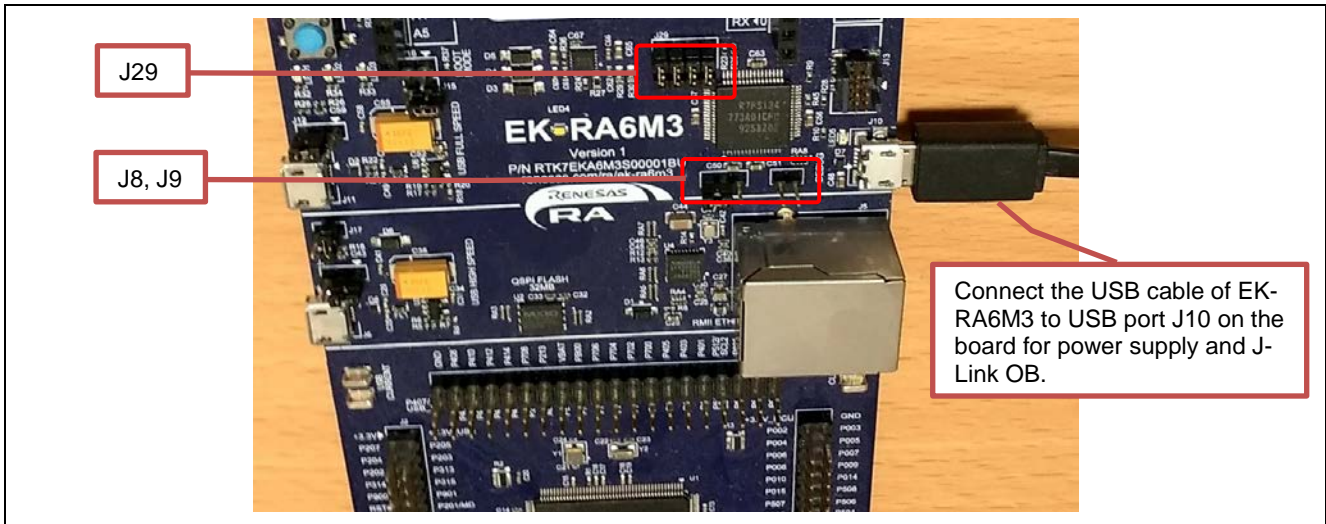


Figure 10. Connection between PC and EK-RA6M3 Board

## 6. Using the RTOS Resource View

The e<sup>2</sup> studio has an RTOS resource view function that displays the state of resources of Azure RTOS ThreadX. This procedure describes how to use the RTOS resource view.

### 6.1 Displaying the RTOS Resources View

Because the RTOS Resources view functions only with the debugger running, then start the debugger and select **Renesas Views > Partner OS > RTOS Resources**. When the **Select OS** dialog box is displayed, select **ThreadX** as shown in Figure 11. The **RTOS Resources** view appears as shown in Figure 12.

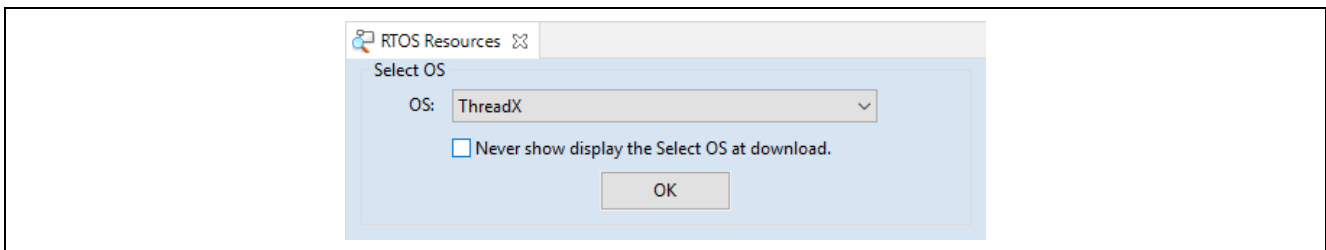
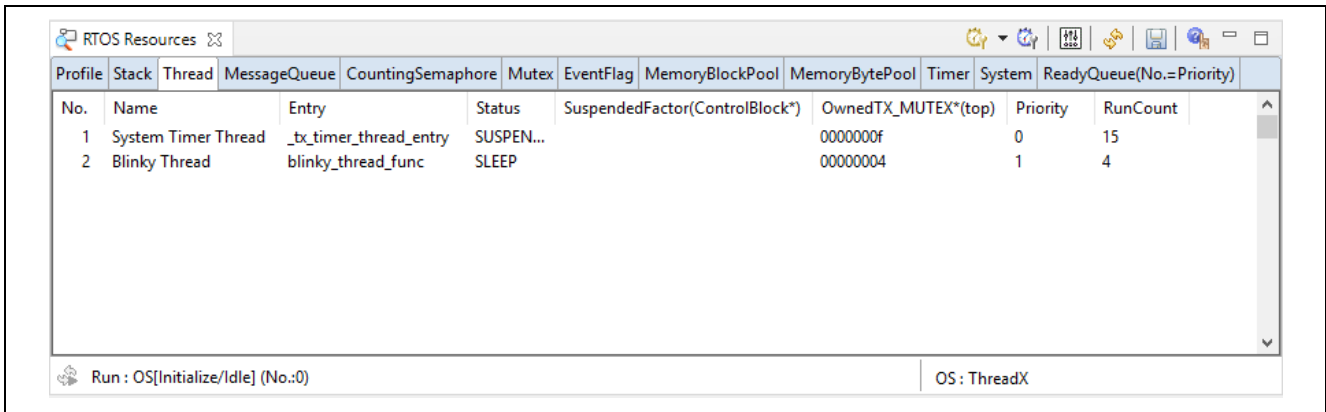


Figure 11. Selecting the OS



Profile	Stack	Thread	MessageQueue	CountingSemaphore	Mutex	EventFlag	MemoryBlockPool	MemoryBytePool	Timer	System	ReadyQueue(No.=Priority)
No.	Name	Entry	Status	SuspendedFactor(ControlBlock*)	OwnedTX_MUTEX*(top)	Priority	RunCount				
1	System Timer Thread	_tx_timer_thread_entry	SUSPEN...		0000000f	0	15				
2	Blinky Thread	blinky_thread_func	SLEEP		00000004	1	4				

Run : OS[Initialize/Idle] (No.:0) OS : ThreadX

Figure 12. RTOS Resources view

## 6.2 Context menu

The context menu is displayed by right-clicking the mouse on the **RTOS Resources** view.

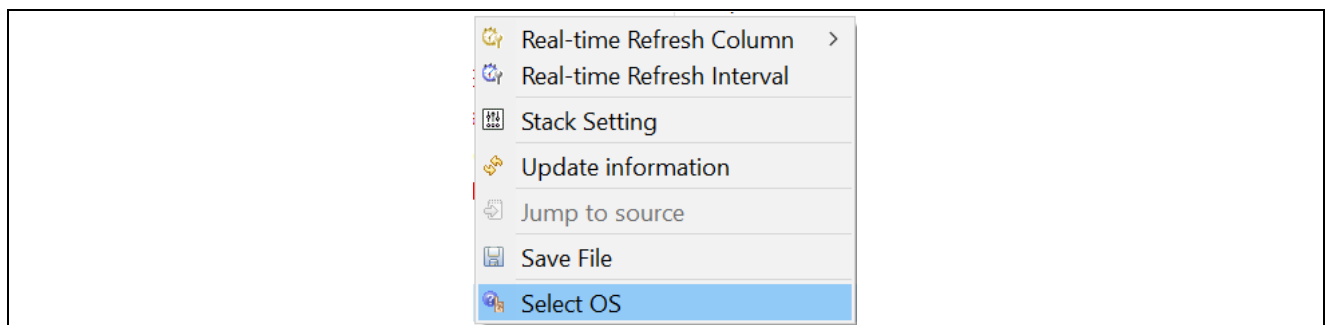


Figure 13 Context menu

- **Real-time Refresh Column:**  
Allows real-time display for the displayed items.  
This is not valid while the program is running.
- **Real-time Refresh Interval:**  
Specifies interval time for updating the real-time display. The specifiable range is 500 ms to 10000 ms.  
This is not valid while the program is running.
- **Stack Setting:**  
Enables/disables Stack Loading and stack threshold setting for stack alert function.  
This is not valid while a program is running.
- **Update information:**  
Updates the information.
- **Jump to source:**  
Opens an editor view in which the source code of the task/thread or handler is displayed. An editor view can also be opened by double-clicking the task/thread or handler.  
This is not valid while the program is running.
- **Save File:**  
Saves the data of the current tab in the text file (\*.txt).  
This is not valid while the program is running.
- **Select OS:**  
Opens the [Select OS] Dialog Box.  
This is not valid while the program is running.

## 6.3 Stack setting

### 6.3.1 Enable load stack data and set stack threshold

1. Open the context menu and select **Stack Setting**.
2. To load stack data to the RTOS Resource view, check the **Enable loading Stack data** checkbox in the **Stack Setting** dialog. If this option is not enabled, stack data will not be loaded in the next debugging session.

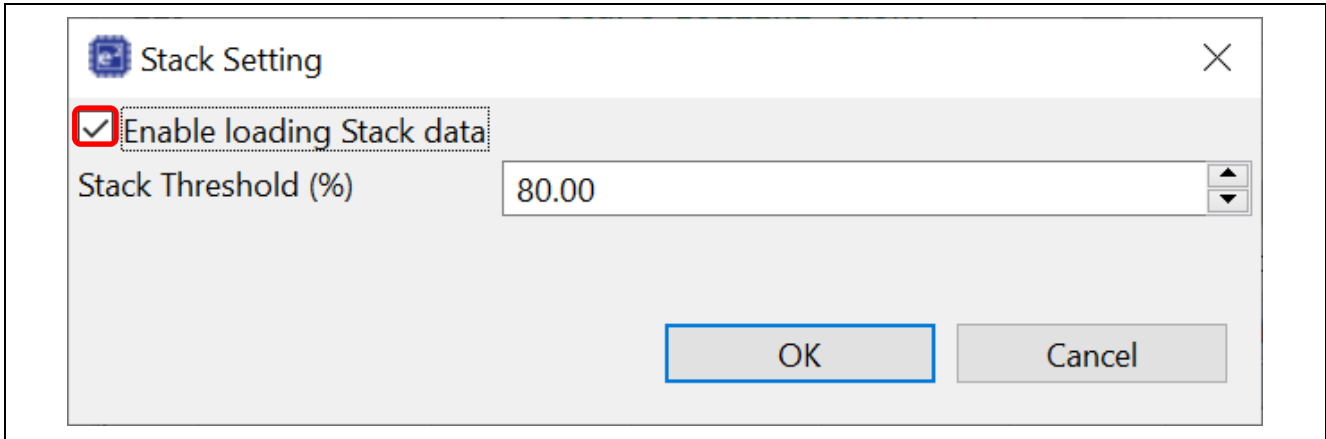


Figure 14. Enable loading stack data

3. The desired threshold value can be set in the **Stack Threshold (%)** textbox. Click **OK** to save the setting.

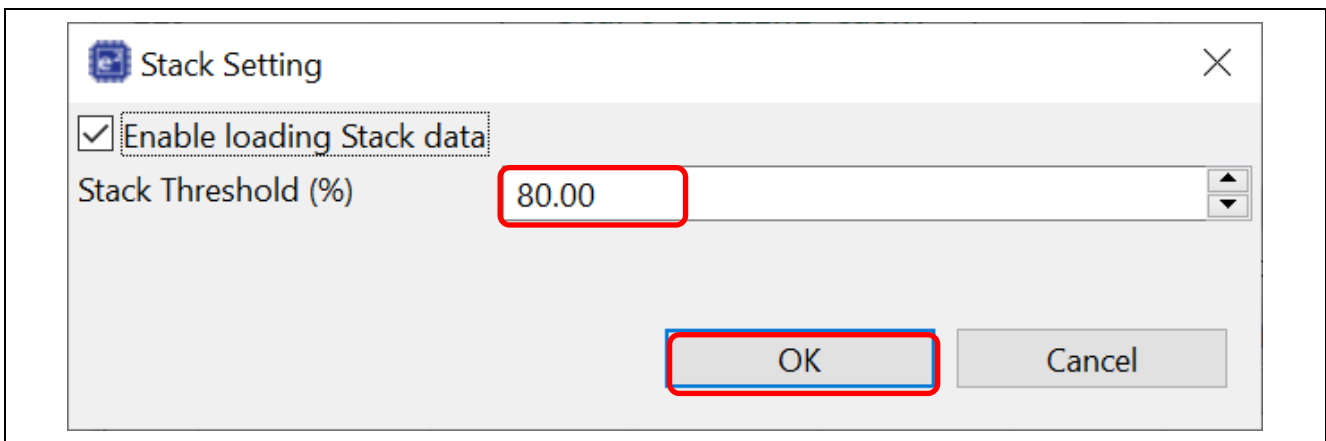


Figure 15. Set up threshold value

## 6.4 Tab Menu

Display items for each tab is shown in Table 3.

**Table 3. Contents of each tabbed window**

Name of tabbed window in the RTOS Resources view	Displayed information and selections	Information to be displayed
Profile	The Profile tab is feature is currently unavailable	-
Stack	Name	Names of the threads
	Entry	Functions that started each of the threads
	StackPointer	Current stack pointer
	StackStart	Address where the stack starts
	StackEnd	Address where the stack ends
	StackSize(bytes)	Stack size
	MaxStackUsage(bytes)	Maximum of the stack used currently
Thread	Name	Names of the threads
	Entry	Functions that start each of the threads
	Status	State of the thread
	Suspended Factor (Control Block*)	Resource that is the source of suspension
	OwnedTX_MUTEX*(top)	Acquired top mutex
	Priority	Priority
	RunCount	Number of times the thread has been executed
MessageQueue	Name	Names of the message queues
	UsedCount	Number of message queues in use
	FreeCount	Number of available message queues
	TotalCount	Total number of message queues
	MessageSize	Message size
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads
	StartAddress	Address where the message queue starts
EndAddress	Address where the message queue ends	
CountingSemaphore	Name	Names of the semaphores
	SemaphoreCount	Number of semaphores
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads
Mutex	Name	Names of the mutexes
	OwnerTX_THREAD*	Acquiring thread
	OwnerCount	Number of owners
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads
EventFlag	Name	Names of the event flags
	Flag	Current flag pattern
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads

MemoryBlockPool	Name	Names of the memory blocks
	FreeCount	Number of available blocks
	TotalCount	Total number of blocks
	BlockSize(bytes)	Block size
	TotalSize(bytes)	Total size of memory block pools
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads
	StartAddress	Top address of a memory block pool
MemoryBytePool	Name	Names of the memory pools
	Free(bytes)	Number of available bytes
	Total(bytes)	Total size of memory byte pools
	FragmentCount	Number of fragments
	SuspendedTX_THREAD*(top)	Thread at the top of waiting threads in a queue
	SuspendedCount	Number of suspended threads
	StartAddress	Address where the memory byte pool starts
Timer	Name	Names of the timers
	Remaining Tick	Remaining time
	Re-initialization Tick	Cycle time
System	SystemClock	System clock
ReadyQueue(No. = Priority)	QueuedTX_THREAD*(top)	Top ready thread

### 7. Start debugging a project with Azure RTOS TraceXLaunch debugger on e<sup>2</sup> studio

Select menu **Run > Debug** to launch the debugger.

Create a data file for Azure RTOS TraceX:

1. Add **"g\_tx\_trace\_buffer"** in **"tx\_user.h"** to **Expressions** to find the data buffer address for Azure RTOS TraceX.

\* In Figure 16, it is allocated in the address **0x1ffe0044**.

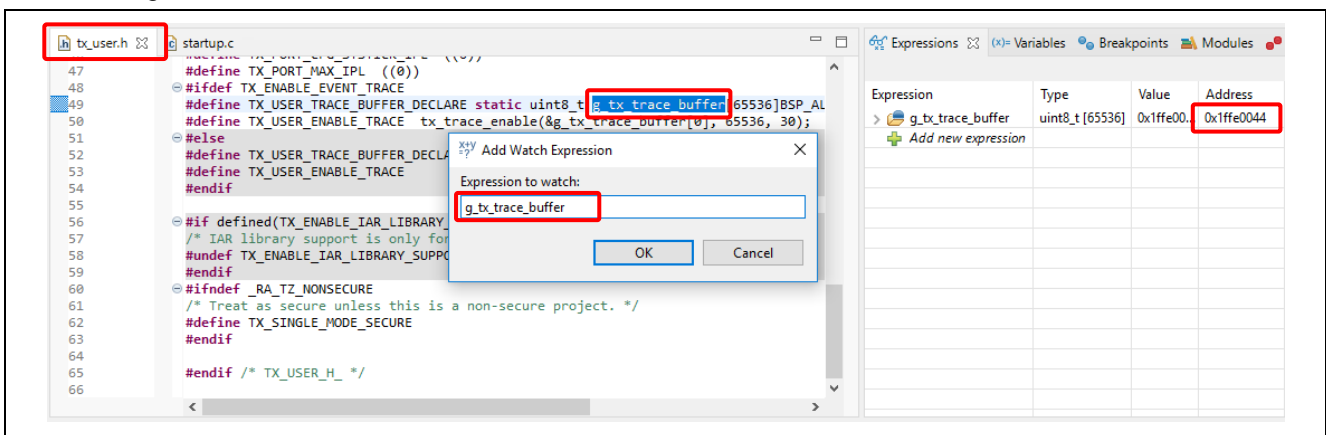


Figure 16. Find the data buffer address

2. Export a data file for TraceX.
  - (1) Select the **Memory** tab.
  - (2) Press the **Export** button to the right of the **Memory** Tab.
  - (3) **Export Memory** pops up.
  - (4) Format: Select RAW Binary.
  - (5) For **Start address**, enter the TraceX data buffer address **0x1ffe0044** found in 1. above.
  - (6) For **Length**, enter **65536** for the TraceX data buffer **g\_tx\_trace\_buffer**.
  - (7) For **File name**, create an arbitrary file name with the extension **\*\*\*\*.trx**.
  - (8) Press **OK**. (The data file for TraceX is exported with the file name specified in (7),)

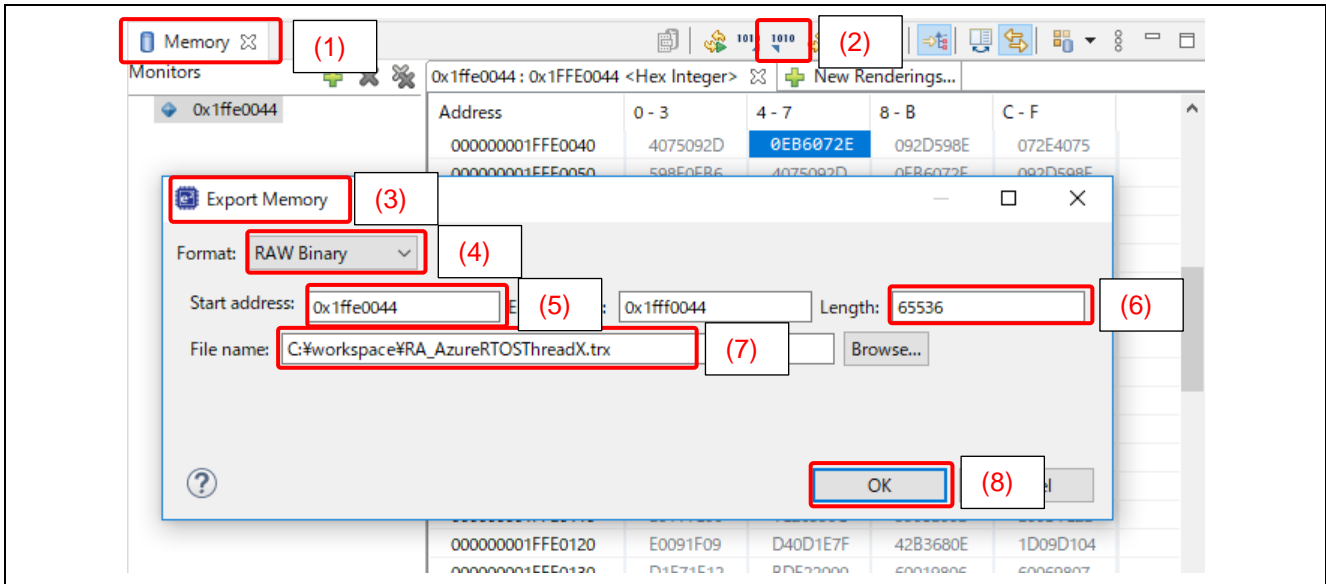


Figure 17. Export a Data File

## 7.2 Launch Azure RTOS TraceX

Launch installed Azure RTOS TraceX on PC.

Click the **File > Open** menu on Azure RTOS TraceX

Select the TraceX data file **\*\*\*\*.trx** exported in the previous step of Export a data file for TraceX..

The trace information is seen on TraceX.

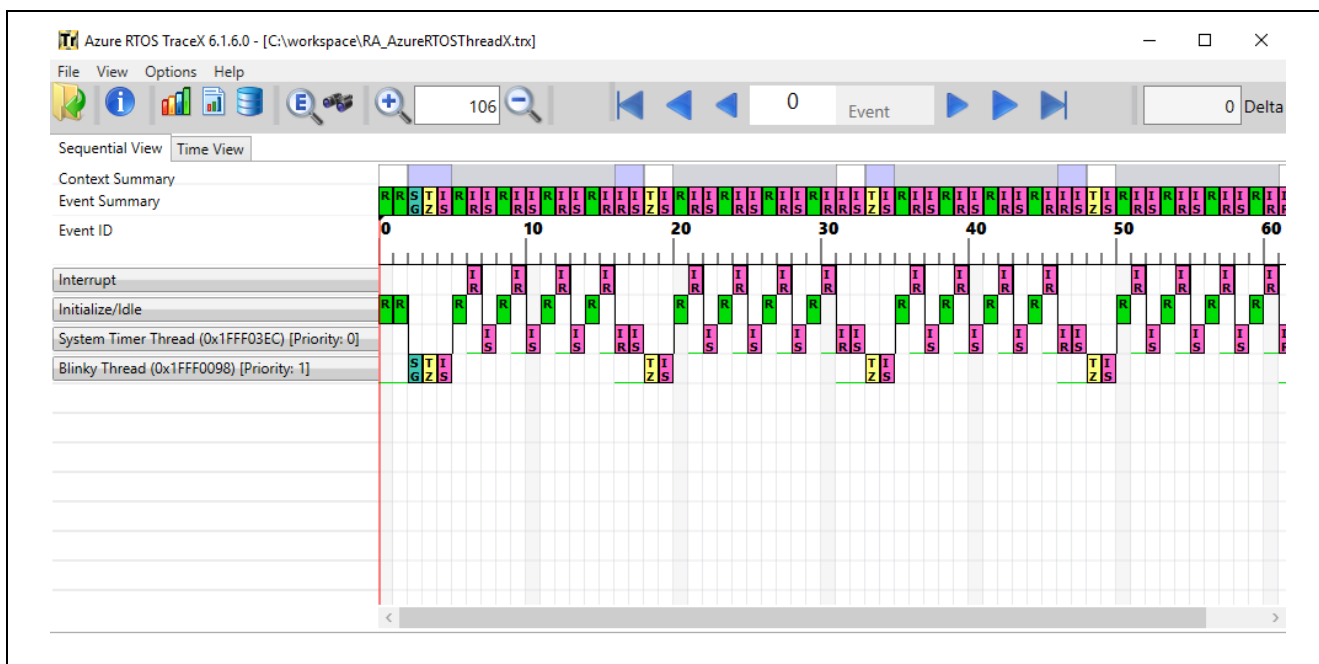


Figure 18. Launch Azure RTOS TraceX

### 7.3 Various Analysis Modes

Various analysis modes are provided. For more information, see **Help > Manual**.

Refer to "[Chapter 3 – Description of Azure RTOS TraceX](#)" of the *TraceX User Guide* | Microsoft Docs

- Sequential View Mode
- Time View Mode

Refer to "[Chapter 4 - Azure RTOS TraceX performance analysis](#)" of the *TraceX User Guide* | Microsoft Docs.

- Execution Profile
- Popular Services
- Thread Stack Usage
- Performance Statistics
- FileX Statistics
- NetX Statistics
- Trace File Information





**Website and Support**

Visit the following vanity URLs to learn about key elements of the RA family, download components and related documentation, and get support.

RA Product Information	<a href="http://www.renesas.com/ra">www.renesas.com/ra</a>
RA Product Support Forum	<a href="http://www.renesas.com/ra/forum">www.renesas.com/ra/forum</a>
RA Flexible Software Package	<a href="http://www.renesas.com/FSP">www.renesas.com/FSP</a>
Renesas Support	<a href="http://www.renesas.com/support">www.renesas.com/support</a>

**Revision History**

Rev.	Date	Description	
		Page	Summary
1.00	Jun.18.21	—	First release document

## 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](http://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/](http://www.renesas.com/contact/).