RL78/G24 Simulator V1.00.00

Release Note

Thank you for using the RL78/G24 simulator.

This document describes the target devices, simulation functions, and points for caution regarding the RL78/G24 simulator which operates in e^2 studio 2023-04.

Read this document before using the product.

Contents

Chapter 1. Target Devices and Supported Simulation Functions

The RL78/G24 simulator supports the following target device[Note].

Device group	Device name
RL78/G24	R7F101GLG (64 pins) and R7F101GAG (30 pins)

As well as CPU instructions, the RL78/G24 simulator is capable of simulating the following items in the target devices.

- Peripheral modules such as timers, the serial array unit, the serial interface, and the flexible application accelerator (FAA)
- Virtual target board (simulation via the [I/O panel] window)
- MCU pin signal waveforms (simulation via the [Timing chart] window)

Note that the RL78/G24 simulator does not support simulation of current drawn by this MCU.

Note: When the device of the RL78/G24 group other than the target device is used, the simulator operates as the RL78/G24 FAA simulator which handles simulation of the FAA.
For the functions of the RL78/G24 FAA simulator, refer to Chapter 2, Functions of the RL78/G24 FAA Simulator.

Chapter 2. Functions of the RL78/G24 FAA Simulator

This section describes the functions of RL78/G24 FAA simulator.

2.1 Functions supported by the RL78/G24 FAA simulator

The RL78/G24 FAA simulator simulates the functions of the RL78 instruction simulator and the following items for the RL78/G24.

- FAA (instructions and functions)
- Data shared memory (SHDMEM)
- Divider
- Interrupt function
- Timer array unit (only channels 0 to 3 in unit 0 and only available as interval timers)
- Clock generator

2.2 Points for caution regarding the RL78/G24 FAA simulator

For details on points for caution regarding the functions supported by the RL78/G24 FAA simulator, refer to Chapter 3, Points for Caution, which describes the points for caution regarding the RL78/G24 simulator. This section describes points for caution specific to the RL78/G24 FAA simulator.

2.2.1 Clock generator

For the clock generator of the RL78/G24 FAA simulator, the following points differ from those of the RL78/G24 simulator.

- In the RL78/G24 FAA simulator, the XTSEL bit in the clock operation mode control register (CMC) for the 20- to 32-pin products is fixed to 0.
- The following SFRs which belong to the clock generator are not simulated in the RL78/G24 FAA simulator.

  Although read/write access for the register can proceed normally, the operation does not change even if the value is changed.
  - High-speed clock select register (HSCLKSEL)
  - PLL control register (DSCCTL)
  - Main clock control register (MCKC)
  - Peripheral clock control register (PCKC)

2.2.2 Input event controller (GRNINPUTC)

In the RL78/G24 simulator, the input event controller (GRNINPUTC) detects the edge of an event input signal externally input to the FAA. In the RL78/G24 FAA simulator, the GRNINPUTC only detects the end of division operation.

Chapter 3. Points for Caution

This section lists points for caution on using the RL78/G24 simulator. These points for caution are in the following two categories.

• Differences in behavior between the target devices and the simulator due to simulator specifications

• Usage of simulation functions (operations in and configuration of the GUI windows)

3.1 Differences in behavior between the target devices and the simulator

3.1.1 Peripheral functions not supported by the simulator

The simulator is not capable of simulating the following peripheral functions of the target devices.

- Regulator
- Power-on-reset circuit
- Voltage detector
- Flash self-programming
- Operation state control
- Digital addressable lighting interface (DALI)
- Security function
- CSI slave communication mode of the serial array unit
- 16-bit timers KB30, KB31, and KB32

3.1.2 Peripheral I/O redirection registers x (PIORx)

The peripheral I/O redirection registers x (PIORx) can be manipulated by a program or debugger operations to re-assign specific multiplexed pin functions to alternative port pins in the same way as on the actual device.

Note, however, that the assignment of serial interface functions to port pins must not be changed since doing so will disable normal connections through the [Serial] window.

After re-assigning a given pin function by using the PIORx, be sure to select the name of the port pin you are currently using in the [Select Pin] dialog box of the simulator GUI. Note that when a multiplexed function is assigned to a port pin by using the PIORx, the name of the multiplexed pin function is not displayed as the pin name by the simulator.

3.1.3 Operation of PIOR06 in the peripheral I/O redirection register 0

When the setting of bit 6 (PIOR06) of the peripheral I/O redirection register 0 (PIOR0) is 1 in 64-, 52-, and 48-pin products of the target device, the TxD0_1 and RxD0_1 functions can be assigned to P12 and P11, respectively.

At this time, TxD0 and RxD0 function assignments made by the PIOR01 bit become invalid, and the TxD0_1 and RxD0_1 functions become valid.

In the simulator, when PIOR06 is set to 1, the SO00/TxD0 and SI00/SDA00 (input pin)/RxD0 functions are assigned to P12 and P11, respectively, regardless of the setting of PIOR01. Accordingly, the methods of assignment for SO00 and SI00/SDA00 (input pin) differ from those in the target device.

When the SO00 and SI00/SDA00 (input pin) functions are to be used in the simulator, set PIOR06 to 0.

3.1.4 Special function registers (SFRs) for controlling port functions

The following SFRs which control port functions are not simulated.

Although read/write access for each register can proceed normally, the operation does not change even though the value is changed.

- Port input mode registers (PIMxx)
- Port digital input disable registers (PDIDISxx)
- Global digital input disable register (GDIDIS)
- Output current control enable register (CCDE)

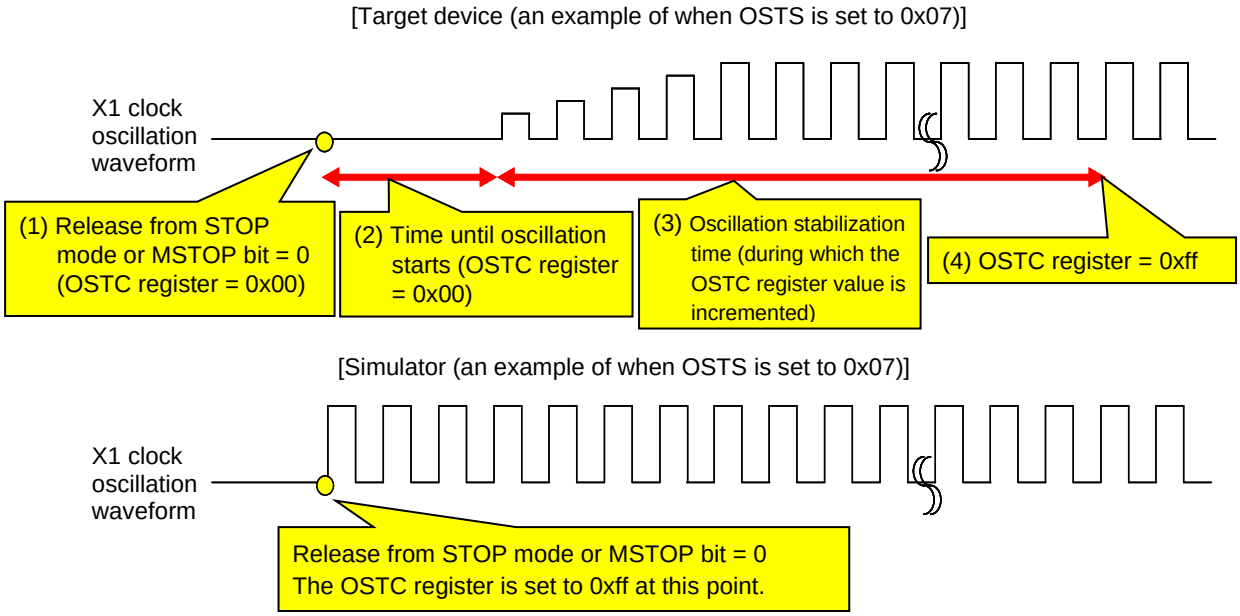
3.1.5 Oscillation stabilization time for the clock generator

Since the simulator does not simulate the clock oscillator oscillation stabilization time, stabilization always takes no time. When the oscillation is started, the OSTC register is set to one of the following values (i.e. not incremented).

OSTS Setting	OSTC Value
0x0 : 2^8/fx	0x80
0x1 : 2^9/fx	0xc0
0x2 : 2^{10}/fx	0xe0
0x3 : 2^{11}/fx	0xf0
0x4 : 2^{13}/fx	0xf8
0x5 : 2^{15}/fx	0xfc
0x6 : 2^{17}/fx	0xfe
0x7 : 2^{18}/fx	0xff

The following figure illustrates this operation.

In the target device, oscillation by the X1 clock starts after operation has passed through states (1) to (4). In the simulator, states (1) through (4) are skipped and oscillation instantly starts.



Therefore, pay attention to the code that waits for oscillation stabilization.

There is no problem if a program is created under the condition that execution proceeds after the oscillation stabilization wait period when the OSTC register value becomes the maximum value, or when the OSTC register value exceeds the specified value, but if a program is created under the condition that execution proceeds after the oscillation stabilization wait period when the OSTC register value becomes a value other than the maximum value, execution will enter an endless loop.

The following shows examples of code that causes and does not cause problems.

The examples are when the OSTS is set to 0x07.

Correct code example (1)

```
while(OSTC != 0xff)
{
NOP();/* wait */
}
```

Correct code example (2)

```
while(OSTC <= 0xf0)
{
NOP();/* wait */
}
```

Example of code that may cause problems

```
while(OSTC != 0xf0)
{
NOP();/* wait */
}
```

3.1.6 SFRs (CMC, OSMC, HIOTRM, MIOTRM, and LIOTRM) in the clock generator

The following SFRs which belong to the clock generator are not simulated. Although read/write access for each register can proceed normally, the operation does not change even if the value is changed.

- Bits 0, 1, and 2 (AMPH, AMPHS0, and AMPHS1) of the clock operating mode control register (CMC)
- Bit 0 (HIPREC)[Note] of the subsystem clock supply mode control register (OSMC)
- High-speed on-chip oscillator trimming register (HIOTRM)
- Middle-speed on-chip oscillator trimming register (MIOTRM)
- Low-speed on-chip oscillator trimming register (LIOTRM)

Note: In the simulator, the HIPREC bit is fixed to 0. However, if the STOP instruction is executed, the RL78/G24 will enter the STOP mode.

The operation in initialization of the EXCLKS, OSCSELS, XTSEL, AMPHS1, and AMPHS0 bits of the clock operation mode control register (CMC) differs between the target device and the simulator.

[Target device]

EXCLKS, OSCSELS, XTSEL, AMPHS1, and AMPHS0 bits are only initialized by a power-on reset and retain their values following resetting by other reset sources.

[Simulator]

The bits are initialized by the following reset sources.

- External reset input via the RESET pin
- Internal reset due to detection of a program malfunction by the watchdog timer

3.1.7 Prefetching

When the main system clock (fMAIN) is running at 48 MHz in the target device, setting the PFBE bit in the prefetch buffer enable register (PFBER) to 1 enables prefetching. However, in the simulator, the main system clock (fMAIN) runs at 48 MHz even if prefetching is not enabled.

3.1.8 Operating clock of the timer array unit

Do not specify an operating clock that runs at or below 233 Hz. If the operating clock for the timer array unit runs at or below 233 Hz, then the timer array unit will not work properly (it will behave as if operating with a clock that is faster than the one selected).

3.1.9 Noise filter of the timer array unit

Although the target device's timer array unit has a function to turn the noise filters on and off in order to reduce noise from the timer input pins, the simulator does not simulate this function since there is no noise in the simulator's signals. That is, whether filtering is on or off makes no difference to the behavior.

3.1.10 Digital filters in timers RJ, RD2, and RG2

The simulator does not simulate the operation of the digital filters in timers RJ, RD2, and RG2.

3.1.11 Interrupts generated by Timer RD2

The timing of the generation of interrupts by extended complementary PWM mode or timer-KB PWM output gating mode in Timer RD2 differs between the target device and the simulator.

[Target device]

Regardless of the settings of the Timer RD interrupt enable registers i (TRDIERi) or the flags of Timer RD status registers i (TRDSRi), timer RD2i (i = 0 or 1) interrupt request is generated at the interrupt request occurrence timing of each operation mode (compare match between the TRD0 counter and TRDGRA0 register, TRD1 counter underflow in extended complementary PWM mode, periodic match with TRDi and counting start in timer-KB PWM output gating mode).

[Simulator]

An interrupt request is generated when the flag in Timer RD status registers i (TRDSRi) changes from "0" to "1". Therefore, if the flag is not cleared to "0" before the interrupt request generation timing, the interrupt request will not be generated.

Even in the case of extended complementary PWM mode or timer-KB PWM output gating mode, if the flag is cleared to "0" in the interrupt processing of the program, an interrupt request will be generated same as the target device.

However, when skipping of interrupt requests is performed in extended complementary PWM mode, the status flag becomes "1" when underflow occurs during the skipping period, but the interrupt processing is not executed and the flag remains "1", so no interrupt occurs thereafter.

3.1.12 SFRs (RTCC0, RTCC1, and SUBCUD) of the realtime clock (RTC)

The operation of realtime clock control register 0 (RTCC0), realtime clock control register 1 (RTCC1), and the time error correction register (SUBCUD) of the realtime clock (RTC) differs between the target device and the simulator.

[Target device]

The registers are cleared to 00H in response to an internal reset by the power-on reset circuit.

[Simulator]

The registers are cleared to 00H in response to resetting by the following reset sources.

- External reset input via the RESET pin
- Internal reset due to detection of a program malfunction by the watchdog timer

3.1.13 Clock output/buzzer output controller

When f_{MAIN} is selected as an output clock, the [Timing chart] window does not show the clock waveform of the PCLBUZn signal.

When $f_{MAIN}/2$ or a slower signal is selected as an output clock, the [Timing chart] window shows the clock waveform.

3.1.14 Interval interrupts generated by the watchdog timer

The timing of the generation of interval interrupts by the watchdog timer differs between the target device and the simulator.

[Target device]

When 75% + 1/4f_{IL} of overflow time is reached

[Simulator]

When 75% of overflow time is reached

3.1.15 A/D converter

When no voltage is being applied to the VDD or AVREFP pin, the default reference voltage of the A/D converter is 5.0 V.

To change the reference voltage, input the desired voltage values for VDD and AVREFP via the [Signal Data Editor] window.

The temperature sensor output voltage is always 1.05 V.

3.1.16 Conversion start time in the A/D converter

The simulator does not support the conversion start time but recognizes the time as 0 clock cycles in simulation.

3.1.17 D/A converter

When no voltage is being applied to the VDD pin, the simulator operates on the assumption that 5 V is being input to the VDD pin. To change the voltage, specify the desired voltage value for the VDD pin via the [Signal Data Editor] window or some other means.

3.1.18 Digital filters in the comparators (CMP)

The simulator does not simulate the operation of the digital filters in the comparators (CMP).

3.1.19 Voltages amplified by the programmable gain amplifier (PGA)

In the hardware configuration of the target device, the voltages amplified by the programmable gain amplifier (PGA) depend on the voltages on the PGAGND and VSS pins. The voltages amplified by the simulated PGA, on the other hand, only depend on the voltage on the PGAI pin (i.e. not on those on the PGAGND and VSS pins).

3.1.20 Output pin for the programmable gain amplifier (PGA)

When the PGAPOE bit in the PGA control register (PGACTL) is set to 0, the voltage output by the programmable gain amplifier is not output from the PGAO pin in the target device. In the simulator, however, the pin remains in the state of 0 being output.

3.1.21 Clock used in the serial array unit

Do not specify a clock that is 233 Hz or lower in the following cases. If the following clock of the serial array unit is 233 Hz or lower, then the serial array unit will not operate correctly (it will behave as if operating via a clock that is faster than the one selected).

- Operating clock (f_{MCK}) is 233 Hz or lower.
- Transfer clock setting by dividing the operation clock (f_{MCK} ÷ (SDRmn[15:9] + 1)) is 233 Hz or lower.

3.1.22 Noise filter of the serial array unit

Although the target device's serial array unit has a function to turn the noise filter on and off in order to reduce noise on the input pin, the simulator does not simulate this function since there is no noise in the simulator's signals. That is, whether filtering is on or off makes no difference to the behavior.

3.1.23 SDRmn registers of the serial array unit

The values read from the seven higher-order bits of the serial data registers (SDRmn) during serial operation differ between the target device and the simulator.

[Target device]

0 is read.

[Simulator]

The value read is that at the time serial operation starts.

3.1.24 SSm registers of the serial array unit

During serial communications, when the operation start trigger of channel n (SSmn) in the serial channel start register m (SSm) is set to 1, operation of the simulator differs from that of the actual target device in the way stated below.

[Target device]

The target device stops communications and enters the suspended state.

[Simulator]

The simulator does not stop communications. Accordingly, the TSFmn and BFFmn bits in the serial status register mn (SSRmn) are not cleared to 0.

3.1.25 IICA serial interface

IICA supports pin waveform generation and the communications through the [Serial] window. The following functions are not supported.

- Digital filter
- Arbitration
- Detection of transmission errors
- Communication reservation

3.1.26 SFRs (for IICCTL01 and IICM) for the IICA serial interface

The following SFRs which control the IICA serial interface are not simulated.

Although read/write access for the register can proceed normally, the operation does not change even if the value is changed.

- Bit 2 (DFC0) Note and bit 3 (SMC0) of the IICA control register 01 (IICCTL01)
- Bit 0 (IICSH) of the IICA input mode selection register (IICM)

Note: In the simulator, the DFC0 bit can be set to 1 only when the SMC0 bit is 1.

3.1.27 Times taken for data transfer by the data transfer controller (DTC)

The times taken for data transfer by the data transfer controller (DTC) differ between the target device and the simulator.

[Target device]

- A response time is required from detection of a DTC activation source until data transfer starts.
- A waiting time is required for access to extended special function registers (2nd SFRs).
- The DTC puts the data transfer on hold when the CPU executes any instruction that holds the DTC pending.
- Access to the data bus by the CPU is put on hold during DTC transfer.

[Simulator]

- Data transfer starts immediately after detection of a DTC activation source.
- No waiting time is required even for access to extended special function registers (2nd SFRs).

- The DTC does not put the data transfer on hold even when the CPU executes any instruction that should hold the DTC pending.
- Access to the data bus by the CPU is not put on hold even during DTC transfer.

3.1.28 Repeat mode of the data transfer controller (DTC)

If any of the conditions listed below is satisfied while the data transfer controller (DTC) is in repeat mode, the DTC ignores activation sources and will thus fail to transfer data.

- A DTC transfer count register j (DTCCTj) is set to 00H (number of transfers: 256 times).
- A DTC block size register j (DTBLSj) is set to 00H (block size: 256 or 512 bytes).
- A DTC control register j (DTCCRj) is used to set the transfer data size to 16 bits and the corresponding DTC block size register j (DTBLSj) is used to set the block size to 256 bytes or more.

3.1.29 Event link controller (ELC)

If any of the peripheral-module functions listed below is selected for linking by the event link controller (ELC), the simulator causes the peripheral-module function to operate immediately after reception of the event signal. The ELC in the actual device, on the other hand, causes the peripheral-module function to start operation several cycles after the ELC has received the event signal.

[Peripheral-module functions]

- Timer input of timer array unit 0 channel 0
- Timer input of timer array unit 0 channel 1
- A/D converter
- Timer RG2
- Timer RD2 event input 0
- Timer RD2 event input 1
- Timer RD2 PWM option unit A (PWMOPA)
- D/A converter 0 (DAC0)
- D/A converter 1 (DAC1)
- D/A converter 2 (DAC2)

3.1.30 SFR (WKUPMD) for the standby function

The following SFR which controls the standby function is not simulated.

Although read/write access for the register can proceed normally, the operation does not change even if the value is changed.

- Bit 0 (FWKUP) of the standby mode release setting register (WKUPMD)

3.1.31 Reset

Among the sources for generating reset signals, the following types of internal reset do not occur in the simulator.

- Internal reset by comparison of supply voltage and detection voltage of the power-on-reset circuit (POR)
- Internal reset by comparison of supply voltage and detection voltage of the voltage detectors (LVD0 and LVD1)
- Internal reset due to execution of an illegal instruction
- Internal reset due to a RAM parity error

- Internal reset due to illegal-memory access

In addition, the behavior differs as follows if a reset signal is input from the RESET pin.

[Target device]

The MCU is reset when the RESET pin goes low. Release from the reset state proceeds when the RESET pin goes high.

[Simulator]

The MCU is not reset when the RESET pin goes low. The simulator is reset momentarily and then released when the RESET pin goes high.

3.1.32 Reset control flag register (RESF)

The simulator only supports the WDTRF bit of the reset control flag register (RESF).

The simulator is not capable of simulating the operations of the other bits (TRAP, RPERF, IAWRF, and LVIRF). Only the default values of these bits are indicated.

Additionally, in the simulator, the reset control flag register (RESF) is only cleared by a reset input via the RESET pin.

3.1.33 Safety functions

The simulator does not support the following safety functions.

- Flash memory CRC operation function (high-speed CRC, general-purpose CRC)
- Flash memory guard function
- RAM parity error detection
- RAM guard function
- SFR guard function
- Illicit memory access detection
- Guard function of invalid memory access detection control register (IAWCTL)

3.1.34 Power supply for battery backup (VBAT)

The simulator does not simulate the power supply for battery backup (VBAT).

3.1.35 Executing illegal instructions

If an illegal instruction (instruction code: 0xFF) is executed, the target device will be reset, but the simulator will go into an endless loop (the illegal instruction will be executed repeatedly).

3.1.36 FAA functions not supported by the simulator

The simulator does not support the following FAA functions.

- Low-power functions (SYSC.SLP and DSYSC.SLP bits)

3.1.37 Reference timing controller of the FAA

For the reference timing controller, when a free-running counter starts operation (FCCNT.FCEN = 1), it continues operating even if a break occurs on the CPU side or FAA side or both sides. The free-running counter stops operation when FCCNT.FCEN = 0 or FAAEN = 0.

3.1.38 Interrupts generated by the reference timing controller of the FAA

If a free-running counter in the reference timing controller of the FAA is operated in the CPU program or the SFR panel with the following steps and the comparison value of the timing is small, an interrupt generated by the compare match with the timer may be ignored.

(a) Setting a register in the reference timing controller

Example: TMCMP0 = 0xF, TMMSK0 = 0x0

(b) Operating a free-running counter

(c) Setting “WIND = 0”

(d) Setting “SYSC.ENB = 1”

Note:

In the steps above, an interrupt generated by the compare match with the timer is ignored if the operation of a free-running counter starts after step (b) and FCNT is counted up to 0xF before step (d).

3.1.39 FAA instruction code memory and FAA data memory

When FAAEN = 0, the simulator will access (read or write) the FAA instruction code memory and FAA data memory from the target resource allocation area.

3.1.40 Divider

When the DIVST bit in the FAADUC register is set to 1 so that the divider starts a division in the actual hardware, the operation will end in 16 clock cycles; the simulator promptly completes the division and generates the interrupt (INTMD).

3.2 Usage of simulation functions

3.2.1 Simulation speed

The simulation speed of RL78/G24 simulator depends on the number of operating peripheral functions.

If many peripheral functions are operating, the simulation speed becomes from several to ten times slower than the actual device. Note

With the use of only a few, or even no peripheral functions, the simulation speed may become faster than the actual device.

Note: The measurement environment for simulation speed is as follows.

CPU: 3.20 GHz (Quad-Core); memory: 8 Gbytes; OS: Windows10 64-bit edition

3.2.2 Pin waveforms in the [Timing chart] window

The maximum length of a pin waveform is 4096 signal-level changing points. After reaching this maximum length, the data will be overwritten from the oldest value. If this length is not sufficient, use the following methods.

- Reduce the number of registered pins
- Stop the user program at the place where you want to confirm the waveform by using a breakpoint

3.2.3 Controlling windows

The following keyboard operations are not available in the simulator windows ([Signal Data Editor], [I/O panel], and [Serial]).

- Navigation via tab or arrow keys (←, ↑, →, ↓)
- Deletion via the Del or Backspace keys
- Cut & paste and other operations via the Ctrl + C, V, X, A, or Z keys

Perform the above operations as follows.

- Navigation: Navigate by using the mouse.
- Deletion: Right-click and perform the action from the context menu.
- Cut & paste, etc.: Right-click and perform the action from the context menu.

3.2.4 Closing the [Simulator GUI] window

The [Simulator GUI] window can only be closed by disconnecting from the debugging tool, or by closing CS+ in proper manner. The [X] button cannot be used.

Additionally, although it appears that the [X] button can be pressed if Aero is enabled in Windows, pressing this button will not close the [Simulator GUI] window.

3.2.5 Disconnecting the debug tool

CS+ may be closed if the debugging tool is disconnected while any of the following dialog boxes is open from the [Simulator GUI] window. Be sure that the following dialog boxes have been closed before disconnecting the simulator.

- Save As
- Open
- New
- Color
- Font
- Customize
- Loop
- Select Pin
- Search Data
- Format (UART)
- Format (CSI)
- Format (IIC)
- Message (e.g. Error)
- Parts Button Properties
- Analog Button Properties
- Parts Key Properties
- Parts Level Gauge Properties
- Parts Led Properties
- Parts Segment LED Properties
- Parts Matrix Led Properties
- Parts Buzzer Properties
- Pull up / Pull down
- Entry Bitmap
- Object Properties

3.2.6 [Serial] window

When using the [Serial] window as the data receiver for the simplified I^2C of the serial array unit or IICA, only ACK can be generated after receiving the data. NACK cannot be generated.

3.2.7 Inclusion of the ranges where the FAA is being executed in Run-Break times of the CPU

Since the CPU and FAA are not run in synchronization by the simulator, inclusion of the ranges where the FAA is being executed in Run-Break times of the CPU is not correct.

3.2.8 Executing a program for controlling the peripheral functions by using the FAA

When the simulator executes a program of the FAA which controls peripheral functions other than the divider, the CPU side must be in the execution state. If operation on the CPU side is stopped, the FAA becomes unable to control the peripheral functions.

Revision History

Rev.	Date	Description	
		Page	Summary
Rev.1.00	Apr.03.23	-	First Edition

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/.