Introduction
This application note explains how to realize automatic hand soap dispenser using RL78/G10.

Target Device
RL78/G10

When applying the sample program covered in this application note to another microcontroller, modify the program according to the specifications of the microcontroller and conduct an extensive evaluation of the modified program.
Contents

1. Specifications ................................................................................................................................. 3
   1.1 DC Motor .................................................................................................................................. 3
   1.2 Human Sensor ............................................................................................................................ 3

2. Operation Check Conditions ......................................................................................................... 4

3. Hardware Descriptions .................................................................................................................. 5
   3.1 Hardware Configuration ............................................................................................................. 5
   3.2 List of Pins used ....................................................................................................................... 5

4. Software Descriptions .................................................................................................................... 6
   4.1 Operation Summary .................................................................................................................. 6
   4.2 List of Option Byte Setting .................................................................................................... 6
   4.3 List of Variables ....................................................................................................................... 6
   4.4 List of Functions (Subroutines) .............................................................................................. 7
   4.5 Function Specifications .......................................................................................................... 8
   4.6 Flowcharts ............................................................................................................................... 11
     4.6.1 Initial Setting Function ...................................................................................................... 11
     4.6.2 System Function ............................................................................................................. 12
     4.6.3 I/O Port Setup .................................................................................................................. 13
     4.6.4 CPU Clock Setup ............................................................................................................ 14
     4.6.5 Timer array unit setup .................................................................................................... 15
     4.6.6 Buzzer alarm setup ........................................................................................................ 17
     4.6.7 A/D converter Setup ....................................................................................................... 18
     4.6.8 12-bit interval timer setup ............................................................................................. 19
     4.6.9 Main Processing .............................................................................................................. 20
     4.6.10 Initialization Function .................................................................................................... 21
     4.6.11 12-bit interval timer start Setting ................................................................................. 21
     4.6.12 12-bit interval timer interrupt ....................................................................................... 22
     4.6.13 A/D Conversion Start Setting ....................................................................................... 23
     4.6.14 Obtaining battery voltage ............................................................................................. 23
     4.6.15 A/D Conversion Stop Setting ....................................................................................... 24
     4.6.16 Timer array unit channel 0 operation start ................................................................. 24
     4.6.17 Timer array unit channel 0 operation stop ................................................................. 25
     4.6.18 Timer array unit channel 3 operation start ................................................................. 25
     4.6.19 Timer array unit channel 3 operation stop ................................................................. 26
     4.6.20 Buzzer alarm start ....................................................................................................... 26
     4.6.21 Buzzer alarm stop ....................................................................................................... 27

Revision History .................................................................................................................................. 29
1. Specifications

In this application note, when the automatic hand soap dispenser is turned on, RL78/G10 will be stop mode. The STOP mode is canceled with every 50ms of 12-bit interval timer, then the human sensor detects the hand. When the hand is detected, the battery voltage is checked. If the battery voltage is less than 4.8V(lower limit of motor drive voltage), the buzzer will alert you for 2 seconds and return to STOP mode. When the battery voltage is 4.8V or more, the LED blinks and the motor is driven to discharge the hand soap for 2 seconds. Then return to STOP mode.

Figure 1.1 shows the system configuration outline.

![System Configuration Diagram](image)

1.1 DC Motor

In this application note, the rotation speed of the fan motor is controlled by changing the duty ratio of the PWM output. In order to suppress power losses in fan driving, a power MOSFET capable of fast switching and with a low ON-resistance is used. When actually building a circuit, the design should satisfy the electrical characteristics of the model used.

1.2 Human Sensor

This application system uses a module with a pyroelectric infrared sensor (hereinafter called a human sensor). According to the specifications, the system begins to monitor all around the sensor at some seconds after power on and changes the output level of the signal to low when an object of approximately 35°C such as a human move. When preparing application circuits, make sure to design them to satisfy the electrical characteristics.
2. **Operation Check Conditions**

The sample code contained in this application note has been checked under the conditions listed in the table below.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microcontroller used</td>
<td>RL78/G10 (R5F10Y47ASP)</td>
</tr>
<tr>
<td>Operating frequency</td>
<td>• High-speed on-chip oscillator (HOCO) clock: 1.25 MHz</td>
</tr>
<tr>
<td></td>
<td>• CPU/peripheral hardware clock: 1.25 MHz</td>
</tr>
<tr>
<td>Operating voltage</td>
<td>3.3 V (can run on a voltage range of 2.7 V to 5.5 V)</td>
</tr>
<tr>
<td></td>
<td>SPOR operation: Max 2.90 V at rise(2.76~3.02V)</td>
</tr>
<tr>
<td></td>
<td>Min 2.84 V at fall(2.70~2.96V)</td>
</tr>
<tr>
<td>Integrated development environment</td>
<td>CS+ for CC V6.01.00 from Renesas Electronics Corp.</td>
</tr>
<tr>
<td>(CS+)</td>
<td>C compiler (CS+) CC-RL V1.06.00 from Renesas Electronics Corp.</td>
</tr>
<tr>
<td>Integrated development environment</td>
<td>e² studio V5.1.0.022 from Renesas Electronics Corp.</td>
</tr>
<tr>
<td>(e² studio)</td>
<td>C compiler (e² studio) CC-RL V1.06.00 from Renesas Electronics Corp.</td>
</tr>
</tbody>
</table>

Caution: The code in this application note applies only to the RL78/G10 (R5F10Y47ASP).
3. Hardware Descriptions

3.1 Hardware Configuration

Figure 3.1 shows an example of the hardware configuration for the system described in this application note.

Figure 3.1 the hardware configuration

Notes:
1. The above figure is a simplified circuit image for showing the outline of the connections. The actual circuit should be designed so that the pins are handled appropriately and that the electrical characteristics are satisfied (input-only ports should be each connected to VDD or VSS via a resistor).

2. VDD must be equal to or greater than the reset release voltage (VSPOR) specified with SPOR.

3.2 List of Pins used

Table 3.1 lists the pins used and their functions.

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>I/O</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P137</td>
<td>Input</td>
<td>Signal input from human sensor</td>
</tr>
<tr>
<td>P07</td>
<td>Input</td>
<td>Selection of hand soap discharge amount(P07=H:Many,P07=L:Few)</td>
</tr>
<tr>
<td>P05/TO02</td>
<td>Output</td>
<td>LED drive</td>
</tr>
<tr>
<td>P04/TO01</td>
<td>Output</td>
<td>Fan drive</td>
</tr>
<tr>
<td>P03/ANI2</td>
<td>Input</td>
<td>Battery voltage measurement</td>
</tr>
<tr>
<td>P02/PCLBUZ0</td>
<td>Output</td>
<td>Buzzer alarm</td>
</tr>
<tr>
<td>P01</td>
<td>Output</td>
<td>Battery voltage measurement circuit control</td>
</tr>
<tr>
<td>P00</td>
<td>Output</td>
<td>Human sensor power control</td>
</tr>
</tbody>
</table>
4. Software Descriptions

4.1 Operation Summary

When the hand soap auto dispenser is turned on, the RL78/G10 enters the STOP mode. The STOP mode is canceled with every 50ms of 12-bit interval timer interrupt, then the human sensor detects the hand. When the human sensor output level (P137 input voltage level) is low, it is determined that a hand has been detected.

When a hand is detected, the battery voltage is checked using an A/D converter. If the battery voltage is less than 4.8V (lower limit of motor drive voltage), the buzzer will alert you for 2 seconds and return to STOP mode. When the battery voltage is 4.8V or more, the LED blinks and the motor is driven to discharge the hand soap for 2 seconds. Then return to STOP mode.

4.2 List of Option Byte Setting

Table 4.1 shows the option byte settings.

<table>
<thead>
<tr>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000C0H</td>
<td>11101001B</td>
<td>Disables the watchdog timer. (Stops counting after the release from the reset state.)</td>
</tr>
<tr>
<td>000C1H</td>
<td>11110111B</td>
<td>SPOR detection voltage: 2.90 V at fall; 2.84 V at rise</td>
</tr>
<tr>
<td>000C2H</td>
<td>11111101B</td>
<td>HOCO: 1.25 MHz</td>
</tr>
<tr>
<td>000C3H</td>
<td>10000101B</td>
<td>Enables the on-chip debugger.</td>
</tr>
</tbody>
</table>

4.3 List of Variables

Table 4.2 lists the global variables.

<table>
<thead>
<tr>
<th>Type</th>
<th>Variable Name</th>
<th>Contents</th>
<th>Function Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>uint8_t</td>
<td>g_count</td>
<td>Timer counter</td>
<td>r_tau0_channel3_interrupt()</td>
</tr>
<tr>
<td>uint8_t</td>
<td>g_edge</td>
<td>Hand detection flag</td>
<td>main()</td>
</tr>
<tr>
<td>uint16_t</td>
<td>g_adc_ResultT</td>
<td>A/D conversion result</td>
<td>main()</td>
</tr>
</tbody>
</table>
### 4.4 List of Functions (Subroutines)

Lists the functions (subroutines).

<table>
<thead>
<tr>
<th>Function (Subroutine) Name</th>
<th>Outline</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_MAIN_UserInit</td>
<td>User application initialization</td>
</tr>
<tr>
<td>R_ADC_Start</td>
<td>A/D converter start</td>
</tr>
<tr>
<td>R_ADC_Get_Result()</td>
<td>Function of getting temperature data</td>
</tr>
<tr>
<td>R_ADC_Stop()</td>
<td>A/D converter stop</td>
</tr>
<tr>
<td>R_PCLBUZ0_Start()</td>
<td>Buzzer alarm start</td>
</tr>
<tr>
<td>R_PCLBUZ0_Stop()</td>
<td>Buzzer alarm stop</td>
</tr>
<tr>
<td>R_TAU0_Channel0_Start()</td>
<td>PWM output start and LED flashing start</td>
</tr>
<tr>
<td>R_TAU0_Channel0_Stop()</td>
<td>PWM output stop and LED flashing stop</td>
</tr>
<tr>
<td>R_TAU0_Channel3_Start()</td>
<td>Buzzer alarm time and discharge time start</td>
</tr>
<tr>
<td>R_TAU0_Channel3_Stop()</td>
<td>Buzzer alarm time and discharge time stop</td>
</tr>
</tbody>
</table>

Note: These functions are automatically generated by the integrated development environment.
4.5 Function Specifications

This section gives the specifications of the functions used in the sample program.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Synopsis</th>
<th>Header</th>
<th>Declaration</th>
<th>Explanation</th>
<th>Arguments</th>
<th>Return value</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_MAIN_UserInit</td>
<td>User Application Initialization</td>
<td>r_cg_macrodriver.h</td>
<td>void R_MAIN_UserInit(void)</td>
<td>Performs initialization necessary for the operation of the application.</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>R_ADC_Start</td>
<td>A/D converter conversion start</td>
<td>r_cg_adc.h</td>
<td>void R_ADC_Start(void)</td>
<td>Enables A/D converter conversion.</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>R_ADC_Get_Result</td>
<td>Temperature data get</td>
<td>r_cg_macrodriver.h</td>
<td>void R_ADC_Get_Result (void)</td>
<td>A/D converter result</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>R_ADC_Stop</td>
<td>A/D converter conversion stop</td>
<td>r_cg_macrodriver.h</td>
<td>void R_ADC_Stop(void)</td>
<td>Enables A/D converter conversion.</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
### [Function Name] R_PCLBUZ0_Start

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Buzzer alarm start</th>
</tr>
</thead>
</table>
| Header   | r_cg_macrodriver.h  
|          | r_cg_pclbuz.h      
|          | r_cg_userdefine.h |
| Declaration | void R_PCLBUZ0_Start(void) |
| Explanation | Buzzer alarm start. |
| Arguments | None |
| Return value | None |
| Remarks | None |

### [Function Name] R_PCLBUZ0_Stop

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Buzzer alarm stop</th>
</tr>
</thead>
</table>
| Header   | r_cg_macrodriver.h  
|          | r_cg_pclbuz.h      
|          | r_cg_userdefine.h |
| Declaration | void R_PCLBUZ0_Stop(void) |
| Explanation | Buzzer alarm stop. |
| Arguments | None |
| Return value | None |
| Remarks | None |

### [Function Name] R_TAU0_Channel0_Start

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Timer array unit channel0 start</th>
</tr>
</thead>
</table>
| Header   | r_cg_macrodriver.h  
|          | r_cg_tau.h             
|          | r_cg_userdefine.h |
| Declaration | void R_TAU0_Channel0_Start (void) |
| Explanation | Timer array unit channel0 start |
| Arguments | None |
| Return value | None |
| Remarks | None |

### [Function Name] R_TAU0_Channel0_Stop

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Timer array unit channel0 stop</th>
</tr>
</thead>
</table>
| Header   | r_cg_macrodriver.h  
|          | r_cg_tau.h             
|          | r_cg_userdefine.h |
| Declaration | void R_TAU0_Channel0_Stop (void) |
| Explanation | Timer array unit channel0 stop |
| Arguments | None |
| Return value | None |
| Remarks | None |
### [Function Name] R_TAU0_Channel3_Start

<table>
<thead>
<tr>
<th><strong>Synopsis</strong></th>
<th>Timer array unit channel3 start</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Header</strong></td>
<td>r_cg_macrodriver.h</td>
</tr>
<tr>
<td></td>
<td>r_cg_tau.h</td>
</tr>
<tr>
<td></td>
<td>r_cg_userdefine.h</td>
</tr>
<tr>
<td><strong>Declaration</strong></td>
<td>void R_TAU0_Channel3_Start (void)</td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>Timer array unit channel3 start</td>
</tr>
<tr>
<td><strong>Arguments</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Return value</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Remarks</strong></td>
<td>None</td>
</tr>
</tbody>
</table>

### [Function Name] R_TAU0_Channel3_Stop

<table>
<thead>
<tr>
<th><strong>Synopsis</strong></th>
<th>Timer array unit channel3 stop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Header</strong></td>
<td>r_cg_macrodriver.h</td>
</tr>
<tr>
<td></td>
<td>r_cg_tau.h</td>
</tr>
<tr>
<td></td>
<td>r_cg_userdefine.h</td>
</tr>
<tr>
<td><strong>Declaration</strong></td>
<td>void R_TAU0_Channel3_Stop (void)</td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>Timer array unit channel3 stop</td>
</tr>
<tr>
<td><strong>Arguments</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Return value</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Remarks</strong></td>
<td>None</td>
</tr>
</tbody>
</table>

### [Function Name] main

<table>
<thead>
<tr>
<th><strong>Synopsis</strong></th>
<th>Main function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Header</strong></td>
<td>—</td>
</tr>
<tr>
<td><strong>Declaration</strong></td>
<td>Main processing function for the sample codes</td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Arguments</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Return value</strong></td>
<td>None</td>
</tr>
</tbody>
</table>
4.6 Flowcharts

Figure 4.1 shows an overall flow of the sample program described in this application note.

![Flowchart](image)

**Figure 4.1 The overall flow**

4.6.1 Initial Setting Function

Figure 4.2 shows the flowchart of the initial setting function.

![Flowchart](image)

**Figure 4.2 Initial Setting Function**
### 4.6.2 System Function

Figure 4.3 shows the flowchart of the system function.

![Diagram of System Function](image)

- **R_Systeminit()**
- **Set peripheral I/O redirection**
  - PIOR register ← 0x00
- **Set port**
  - R_PORT_Create()
- **Make initial setting of CPU clock**
  - R_CGC_Create()
- **Set timer array unit**
  - R_TAU0_Create()
- **Set buzzer**
  - R_PCLBUZ0_Create()
- **Set ADC**
  - R_ADC_Create()
- **Set 12-bit interval timer**
  - R_IT_Create()

**return**
4.6.3 I/O Port Setup

Figure 4.4 shows the flowchart for setting up the I/O ports.

```
R_PORT_Create()
  Set port register
    P0.0 bit ← 00H: Set 0 to the output latch of P00
    P0.1 bit ← 00H: Set 0 to the output latch of P01
  Set mode control register
    PMC0.0 bit ← 0: Set digital I/O port
    PMC0.1 bit ← 0: Set digital I/O port
  Set mode register
    PM0.0 bit ← 0: Set output mode
    PM0.1 bit ← 0: Set output mode
    PM0.7 bit ← 1: Set input mode
return
```

**Figure 4.4 I/O Port Setup**

Notes:
1. For details on register setting when using the ports as the alternate functions of the peripheral functions, refer to the RL78/G10 User’s Manual: Hardware.
2. Provide proper treatment for unused pins so that their electrical specifications are observed. Connect each of unused input-only ports to VDD or VSS via a separate resistor.
4.6.4 CPU Clock Setup

Figure 4.5 shows the flowchart for setting up the CPU clock.

```
R_CGC_Create()

Set high-speed system clock/subsystem clock

Set clock operation control

Set main system clock

Set operation speed mode

Set High speed on-chip oscillator clock

return

CMC register ← 00H
X1,X2: input port mode

MSTOP bit ← 1
X1 oscillation circuit stop

MCM0 bit ← 0
fiH selection

OSMC register ← 10H
fiL Supply

HIOSTOP bit ← 0
```

Figure 4.5 CPU Clock Setup
4.6.5 Timer array unit setup

Figure 4.6, Figure 4.7 shows the flowchart for setting up the timer array unit setup.

![Diagram of timer array unit setup]

**Figure 4.6 the timer array unit setup (1/2)**
Initial setting of channel 2
- Operation clock: CK00
- Operation mode: Interval timer
  mode(PWM output(slave))
- Simultaneous channel operation function
- Software trigger start
- Set channel 2 output to 0
- Output of channel 2 is enabled

Initial setting of channel 3
- Operation clock: CK00
- Operation mode: Interval timer
  mode(Interval timer)
- Simultaneous channel operation function
- Software trigger start
- Set channel 3 output to 0
- Output of channel 3 is enabled

Set P04 to output port

Set P05 to output port

return

Figure 4.7 the timer array unit setup (2/2)
4.6.6 Buzzer alarm setup

Figure 4.8 shows the flowchart for setting up the buzzer alarm setup.

```
R_PCLBUZ0_Create()

Buzzer port output prohibited

Output clock setting for buzzer port

Buzzer port set output

return

PCLBUZ0 bit ← 0

CKS0 register ← 02H

PMC0 register ← FBH
P0 register ← FBH
PM0 register ← FBH
```

Figure 4.8 Buzzer alarm setup
4.6.7 A/D converter Setup

Figure 4.9 shows the flowchart for setting up the A/D converter.

```
R_ADC_Create()
  Supply clock to A/D converter
    ADCEN bit ← 1: Start supply of input clock
  Initialize A/D converter
    "Set conversion time to about 4.6 μs"
    ADM0 register ← 00H
    ADMK bit ← 1
    ADIF bit ← 0
  Disabling INTAD interrupt
    Clear INTAD interrupt disabling flag
  Set A/D converter conversion of channel ANI1
    PMC0 register ← 0x08
    PM0 register ← 0x08
    ADM0 register ← 18H
    ADM2 register ← 00H
    ADS register ← 0x02
  Specify analog input channel
  Enables A/D voltage comparator operation
    (A/D conversion operation)
    ADCE bit ← 1
  return
```

Figure 4.9 A/D Converter Setup
### 4.6.8 12-bit interval timer setup

Figure 4.10 shows the flowchart for setting up the 12-bit interval timer setup.

![Flowchart of 12-bit interval timer setup](image)

- **R_IT_Create()**
- Stop input clock supply
- Interval timer control setting
- Enable interrupt mask flag
- Clear interrupt request flag
- Set INTIT priority
- return

**Figure 4.10 12-bit Interval timer Setup**

- TMKAEN ← 1
- ITMCH register ← 00H
- ITM ← 1
- ITIF ← 0
- ITPR1 ← 1
- ITPR0 ← 1
- ITMCH ← 0x02
- ITMCL ← 0xED
4.6.9 Main Processing

Figure 4.11 shows the flowchart of the main processing.

Figure 4.11 Main Processing
4.6.10 Initialization Function
Figure 4.12 shows the flowchart for setting the initialization.

![Figure 4.12 Initialization Function](image)

4.6.11 12-bit interval timer start Setting
Figure 4.13 shows the flowchart for setting the 12-bit interval timer start.

![Figure 4.13 12-bit Interval timer](image)
4.6.12 12-bit interval timer interrupt

Figure 4.14 shows the flowchart for setting the 12-bit interval timer interrupt.

![Flowchart](image)

**Figure 4.14 12-bit interval timer interrupt**
4.6.13 A/D Conversion Start Setting

Figure 4.15 shows the flowchart for setting the A/D conversion start.

![Flowchart for A/D Conversion Start Setting](image)

4.6.14 Obtaining battery voltage

Figure 4.16 shows the flowchart for obtaining battery voltage.

![Flowchart for Obtaining Battery Voltage](image)
4.6.15 A/D Conversion Stop Setting

Figure 4.17 shows the flowchart for A/D conversion stop setting.

```
R_ADC_Stop()

A/D converter operation prohibited

Enable INTAD interrupt

Clear INTAD interrupt flag

return

ADCS bit ← 0

ADMK bit ← 1

ADIF bit ← 0
```

Figure 4.17 A/D conversion stop

4.6.16 Timer array unit channel 0 operation start

Figure 4.18 shows the flowchart for timer array unit channel 0 operation start.

```
R_TAU0_Channel0_Start()

Channel output enable

Operation enable for channel0, 1, 2 of timer array unit

return

TOE0 register ← 0x06

TS0 register ← 0x07
```

Figure 4.18 Timer array unit channel 0 operation start
4.6.17 Timer array unit channel 0 operation stop

Figure 4.19 shows the flowchart for timer array unit channel 0 operation stop.

![Flowchart for Timer array unit channel 0 operation stop]

4.6.18 Timer array unit channel 3 operation start

Figure 4.20 shows the flowchart for timer array unit channel 3 operation start.

![Flowchart for Timer array unit channel 3 operation start]
4.6.19 Timer array unit channel 3 operation stop

Figure 4.21 shows the flowchart for timer array unit channel 3 operation stop.

```
R_TAU0_Channel3_Stop()

Operation prohibited

Disable interrupt

Clear interrupt flag

return
```

Figure 4.21 Timer array unit channel 3 operation stop

4.6.20 Buzzer alarm start

Figure 4.22 shows the flowchart for buzzer alarm start.

```
R_PCLBUZ0_Start()

Start buzzer output

return
```

Figure 4.22 Buzzer alarm start
4.6.21 Buzzer alarm stop

Figure 4.23 shows the flowchart for buzzer alarm stop.

```
R_PCLBUZ0_Stop()

Stop buzzer output

PCLOE0 ← 0

return
```

Figure 4.23 Buzzer alarm stop
Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact/

All trademarks and registered trademarks are the property of their respective owners.
## Revision History

<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Description</th>
<th>Page</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>Sep.30.2019</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
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 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.

5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each product depend 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.

6. 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 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 or other Renesas Electronics document.

7. 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 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 or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document.

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 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 or other Renesas Electronics document.

9. 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 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. 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. 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.

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

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/

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

© 2019 Renesas Electronics Corporation. All rights reserved.