# Applilet3 for RL78 V1.00.00

Release Note

R20UT2750EJ0100
Rev.1.00
September 27, 2013

## Contents

# 1. Introduction

Applilet3 for RL78 is a software tool to generate device driver code for on-chip peripherals. It generates device driver codes using user settings through GUI. Initialize code and API functions are provided.

# 2. Target Devices

| Below is a list of devices supported by the Applilet3 for RL78/I1A V2.00.00.04 | |
|---|---|
| PIN | Device name |
| 20pin | R5F1076C |
| 30pin | R5F107AC, R5F107AE |
| 32pin | R5F107BC |
| 38pin | R5F107DE |
| The Applilet3 is based on the following documents. | |
| Manual Name | Document Number |
| RL78/I1A User's Manual: Hardware | R01UH0169JJ0002 Rev.0.02 |
| | R01UH0169EJ0002 Rev.0.02 |

| Below is a list of devices supported by the Applilet3 for RL78/G12 V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 20pin | R5F10266, R5F10267, R5F10268, R5F10269, R5F1026A<br>R5F10366, R5F10367, R5F10368, R5F10369, R5F1036A |
| 24pin | R5F10277, R5F10278, R5F10279, R5F1027A<br>R5F10377, R5F10378, R5F10379, R5F1037A |
| 30pin | R5F102A7, R5F102A8, R5F102A9, R5F102AA<br>R5F103A7, R5F103A8, R5F103A9, R5F103AA |
| The Applilet3 is based on the following documents. | |
| Manual Name | Document Number |
| RL78/G12 User's Manual: Hardware | R01UH0200JJ0100 Rev.1.00 |
| | R01UH0200EJ0100 Rev.1.00 |

| Below is a list of devices supported by the Applilet3 for RL78/G13 V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 20pin | R5F1006A, R5F1006C, R5F1006D, R5F1006E<br>R5F1016A, R5F1016C, R5F1016D, R5F1016E |
| 24pin | R5F1007A, R5F1007C, R5F1007D, R5F1007E<br>R5F1017A, R5F1017C, R5F1017D, R5F1017E |
| 25pin | R5F1008A, R5F1008C, R5F1008D, R5F1008E<br>R5F1018A, R5F1018C, R5F1018D, R5F1018E |
| 30pin | R5F100AA, R5F100AC, R5F100AD, R5F100AE, R5F100AF, R5F100AG<br>R5F101AA, R5F101AC, R5F101AD, R5F101AE, R5F101AF, R5F101AG |
| 32pin | R5F100BA, R5F100BC, R5F100BD, R5F100BE, R5F100BF, R5F100BG<br>R5F101BA, R5F101BC, R5F101BD, R5F101BE, R5F101BF, R5F101BG |
| 36pin | R5F100CA, R5F100CC, R5F100CD, R5F100CE, R5F100CF, R5F100CG<br>R5F101CA, R5F101CC, R5F101CD, R5F101CE, R5F101CF, R5F101CG |
| 40pin | R5F100EA, R5F100EC, R5F100ED, R5F100EE, R5F100EF, R5F100EG, R5F100EH<br>R5F101EA, R5F101EC, R5F101ED, R5F101EE, R5F101EF, R5F101EG, R5F101EH |
| 44pin | R5F100FA, R5F100FC, R5F100FD, R5F100FE, R5F100FF, R5F100FG, R5F100FH<br>R5F100FJ, R5F100FK, R5F100FL<br>R5F101FA, R5F101FC, R5F101FD, R5F101FE, R5F101FF, R5F101FG, R5F101FH<br>R5F101FJ, R5F101FK, R5F101FL |
| 48pin | R5F100GA, R5F100GC, R5F100GD, R5F100GE, R5F100GF, R5F100GG, R5F100GH<br>R5F100GJ, R5F100GK, R5F100GL<br>R5F101GA, R5F101GC, R5F101GD, R5F101GE, R5F101GF, R5F101GG, R5F101GH<br>R5F101GJ, R5F101GK, R5F101GL |
| 52pin | R5F100JC, R5F100JD, R5F100JE, R5F100JF, R5F100JG, R5F100JH<br>R5F100JJ, R5F100JK, R5F100JL<br>R5F101JC, R5F101JD, R5F101JE, R5F101JF, R5F101JG, R5F101JH<br>R5F101JJ, R5F101JK, R5F101JL |
| 64pin | R5F100LC, R5F100LD, R5F100LE, R5F100LF, R5F100LG, R5F100LH<br>R5F100LJ, R5F100LK, R5F100LL<br>R5F101LC, R5F101LD, R5F101LE, R5F101LF, R5F101LG, R5F101LH<br>R5F101LJ, R5F101LK, R5F101LL |
| 80pin | R5F100MF, R5F100MG, R5F100MH, R5F100MJ, R5F100MK, R5F100ML<br>R5F101MF, R5F101MG, R5F101MH, R5F101MJ, R5F101MK, R5F101ML |
| 100pin | R5F100PF, R5F100PG, R5F100PH, R5F100PJ, R5F100PK, R5F100PL<br>R5F101PF, R5F101PG, R5F101PH, R5F101PJ, R5F101PK, R5F101PL |
| 128pin | R5F100SH, R5F100SJ, R5F100SK, R5F100SL<br>R5F101SH, R5F101SJ, R5F101SK, R5F101SL |

The Applilet3 is based on the following documents.

| Manual Name | Document Number |
|---|---|
| RL78/G13 User's Manual: Hardware | R01UH0146JJ0210 Rev.1.00 |
| | R01UH0146EJ0210 Rev.1.00 |

| Below is a list of devices supported by the Applilet3 for RL78/G14 V2.01.00.02 | |
|---|---|
| PIN | Device name |
| 30pin | R5F104AA, R5F104AC, R5F104AD, R5F104AE, R5F104AF, R5F104AG |
| 32pin | R5F104BA, R5F104BC, R5F104BD, R5F104BE, R5F104BF, R5F104BG |
| 36pin | R5F104CA, R5F104CC, R5F104CD, R5F104CE, R5F104CF, R5F104CG |
| 40pin | R5F104EA, R5F104EC, R5F104ED, R5F104EE, R5F104EF, R5F104EG, R5F104EH |
| 44pin | R5F104FA, R5F104FC, R5F104FD, R5F104FE, R5F104FF, R5F104FG, R5F104FH R5F104FJ |
| 48pin | R5F104GA, R5F104GC, R5F104GD, R5F104GE, R5F104GF, R5F104GG, R5F104GH R5F104GJ |
| 52pin | R5F104JC, R5F104JD, R5F104JE, R5F104JF, R5F104JG, R5F104JH, R5F104JJ |
| 64pin | R5F104LC, R5F104LD, R5F104LE, R5F104LF, R5F104LG, R5F104LH, R5F104LJ |
| 80pin | R5F104MF, R5F104MG, R5F104MH, R5F104MJ |
| 100pin | R5F104PF, R5F104PG, R5F104PH, R5F104PJ |

| The Applilet3 is based on the following documents. | |
|---|---|
| Manual Name | Document Number |
| RL78/G14 User's Manual: Hardware | R01UH0186JJ0100 Rev.1.00 |
| | R01UH0186EJ0100 Rev.1.00 |

| Below is a list of devices supported by the Applilet3 for RL78/G1A V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 25pin | R5F10E8A, R5F10E8C, R5F10E8D, R5F10E8E |
| 32pin | R5F10EBA, R5F10EBC, R5F10EBD, R5F10EBE |
| 48pin | R5F10EGA, R5F10EGC, R5F10EGD, R5F10EGE |
| 64pin | R5F10ELC, R5F10ELD, R5F10ELE |

| The Applilet3 is based on the following documents. | |
|---|---|
| Manual Name | Document Number |
| RL78/G1A User's Manual: Hardware | R01UH0305JJ0002 Rev.0.02 |
| | R01UH0305EJ0002 Rev.0.02 |

| Below is a list of devices supported by the Applilet3 for RL78/F12 V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 20pin | R5F109AE, R5F109AD, R5F109AC, R5F109AB, R5F109AA |
| 30pin | R5F109BE, R5F109BD, R5F109BC, R5F109BB, R5F109BA |
| 32pin | R5F109BE, R5F109BD, R5F109BC, R5F109BB, R5F109BA |
| 48pin | R5F109GE, R5F109GD, R5F109GC, R5F109GB, R5F109GA |
| 64pin | R5F109LE, R5F109LD, R5F109LC, R5F109LB, R5F109LA |
| The Applilet3 is based on the following documents. | |
| Manual Name | Document Number |
| RL78/F12 User's Manual: Hardware | R01UH0231JJ0003 Rev.0.03 |
| | R01UH0231EJ0003 Rev.0.03 |

| Below is a list of devices supported by the Applilet3 for RL78/L12 V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 32pin | R5F10RBC, R5F10RBA, R5F10RB8 |
| 44pin | R5F10RFC, R5F10RFA, R5F10RF8 |
| 48pin | R5F10RGC, R5F10RGA, R5F10RG8 |
| 52pin | R5F10RJC, R5F10RJA, R5F10RJ8 |
| 64pin | R5F10RLC, R5F10RLA |
| The Applilet3 is based on the following documents | |
| Manual Name | Document Number |
| RL78/L12 User's Manual: Hardware | R01UH0330JJ0003 Rev.0.03 |
| | R01UH0330EJ0003 Rev.0.03 |

| Below is a list of devices supported by the Applilet3 for RL78/D1A V2.01.00.01 | |
|---|---|
| PIN | Device name |
| 48pin | R5F10CGB, R5F10CGC, R5F10DGC, R5F10DGD, R5F10DGE |
| 64pin | R5F10CLD, R5F10DLD, R5F10DLE |
| 80pin | R5F10CMD, R5F10CME<br>R5F10DMD, R5F10DME, R5F10DMF, R5F10DMG, R5F10DMJ |
| 100pin | R5F10DPE, R5F10DPF, R5F10DPG, R5F10DPJ, R5F10TPJ |

| Applilet3 for RL78_D1A is based on the following documents | |
|---|---|
| Manual Name | Document Number |
| RL78/D1A User's Manual: Hardware | R01UH0317EJ0003 Rev.0.03 |

# 3. Operating Environment

Host machine

- IBM PC/AT compatibles (Windows® 7, Windows Vista®)
- Processor: 1 GHz or higher (must support hyper-threading, multi-core CPUs)
- Memory capacity: 2 GB or more recommended. Minimum requirement is 1 GB or more (64-bit Windows requires 2 G or more)
- Hard disk capacity: 200 MB or more spare capacity
- Display: 1024 x 768 or higher resolution, 65,536 or more colors
- Interface: USB 2.0
- All other necessary software environments in addition to WindowsOS
  - .NET Framework version4.0
  - Microsoft Visual C++ 2010 SP1 runtime library

▪ Development Environments

| Product Name | Version |
|---|---|
| IAR Embedded Workbench for Renesas RL78 | V1.30 or later |
| KPIT GNURL78 | V13.01 or later |

# 4. Changes

This chapter describes changes for RL78.

| No | 内容 | Corresponds of code generation | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | RL78/L12 V2.01.00.01 | RL78/F12 V2.01.00.01 | RL78/G1A V2.01.00.01 | RL78/G12 V2.01.00.01 | RL78/G13 V2.01.00.01 | RL78/G14 V2.01.00.02 | ~~RL78/H1A V2.01.00.02~~ | RL78/D1A V2.01.00.01 |
| 1 | Output code changes of real-time clock | / | / | / | - | - | - | ~~=~~ | / |
| 2 | Output code changes of serial array unit | / | / | / | - | - | - | ~~=~~ | / |
| 3 | Addition of PMC register setup | / | / | / | - | - | - | ~~≠~~ | / |
| 4 | Output code changes of the receiving function of UARTn | / | / | / | - | - | - | ~~=~~ | / |
| 5 | Control correction of a competition pin | / | / | / | / | - | / | ~~≠~~ | / |
| 6 | GUI correction of an A/D converter | / | / | / | - | - | - | ~~≠~~ | / |
| 7 | Changes of square wave output of a timer | / | / | / | / | / | / | ~~≠~~ | / |
| 8 | Changes of the TTL check box of a port | / | / | / | / | / | / | ~~≠~~ | / |
| 9 | Changes of PIOR setup | / | / | / | / | / | / | ~~≠~~ | / |
| 10 | Changes of TAU1 setup | / | / | / | / | / | / | ~~≠~~ | / |
| 11 | Changes of UART2 setup | / | / | / | / | / | / | ~~≠~~ | / |
| 12 | Changes of key interrupt function | / | / | / | / | / | / | ~~≠~~ | / |
| 13 | Changes of Simplified I2C | / | / | / | / | / | / | ~~≠~~ | / |
| 14 | Additional function generation file mode | O | O | O | O | O | O | ~~O~~ | O |
| 15 | Changes of hdwinit() function | O | O | O | O | O | O | ~~O~~ | O |
| 16 | Changes of API about Simplified I2C by SAU | / | / | / | / | / | / | ~~≠~~ | / |
| 17 | Changes of the timer RD and the timer RJ0 | / | / | / | / | / | - | ~~≠~~ | / |
| 18 | Changes of input pulse interval measurement of TAU | - | - | - | - | - | - | ~~=~~ | - |
| 19 | Changes the notation of a power supply | / | / | / | - | / | / | ~~≠~~ | / |

○ : Correspondence, -: Not correspondence(finish of correction), /: Outside of function

## 4.1 Details of Changes

### 4.1.1 Output code changes of real-time clock

a) The output code of R_RTC_Set_ConstPeriodInterruptOff() was changes.

**Before:**

```
void R_RTC_Set_ConstPeriodInterruptOff(void)
{
      RTCC0 &= _88_RTC_INTRTC_CLEAR;
      RTCIF = 0U;             /* clear INTRTC interrupt flag */
}
```

**After:**

```
void R_RTC_Set_ConstPeriodInterruptOff(void)
{
   RTCC0 &= _F8_RTC_INTRTC_CLEAR;
   RTCC1 &= (uint8_t)~_08_RTC_INTC_GENERATE_FLAG;
   RTCIF = 0U;       /* clear INTRTC interrupt flag */

}
```

b) The output code of R_RTC_Interrupt( ) was changes.
- When an alarm interrupt function is checked.

**Before:**

```
__interrupt void R_RTC_Interrupt(void)
{
  R_RTC_Callback_Alarm();
}
```

**After:**

```
__interrupt static void r_rtc_interrupt(void)
{
   if (1U == WAFG)
   {
      RTCC1 &= (uint8_t)~_10_RTC_ALARM_MATCH;       /* clear WAFG */
      r_rtc_callback_alarm();
   }

}
```

- When a constant-period interruption function is checked

**Before:**

```
__interrupt void R_RTC_Interrupt(void)

{

  R_RTC_Callback_ConstPeriod();

}
```

**After:**

```
__interrupt static void r_rtc_interrupt(void)
{
   if (1U == RIFG)
   {
      RTCC1 &= (uint8_t)~_08_RTC_INTC_GENERATE_FLAG;    /* clear RIFG */
      r_rtc_callback_constperiod();
   }

}
```

a) and b) issues has been corrected

### 4.1.2 Output code changes of serial array unit

a) The following function which was not supported by SAU1 was added.

```
void R_SAU0_Set_SnoozeOn(void)

void R_SAU0_Set_SnoozeOff(void)
```

b) The function of simple IIC was changed.

**Before:**

```
void R_IIC00_StartCondition(void)
{
    SO0 &= ~_0001_SAU_CH0_DATA_OUTPUT_1; /* clear IIC00 SDA */
    SOE0 |= _0001_SAU_CH0_OUTPUT_ENABLE;  /* enable IIC00 output */
    SO0 &= ~_0100_SAU_CH0_CLOCK_OUTPUT_1;          /* clear IIC00 SCL */
    SS0 |= _0001_SAU_CH0_START_TRG_ON;     /* enable IIC00 */
}
```

**After:**

```
void R_IIC00_StartCondition(void)
{
    volatile uint8_t  w_count;

    SO0 &= ~_0001_SAU_CH0_DATA_OUTPUT_1;    /* clear IIC00 SDA */

    /* Wait for 5us */
    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }
    SO0 &= ~_0100_SAU_CH0_CLOCK_OUTPUT_1;    /* clear IIC00 SCL */
    SOE0 |= _0001_SAU_CH0_OUTPUT_ENABLE;     /* enable IIC00 output */
    SS0 |= _0001_SAU_CH0_START_TRG_ON;       /* enable IIC00 */

}
```

**Before:**

```
void R_IIC00_StopCondition(void)
{
    ST0 |= _0001_SAU_CH0_STOP_TRG_ON;       /* disable IIC00 */
    SOE0 &= ~_0001_SAU_CH0_OUTPUT_ENABLE;          /* disable IIC00 output */
    SO0 &= ~_0001_SAU_CH0_DATA_OUTPUT_1; /* clear IIC00 SDA */
    SO0 |= _0100_SAU_CH0_CLOCK_OUTPUT_1;  /* set IIC00 SCL */
    SO0 |= _0001_SAU_CH0_DATA_OUTPUT_1;    /* set IIC00 SDA */
}
```

**After:**

```
void R_IIC00_StopCondition(void)
{
    volatile uint8_t  w_count;

    ST0 |= _0001_SAU_CH0_STOP_TRG_ON;         /* disable IIC00 */
    SOE0 &= ~_0001_SAU_CH0_OUTPUT_ENABLE;      /* disable IIC00 output */
    SO0 &= ~_0001_SAU_CH0_DATA_OUTPUT_1;       /* clear IIC00 SDA */
    SO0 |= _0100_SAU_CH0_CLOCK_OUTPUT_1; /* set IIC00 SCL */

    /* Wait for 5us */
    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }
    SO0 |= _0001_SAU_CH0_DATA_OUTPUT_1;       /* set IIC00 SDA */

}
```

c) The interrupt handler function of simple IIC was corrected.
- It was made not to take out an error with the last byte's NACK.

**Before:**

```
if ((SSR00 & _0002_SAU_PARITY_ERROR) == 0x0002U)
{
    R_IIC00_Callback_Master_Error(MD_NACK);
}
```

**After:**

```
if (((SSR00 & _0002_SAU_PARITY_ERROR) == 0x0002U) && (g_iic00_tx_count != 0U))
{
r_iic00_callback_master_error(MD_NACK);
}
```

- Deletion of an unnecessary code

**Before:**

```
if ((g_Iic00MasterStatusFlag & _04_SAU_IIC_SENDED_ADDRESS_FLAG) == 0U)
{
    rxadr = SIO00;
    SCR00 &= ~_C000_SAU_RECEPTION_TRANSMISSION;
    SCR00 |= _4000_SAU_RECEPTION;
    g_Iic00MasterStatusFlag |= _04_SAU_IIC_SENDED_ADDRESS_FLAG;
    SIO00 = 0xFFU;
}
```

**After:**

```
if ((g_iic00_master_status_flag & _04_SAU_IIC_SENDED_ADDRESS_FLAG) == 0U)
{
    ST0 |= _0001_SAU_CH0_STOP_TRG_ON;
    SCR00 &= ~_C000_SAU_RECEPTION_TRANSMISSION;
    SCR00 |= _4000_SAU_RECEPTION;
    SS0 |= _0001_SAU_CH0_START_TRG_ON;
    g_iic00_master_status_flag |= _04_SAU_IIC_SENDED_ADDRESS_FLAG;
    SIO00 = 0xFFU;

}
```

a) , b) and c) issues has been corrected .

### 4.1.3 Addition of PMC register setup

A setup of the PMC register was added about the combination terminal of each circumference.
[The terminal to which a PMC register setup was added]
●20,24,25,30, 32pin devices
P00/ANI17/TI00/TxD1
P01/ANI16/TO00/RxD1
●other devices
P02/ANI17/SO10/TxD1
P03/ANI16/SI10/RxD1/SDA10

This issue has been corrected

### 4.1.4 Output code changes of the receiving function of UARTn

The output code of the receiving function of UARTn was corrected. The following is a case of UART0.
[ r_cg_serial.c ]

```
MD_STATUS R_UART0_Receive(uint8_t * const rx_buf, uint16_t rx_num)
{
    MD_STATUS status = MD_OK;

    if (rx_num < 1U)
    {
        status = MD_ARGERROR;
    }
    else
    {
        g_uart0_rx_count = 0U;
        g_uart0_rx_length = rx_num;
        gp_uart0_tx_address = rx_buf;
    }

    return (status);
}
```

**Before** : gp_uart0_tx_address = rx_buf;
**After** : gp_uart0_rx_address = rx_buf;

This issue has been corrected

### 4.1.5 Control correction of a competition pin

Control of the competition pin when simple-I2C of 24 and 25 pin device of RL78/G13 is set up was corrected.

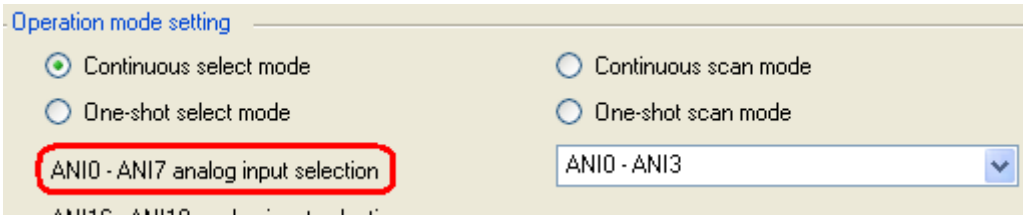
**Before** : P17/SDA11
P30/SCL11
**After** : P50/SDA11
P30/SCL11

This issue has been corrected

### 4.1.6 GUI correction of an A/D converter

In the A/D converter, it corrected so that the message displayed with the fixed value as the number of analog input channels might be dynamically displayed according to a number of channels.

This issue has been corrected




### 4.1.7 Changes of square wave output of a timer

If code is generated so that timers TAUx (x is 1 to 7) of an 80-, 100-, or 128-pin MCU can output square wave, the values of the TOM1 and TOL1 registers, which control TAUx, are not set but those of the TOM0 and TOL0 registers are set.

This issue has been corrected

### 4.1.8 Changes of the TTL check box of a port

There is no check box which sets TTL as P10 and P11 with 30-pin MCU.

This issue has been corrected

### 4.1.9 Changes of PIOR setup

In the code for setting registers PIOR01 and PIOR04 to 1s in an arrangement of pin assignments, incorrect pins are assigned to INTP10 and INTP11 as follows:

| Incorrect: | Correct: |
|---|---|
| P110 assigned to INTP10 | P100 assigned to INTP10 |
| P111 assigned to INTP11 | P110 assigned to INTP11 |

This issue has been corrected

### 4.1.10 Changes of TAU1 setup

If code is generated in an 80- or 100-pin MCU, no one except "interval" can be selected in the functional selection of timer TAU1.

This issue has been corrected

### 4.1.11 Changes of UART2 setup

If the code is generated for making settings of UART2 and any of the ports except 13 and 14, an error arises in building it.
Example:
If you use UART2 and set ports 10, 11, and 12 to the output state, the following code is generated; however, the last "|" is unnecessary:
PMC1 = ・・・ | _80_PMCn7_NOT_USE | ;
If build is performed including this code, an error arises. It must be read as follows:
PMC1 = ・・・ | _80_PMCn7_NOT_USE ;

This issue has been corrected

### 4.1.12 Changes of key interrupt function

If you make settings of the key interrupt flag and the triggering edge, the settings cannot properly be reflected to the KRCTL register by the generated code.

Example:
If you select Use among from the key flag pull-down list and Falling Edge among from the triggering edge pull-down list, the Applilet3 generates the following incorrect codes.

```
KRCTL |= _00_KR_FLAG_UNUSED;
KRCTL |= _01_KR_EDGE_RISING;
```

The correct codes are as follows:

```
KRCTL |= _01_KR_FLAG_USED;
KRCTL |= _00_KR_EDGE_FALLING;
```

This issue has been corrected

### 4.1.13 Changes of Simplified I2C

I When a receiving byte is set to 1 by Simplified I2C, it does not operate normally.

This issue has been corrected

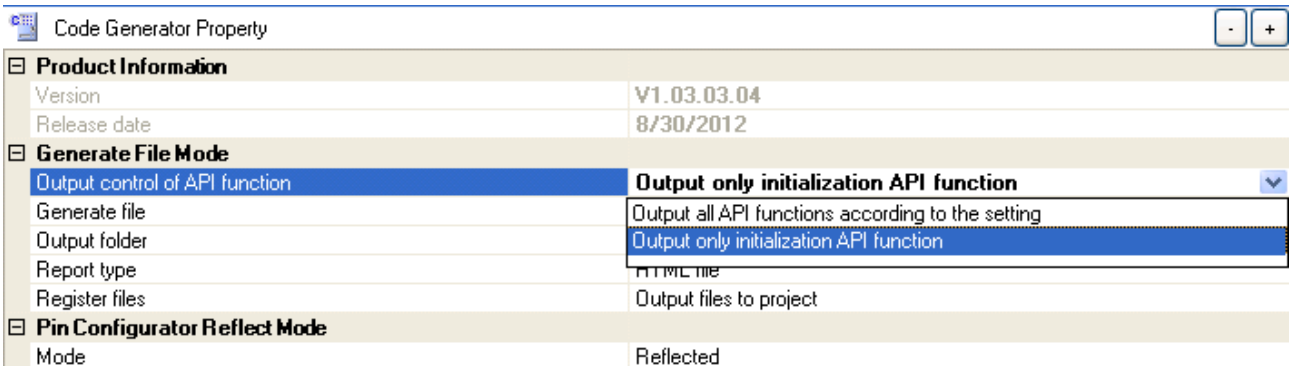
### 4.1.14 Additional function generation file mode

"Output control of API function" has been added to the Applilet3 Property for RL78.

"Output all API functions according to the setting": Outputs necessary API functions according to the GUI settings (conventional output method).

"Output only initialization API function": Outputs only initialization functions (Create functions) regardless of the GUI settings.

Users can configure the settings such as interrupt functions according to their needs.



### 4.1.15 Changes of hdwinit() function

We have changed the initial code for the hdwinit() and main() functions.

```
void hdwinit(void)
{
    DI();
    R_Systeminit();
    EI();
}
```

The above code has been changed to the code given below. Accordingly, interrupts are not enabled within the hdwinit function.

```
void hdwinit(void)
{
    DI();
    R_Systeminit();
}
```

Interrupts are now enabled within the main() function.

```
/***********************************************************************************************************************
* Function Name: main
* Description  : This function implements main function.
***********************************************************************************************************************/
void main(void)
{
    R_MAIN_UserInit();
    /* Start user code. Do not edit comment generated here */
    while (1U)
    {
        ;
    }
    /* End user code. Do not edit comment generated here */
}
/***********************************************************************************************************************
* Function Name: R_MAIN_UserInit
* Description  : This function adds user code before implementing main function.
***********************************************************************************************************************/
void R_MAIN_UserInit(void)
{
    /* Start user code. Do not edit comment generated here */
    EI();
    /* End user code. Do not edit comment generated here */
}
```

When an old project is used in code generation, the definitions of variables within the main function may lead to errors.

[Old project]
```
void main(void)
{
   /* Start user code. Do not edit comment generated here */
   char c;
   while (1U)
   {
      ...
```

[When an old project is loaded into CubeSuite+V1.03.00 and used for code generation]
```
void main(void)
{
   R_MAIN_UserInit();
   /* Start user code. Do not edit comment generated here */
   char c;        <- error!!
   while (1U)
   {
      ...
```

In that case, use { }.
```
void main(void)
{
   R_MAIN_UserInit();
   /* Start user code. Do not edit comment generated here */
   {              <- add "{"
     char c;      <- not error!
     while (1U)
     {
        ...
        }
   }              <- add "}"
```

### 4.1.16 Changes of API about Simplified I2C by SAU

The R_IICmn_StartCondition of Simplified I2C of SAU and the R_IICmn_StopCondition were changed. The waiting code for time required for generation of the start condition of Simplified I2C standard and stop condition was added.

This issue has been corrected

### 4.1.17 Changes of the timer RD and the timer RJ0, and new restriction

a) Change of the timer RD

When the frequency of the high-speed on-chip oscillator clock is 64 MHz, the period and duty cycle for timer RD in PWM mode are incorrect. Although pins multiplexed with PWM are usually marked "!" in the port-setting view to indicate that the other functions are not usable in PWM mode, port pins being used by timer RD are not marked "!".

b) Change of the timer RJ0

Change of the fault in timer RJ pulse period measurement mode.

Specify desired values for the count source and TRJIO0 polarity, and 0xffff for the count value. After code generation by CubeSuite+, correct a part of the output interrupt handler in the following way.

[Output source code]

```
__interrupt static void r_tmr_rj0_interrupt(void)
{
    if ((TRJCR0 & _20_TMRJ_UNDERFLOW_OCCUR) != 0U)
    {
        g_tmrj0_underflow_count += 1U;
        TRJCR0 &= (uint8_t)~_20_TMRJ_UNDERFLOW_OCCUR;
    }
    if ((TRJCR0 & _10_TMRJ_ACTIVE_EDGE_UNRECEIVED) != 0U)
    {
        g_tmrj0_width = (uint32_t)(g_tmrj0_trj_count - TRJ0 + 1U +
                    (g_tmrj0_underflow_count * (_FFFF_TMRJ_TRJ0_VALUE + 1U)));
        g_tmrj0_trj_count = (uint32_t)TRJ0;
        g_tmrj0_underflow_count = 0U;
        TRJCR0 &= (uint8_t)~_10_TMRJ_ACTIVE_EDGE_UNRECEIVED;
    }
}
```

[Correct code]

```
            g_tmrj0_width = (uint32_t)(_FFFF_TMRJ_TRJ0_VALUE - TRJ0 + 1U +
                        (g_tmrj0_underflow_count * (_FFFF_TMRJ_TRJ0_VALUE + 1U)));
```

g_tmrj0_trj_count in the above expression must be the specified count value.

When the count value is 0xffff, for example, enter _FFFF_TMRJ_TRJ0_VALUE.

This issue has been corrected

c) Addition of restriction about timer RJ0

The interruption function in timer RJ0 pulse width measurement mode has fault.

If you select pulse width measurement from among the operating modes of the RJ0 timer, the Applilet3 generates erroneous code

Example:

If you make the above settings for the RJ0 timer and use the Applilet3 under the following settings, errors arise in the generated interrupt handler:

- Count source: any
- Polarity of TRJIO0: positive or negative
- Count value: 0x64

[Output source code]

```
__interrupt static void r_tmr_rj0_interrupt(void)
{
    if ((TRJCR0 & _20_TMRJ_UNDERFLOW_OCCUR) != 0U)
    {
        g_tmrj0_underflow_count += 1U;
        TRJCR0 &= (uint8_t)~_20_TMRJ_UNDERFLOW_OCCUR;
    }

    if ((TRJCR0 & _10_TMRJ_ACTIVE_EDGE_UNRECEIVED) != 0U)
    {
        g_tmrj0_width = (uint32_t)(_0064_TMRJ_TRJ0_VALUE - TRJ0 + 1U +
                    (g_tmrj0_underflow_count * (_0064_TMRJ_TRJ0_VALUE + 1U)));
        g_tmrj0_underflow_count = 0U;
        TRJCR0 &= (uint8_t)~_10_TMRJ_ACTIVE_EDGE_UNRECEIVED;
    }
}
```

[The code to correct ]

```
volatile uint32_t g_tmrj0_trj_count = 0U;    (A global variable is added)

__interrupt static void r_tmr_rj0_interrupt(void)
{
    if ((TRJCR0 & _20_TMRJ_UNDERFLOW_OCCUR) != 0U)
    {
        g_tmrj0_underflow_count += 1U;
        TRJCR0 &= (uint8_t)~_20_TMRJ_UNDERFLOW_OCCUR;
    }

    if ((TRJCR0 & _10_TMRJ_ACTIVE_EDGE_UNRECEIVED) != 0U)
    {
        g_tmrj0_width = (uint32_t)(g_tmrj0_trj_count - TRJ0 + 1U +
                    (g_tmrj0_underflow_count * (_0064_TMRJ_TRJ0_VALUE + 1U)));
        g_tmrj0_trj_count = (uint32_t)TRJ0;
        g_tmrj0_underflow_count = 0U;
       TRJCR0 &= (uint8_t)~_10_TMRJ_ACTIVE_EDGE_UNRECEIVED;
    }
}
```

C) issue has been corrected

### 4.1.18 Changes of input pulse interval measurement of TAU

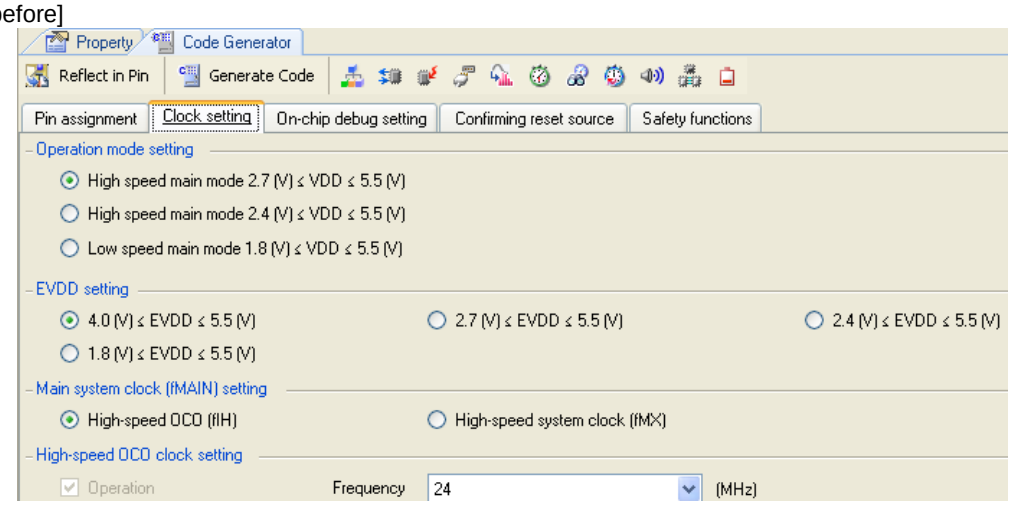
It corrected that the right measured value was not able to be acquired for the input pulse interval measurement function of TAU, and a high / low width measurement function by interruption function r_taux_channelx_interrupt() at the time of use.

This issue has been corrected

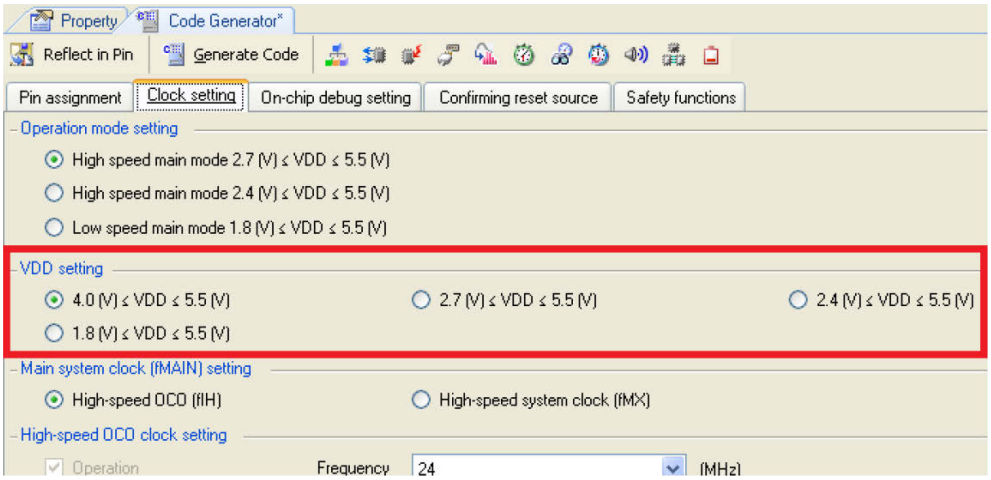
### 4.1.19 Changes the notation of a power supply

Although there was not EVDD in RL78/G12 device, it corrected that the notation remained on GUI.

[before]




[after]




This issue has been corrected in Applilet3 for RL78,78K0R,78K0 V2.00.00.

# 5. Cautions

This section describes cautions for using Applilet3 for RL78

# Chapter 5.

## 5.1 Cautions List

| No. | Description | Corresponds of Applilet3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | RL78/L12 V2.01.00.01 | RL78/F12 V2.01.00.01 | RL78/G1A V2.01.00.01 | RL78/G12 V2.01.00.01 | RL78/G13 V2.01.00.01 | RL78/G14 V2.01.00.02 | ~~RL78/I1A V2.01.00.02~~ | RL78/D1A V2.01.00.01 |
| 1 | Cautions of the LIN-bus function of UART0, UART2, UART3, UART6 or UARTF. | ○ | ○ | ○ | ○ | ○ | ○ | ~~○~~ | ○ |
| 2 | Cautions of extension code, wakeup function and multimaster of serial interface IICA or IIC0 | ○ | ○ | ○ | ○ | ○ | ○ | ~~○~~ | ○ |
| 3 | Cautions of the operation for slave transmission of serial interface IICA or IIC0. | / | / | / | - | - | - | = | - |
| 4 | Restrictions of High-speed on-chip oscillator frequency select register | ○ | ○ | ○ | ○ | ○ | ○ | ~~/~~ | ○ |
| 5 | Restriction of a serial array unit | / | / | / | / | / | / | ~~○~~ | / |
| 6 | Cautions of PORT | / | / | / | - | - | - | = | - |
| 7 | Cautions of setup of a real-time clock | / | / | / | / | - | - | = | / |
| 8 | Cautions when using a DTC function | / | / | / | / | / | ○ | ~~/~~ | / |
| 9 | Cautions of initial function of an A/D converter | / | / | / | - | - | - | ~~/~~ | / |
| 10 | Cautions of initial function at the time of setting up UART transmission | / | / | / | - | - | - | ~~/~~ | / |
| 11 | Cautions of Complementary assistant PWM mode of TimerRD | / | / | / | / | / | - | ~~/~~ | / |
| 12 | Restrictions of internal low-speed or internal high-speed oscillator trimming | ○ | ○ | ○ | ○ | ○ | ○ | ~~○~~ | ○ |
| 13 | Cautions of Safety Functions | ○ | ○ | ○ | ○ | ○ | ○ | ~~○~~ | ○ |
| 14 | Cautions of file merge | ○ | ○ | ○ | ○ | ○ | ○ | ~~○~~ | ○ |

○ : Correspondence, -: Not correspondence, /:Outside of function.

## 5.2 Cautions Details

### 5.2.1 Cautions of the LIN-bus function of UART2 or UART3 or UART6

The Applilet3 is not supporting the LIN-bus functions of serial interface UART2 or UART3 or UART6.
[Workaround] There is no workaround.

### 5.2.2 Cautions of the operation for slave transmission of serial interface IICA or IIC0

The Applilet3 is not supporting the extension code, multimaster, wakeup function of serial interface IIC.
[Workaround] There is no workaround.

### 5.2.3 Cautions of extension code, multimaster,wakeup function of serial interface IICA or IIC0

During slave transmission, if the master receiver does not return an ACK after the final data is received, then the error API IICA_SlaveErrorCallback(MD_NACK) will be called, regardless of whether the actual slave transmission process ended. For this reason, the program will not terminate normally.
[Work-around]
If the master being communicated with does not return an ACK after the final data reception, change IICA_SlaveHandler's internal code as follows. (So that it does not check for an ACK after the final data is received. The figure below for the serial interface IICA.)

```
void IICA_SlaveHandler(void)
{
  ...
    if (TRC0 == 1U)
    {
        if (ACKD0 == 0U)
        {
            IICA_SlaveErrorCallback(MD_NACK);
        }
        else
        {
            if (glicaTxCnt > 0U)
            {
                IICA = *gplicaTxAddress;
                gplicaTxAddress++;
                glicaTxCnt--;
            }
            else
            {
                IICA_SlaveSendEndCallback();
                WREL0 = 1U;
            }
        }
    }
}
```

if ((ACKD0 == 0U) && (glicaTxCnt != 0))

[Workaround] RL78 Applilet3 is corrected

### 5.2.4 Restrictions of High-speed on-chip oscillator frequency select register

Applilet3 is not equivalent to a setup of high-speed on-chip oscillator frequency select register
[Workaround] There is no workaround.

### 5.2.5 Restriction of a serial array unit

Applilet3 is not equivalent to a setup of single-wire UART mode and DMX512 communication [Workaround] There is no workaround.

### 5.2.6 Cautions of PORT

There are notes in the port setting of RL78/G13(R5F100LJ, R5F100LK, R5F100LL).
Please do not use a item of P43, P52, P53, and P54 ("TTL buffer" or "N-ch").
[Workaround] RL78 Applilet3 is corrected

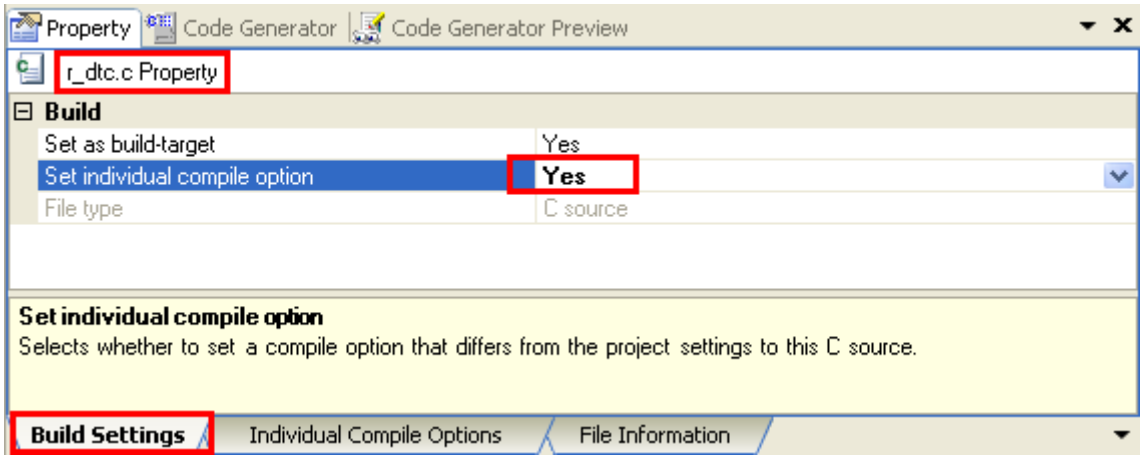
### 5.2.7 Cautions of a setup of a real-time clock

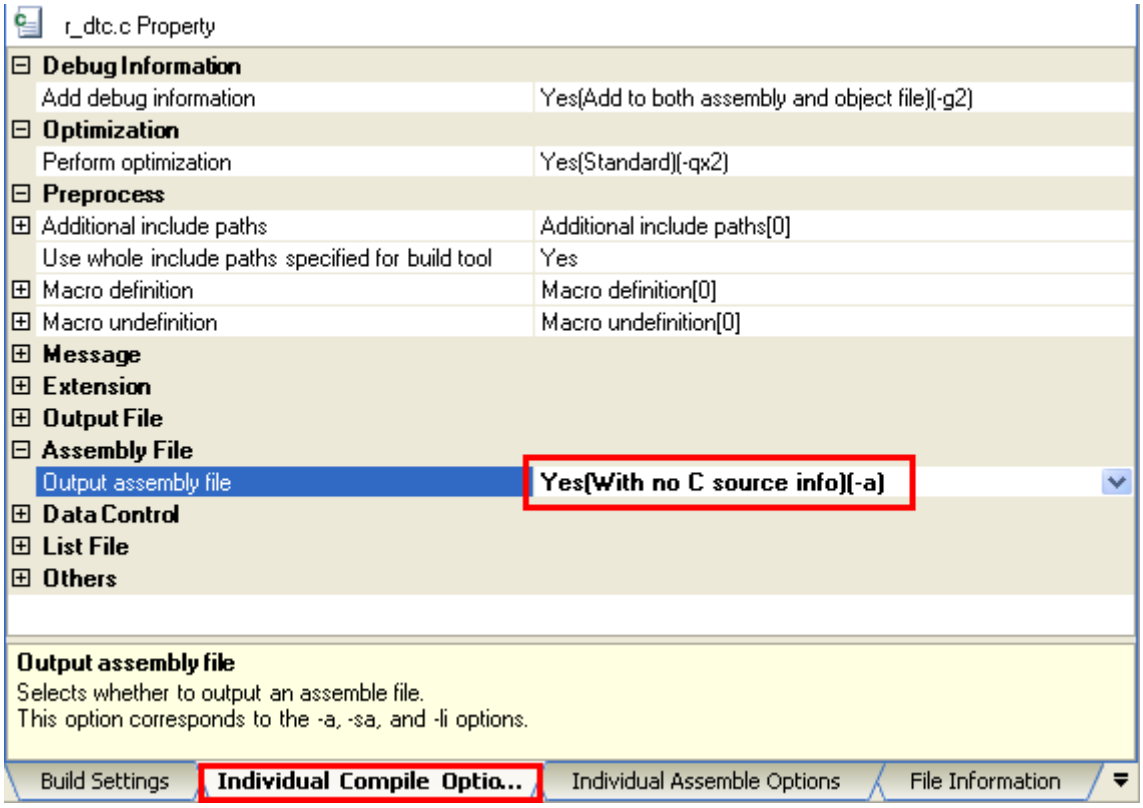
When a clock sauce is set to 15(fIL)kHz on device, clock function cannot be used. However, it is displayed on GUI that clock function seems to be used with 15(fIL)kHz. Please do not set up clock function.
[Workaround] RL78 Applilet3 is corrected

### 5.2.8 Cautions when using a DTC function

When DTC is used, please set up the following individual option of building. The DATA section is added for DTC to output sauce file "r_cg_dtc.c." Unless the individual option is set up, the following warning message is displayed and an object file is not generated.

( CC78K0R warning W0837: Output assembler source file , not object file )




[Workaround] There is no workaround

### 5.2.9 Cautions of initial function of an A/D converter

After making the port 2 a setup which does not compete with an A/D converter, the initialization function at the time of setting up an analog input terminal by an A/D converter has an error. Source code outputted by R_ADC_Create() "PM2 |= 0x??;" The value of 0x?? has an error.

[Workaround] RL78 Applilet3 is corrected

### 5.2.10 Cautions of initial function at the time of setting up UART transmission

The source code of a SDRmn register setup is not outputted to initialization function R_UARTn_Create() at the time of choosing only UART transmission.

[Workaround] R78 Applilet3 is corrected .

### 5.2.11 Cautions of Complementary assistant PWM mode of TimerRD

When TimerRD Complementary PWM mode is used using a high-speed system clock by clock setup of RL78/G14, it is necessary to change a setup of an option byte. RL78/G14 512 pages of R01UH0186JJ0100 Rev.1.00 edited by user's manual hardware Please refer to Notes 1.

[Workaround] RL78 Applilet3 is corrected

### 5.2.12 Restrictions of internal low-speed or internal high-speed oscillator trimming

Applilet3 is not equivalent to a setup of internal low-speed or internal high-speed oscillator trimming register

[Workaround] There is no workaround.

### 5.2.13 Cautions of Safety Functions

RAM parity error detection function of Safety Functions has not corresponded.

[Workaround] There is no workaround.

### 5.2.14 Cautions of file merge

If you select Merge File in Generate File Mode in the property of Applilet3 and the source codes are written between each comment below, the file will be merged.
/* Start user code. Do not edit comment generated here */
/* End user code. Do not edit comment generated here */
However, if the number of braces ("{" and "}") in the edited source codes
(including the comments) are not the same, the edited source codes may disappear
when you run the Applilet3.

[Workaround] There is no workaround.

# 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of 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. 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 should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended 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; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor 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 damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may 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 should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

**SALES OFFICES** Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.


**Renesas Electronics America Inc.**
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

**Renesas Electronics Canada Limited**
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

**Renesas Electronics Europe Limited**
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

**Renesas Electronics Europe GmbH**
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

**Renesas Electronics Hong Kong Limited**
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

**Renesas Electronics Taiwan Co., Ltd.**
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

**Renesas Electronics Malaysia Sdn.Bhd.**
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

**Renesas Electronics Korea Co., Ltd.**
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141


**CAUTION**

If the requirements shown in the "CAUTION" sentences are ignored, the equipment may cause personal injury or damage to the products.

Renesas Tools Website http://www.renesas.com/tools




Colophon 2.2