Abstract
In this application note, four examples of LIN basic communication are shown.

Products
MCUs: SH72A2 Group
     SH72A0 Group

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.
## Contents

1. Specifications ..................................................................................................................................... 3

2. Operation Confirmation Conditions .................................................................................................... 4

3. Reference Application Note................................................................................................................ 4

4. Hardware ............................................................................................................................................ 5
   4.1 Pins Used ..................................................................................................................................... 5
   4.2 Reference Connection ................................................................................................................. 5

5. Software ............................................................................................................................................. 6
   5.1 Operation Overview ..................................................................................................................... 6
      5.1.1 Initial settings ............................................................................................................................ 6
      5.1.2 Frame transmission (Non-frame separate mode) ................................................................. 7
      5.1.3 Frame reception ..................................................................................................................... 10
      5.1.4 Wake–up transmission ........................................................................................................... 12
      5.1.5 Wake–up reception ................................................................................................................ 14
   5.2 Required Memory Size .............................................................................................................. 15
   5.3 Invariable Table ......................................................................................................................... 15
   5.4 Function Table ........................................................................................................................... 15
   5.5 Function Specifications .............................................................................................................. 16
   5.6 Flowchart .................................................................................................................................... 19
      5.6.1 Main Processing ..................................................................................................................... 19
      5.6.2 System Clock Initialization Processing .................................................................................. 20
      5.6.3 LIN Initialization Processing ................................................................................................. 21
      5.6.4 Frame Transmission (Non-frame separate mode) Processing ............................................ 25
      5.6.5 Frame Reception Processing ............................................................................................... 26
      5.6.6 Wake-up Transmission Processing ....................................................................................... 27
      5.6.7 Wake-up Reception Processing ............................................................................................. 28
1. Specifications

This application note describes four LIN basic communications.
- Frame transmission (Non-frame separate mode)
- Frame reception
- Wake-up transmission
- Wake-up reception
2. Operation Confirmation Conditions
The sample code accompanying this application note has been run and confirmed under the conditions below.

Table 2.1 Operation Confirmation Conditions

<table>
<thead>
<tr>
<th>Item</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCU used</td>
<td>SH72A2 Group, SH72A0 Group</td>
</tr>
<tr>
<td>Operating frequency</td>
<td>External input clock 10MHz, CPU clock 100MHz, Bus clock 50MHz, LIN clock 25MHz</td>
</tr>
<tr>
<td>Operating voltage</td>
<td>5 V</td>
</tr>
</tbody>
</table>
| Integrated development environment | Renesas Electronics products  
                             | High-performance Embedded Workshop V.4.08.011                           |
| C compiler                  | Renesas Electronics products  
                             | SuperH RISC engine Standard Toolchain V.9.4.0.0                         |
| Operating mode              | Single-chip mode                                                        |

Table 2.2 shows the allocation of the sections used in this application.

<table>
<thead>
<tr>
<th>Section Name</th>
<th>Description</th>
<th>Area</th>
<th>Start Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>DVECTTBL</td>
<td>Reset vector table</td>
<td>ROM</td>
<td>0x00000000</td>
</tr>
<tr>
<td>DINNTBL</td>
<td>Interrupt vector table</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>PResetPRG</td>
<td>Reset programs</td>
<td>ROM</td>
<td>0x00000800</td>
</tr>
<tr>
<td>PIntPRG</td>
<td>Interrupt programs</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Program area</td>
<td>ROM</td>
<td>0x00001000</td>
</tr>
<tr>
<td>C</td>
<td>Constant area</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>C$BSEC</td>
<td>Structure for uninitialized data area addresses</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>C$DSEC</td>
<td>Structure for initialized data area addresses</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Initialized data(initial values)</td>
<td>ROM</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Uninitialized data area</td>
<td>RAM</td>
<td>0xFFF80000</td>
</tr>
<tr>
<td>R</td>
<td>Initialized data area</td>
<td>RAM</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>Stack area</td>
<td>RAM</td>
<td>0xFFF83800</td>
</tr>
</tbody>
</table>

3. Reference Application Note
None
4. Hardware

4.1 Pins Used

Table 4.1 lists the used pins and their functions.

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>I/O</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD05</td>
<td>Input</td>
<td>LIN reception port</td>
</tr>
<tr>
<td>PD04</td>
<td>Output</td>
<td>LIN transmission port</td>
</tr>
<tr>
<td>PK00</td>
<td>Output</td>
<td>Transceiver sleep control, HIGH level constantly</td>
</tr>
<tr>
<td>PK01</td>
<td>Output</td>
<td>Transceiver wake-up output control, HIGH level constantly</td>
</tr>
</tbody>
</table>

4.2 Reference Connection

Figure 4.1 shows a connection example in this application note.

![Figure 4.1 Connection Example](image-url)
5. Software

5.1 Operation Overview

5.1.1 Initial settings

There are 2 parts about initial settings.

One part is about system clock settings.

The other part is about LIN communication settings.

System clock settings are including:

- Set f(CPU) division, here set as No division
- Set f(BUS) division, here set as division by 2
- Set f(PBB)* division, here set as division by 4.
- Supply the LIN bus clock

* f(PBB) is as LIN bus clock.

LIN communication settings are including:

- Set LIN function pins and LIN transceiver control pins.
  Here, PD05 is as LRX2, PD04 is as LTX2.
  PK00 is used to control NSLP pin, PK01 is used to control NWAKE pin.
- Set LIN communication clock source.
  Here, f(SYS) divided by 4 is used as LIN communication clock source.
- Set LIN communication baud rate. Figure 5.1 shows baud rate generation block example.

![Figure 5.1 Baud rate Generation Block Example](image-url)
5.1.2 Frame transmission (Non-frame separate mode)

- Specification
  LIN master sends frame in non-frame separate mode.
  LIN slave can do process after it received the frame.
  Table 5.1 lists the communication settings

Table 5.1 Communication Settings Example (Frame Transmission)

<table>
<thead>
<tr>
<th>Item</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud rate</td>
<td>9600bps</td>
</tr>
<tr>
<td>Break width</td>
<td>13Tbits</td>
</tr>
<tr>
<td>Break delimiter width</td>
<td>1Tbit</td>
</tr>
<tr>
<td>Inter byte space (header) /Response space</td>
<td>0Tbit</td>
</tr>
<tr>
<td>Inter byte space (response)</td>
<td>0Tbit</td>
</tr>
<tr>
<td>Frame separate mode</td>
<td>Non-frame separate mode</td>
</tr>
<tr>
<td>Checksum type</td>
<td>Classic</td>
</tr>
<tr>
<td>Response field data length</td>
<td>8 data bytes</td>
</tr>
<tr>
<td>ID&amp;IDP</td>
<td>0xF5</td>
</tr>
<tr>
<td>Data[0] ~ Data[7]</td>
<td>0x00, 0x01, 0x02, 0x03, 0x10, 0x11, 0x12, 0x13</td>
</tr>
</tbody>
</table>
### Register being used description

**Table 5.2 Register being used**

<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
</table>
| LIN Baud Rate Prescaler 0 Register (LBRP0)    | H'FF61 0002 | H'50    | Set LIN baud rate prescaler  
LBRP0 = "H'50": The peripheral clock is divided by (80+1)                                                                                   |
| LIN2 Mode Register (L2MD)                     | H'FF61 0048 | H'04    | Select LIN system clock  
LCKS = "B'01": “fb” is selected                                                                                                             |
| LIN2 Break Field Setting Register (L2BRK)     | H'FF61 0049 | H'00    | Set Break delimiter transmission  
BDT = "B'00": 1 Tbit is set  
Set Break transmission  
BLT = "B'0000": 13 Tbits is set                                                                |
| LIN2 Space Setting Register (L2SPC)           | H'FF61 004A | H'00    | Set Interbyte space  
IBS = "B'00": 0 Tbit is set  
Set Interbyte space (Header)/Response space  
IBSH = "B'000": 0 Tbit is set                                                                     |
| LIN2 Interrupt Enable Register (L2IE)         | H'FF61 004C | H'00    | Set LIN interrupt enable bits  
ERRIE = "B'0": Disable Error detection interrupt  
FRCIE = "B'0": Disable Frame/Wake-up receive completion interrupt  
FTCIE = "B'0": Disable Frame/Wake-up transmit completion interrupt |
| LIN2 Error Detection Enable Register (L2EDE)  | H'FF61 004D | H'0F    | Set LIN Error detection enable bits  
FERE = "B'1": Enable Framing error detection  
FTERE = "B'1": Enable Frame timeout error detection  
PBERE = "B'1": Enable Physical bus error detection  
BERE = "B'1": Enable Bit error detection                                                                       |
| LIN2 Response Field Setting Register (L2RFC)  | H'FF61 0054 | H'18    | Set Frame separate mode  
FSM = "B'0": Non-frame separate mode is selected  
Set Check sum mode  
CSM = "B'0": Classic check sum is selected  
Set Response field Transmit/Receive direction  
RFT = "B'1": Transmission direction is set  
Set Response field data length  
RFDL = "B'1000": 8 bytes + check sum is set                                                                 |
| LIN2 ID Buffer Register (L2IDB)               | H'FF61 0055 | H'F5    | Set Parity bits  
IDP = "B'11"  
Set ID bits  
ID = "B'110101"                                                                 |
<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN2 Data 1 Buffer Register (L2DB1)</td>
<td>H'FF61 0058</td>
<td>H'00</td>
<td>Set data 1 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB1 = H'00</td>
</tr>
<tr>
<td>LIN2 Data 2 Buffer Register (L2DB2)</td>
<td>H'FF61 0059</td>
<td>H'01</td>
<td>Set data 2 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB2 = H'01</td>
</tr>
<tr>
<td>LIN2 Data 3 Buffer Register (L2DB3)</td>
<td>H'FF61 005A</td>
<td>H'02</td>
<td>Set data 3 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB3 = H'02</td>
</tr>
<tr>
<td>LIN2 Data 4 Buffer Register (L2DB4)</td>
<td>H'FF61 005B</td>
<td>H'03</td>
<td>Set data 4 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB4 = H'03</td>
</tr>
<tr>
<td>LIN2 Data 5 Buffer Register (L2DB5)</td>
<td>H'FF61 005C</td>
<td>H'10</td>
<td>Set data 5 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB5 = H'10</td>
</tr>
<tr>
<td>LIN2 Data 6 Buffer Register (L2DB6)</td>
<td>H'FF61 005D</td>
<td>H'11</td>
<td>Set data 6 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB6 = H'11</td>
</tr>
<tr>
<td>LIN2 Data 7 Buffer Register (L2DB7)</td>
<td>H'FF61 005E</td>
<td>H'12</td>
<td>Set data 7 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB7 = H'12</td>
</tr>
<tr>
<td>LIN2 Data 8 Buffer Register (L2DB8)</td>
<td>H'FF61 005F</td>
<td>H'13</td>
<td>Set data 8 to be transmitted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L2DB8 = H'13</td>
</tr>
</tbody>
</table>
5.1.3 Frame reception

- Specification
  LIN master sends header.
  LIN slave must return response with correct checksum, 8 response data field and classic checksum is required in this example.
  Table 5.3 lists the communication settings

Table 5.3 Communication Settings Example (Frame Reception)

<table>
<thead>
<tr>
<th>Item</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud rate</td>
<td>9600bps</td>
</tr>
<tr>
<td>Break width</td>
<td>13Tbits</td>
</tr>
<tr>
<td>Break delimiter width</td>
<td>1Tbit</td>
</tr>
<tr>
<td>Inter byte space (header)</td>
<td>0Tbit</td>
</tr>
<tr>
<td>Checksum type</td>
<td>Classic</td>
</tr>
<tr>
<td>Response field data length</td>
<td>8 data bytes</td>
</tr>
<tr>
<td>ID&amp;IDP</td>
<td>0x76</td>
</tr>
</tbody>
</table>
### Register being used description

**Table 5.4 Register being used**

<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN Baud Rate Prescaler 0 Register</td>
<td>H’FF61 0002</td>
<td>H’50</td>
<td>Set LIN baud rate prescaler</td>
</tr>
<tr>
<td>(LBRP0)</td>
<td></td>
<td></td>
<td>LBRP0 = “H’50”: The peripheral clock is divided by (80+1)</td>
</tr>
<tr>
<td>LIN2 Mode Register</td>
<td>H’FF61 0048</td>
<td>H’04</td>
<td>Select LIN system clock</td>
</tr>
<tr>
<td>(L2MD)</td>
<td></td>
<td></td>
<td>LCKS = “B’01”: “fb” is selected</td>
</tr>
<tr>
<td>LIN2 Break Field Setting Register</td>
<td>H’FF61 0049</td>
<td>H’00</td>
<td>Set Break delimiter transmission</td>
</tr>
<tr>
<td>(L2BRK)</td>
<td></td>
<td></td>
<td>BDT = “B’00”: 1 Tbit is set</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Break transmission</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BLT = “B’0000”: 13 Tbits is set</td>
</tr>
<tr>
<td>LIN2 Space Setting Register</td>
<td>H’FF61 004A</td>
<td>H’00</td>
<td>Set Interbyte space</td>
</tr>
<tr>
<td>(L2SPC)</td>
<td></td>
<td></td>
<td>IBS = “B’00”: 0 Tbit is set</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Interbyte space (Header)/Response space</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>IBSH = “B’0000”: 0 Tbit is set</td>
</tr>
<tr>
<td>LIN2 Interrupt Enable Register</td>
<td>H’FF61 004C</td>
<td>H’00</td>
<td>Set LIN interrupt enable bits</td>
</tr>
<tr>
<td>(L2IE)</td>
<td></td>
<td></td>
<td>ERRIE = “B’0”: Disable Error detection interrupt</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FRCIE = “B’0”: Disable Frame/Wake-up receive completion interrupt</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FTCIE = “B’0”: Disable Frame/Wake-up transmit completion interrupt</td>
</tr>
<tr>
<td>LIN2 Error Detection Enable Register</td>
<td>H’FF61 004D</td>
<td>H’0F</td>
<td>Set LIN Error detection enable bits</td>
</tr>
<tr>
<td>(L2EDE)</td>
<td></td>
<td></td>
<td>FERE = “B’1”: Enable Framing error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FTERE = “B’1”: Enable Frame timeout error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PBERE = “B’1”: Enable Physical bus error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BERE = “B’1”: Enable Bit error detection</td>
</tr>
<tr>
<td>LIN2 Response Field Setting Register</td>
<td>H’FF61 0054</td>
<td>H’08</td>
<td>Set Frame separate mode</td>
</tr>
<tr>
<td>(L2RFC)</td>
<td></td>
<td></td>
<td>FSM = “B’0”: Non-frame separate mode is selected</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Check sum mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CSM = “B’0”: Classic check sum is selected</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Response field Transmit/Receive direction</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RFT = “B’0”: Reception direction is set</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Response field data length</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RFDL = “B’1000”: 8 bytes + check sum is set</td>
</tr>
<tr>
<td>LIN2 ID Buffer Register</td>
<td>H’FF61 0055</td>
<td>H’76</td>
<td>Set Parity bits</td>
</tr>
<tr>
<td>(L2IDB)</td>
<td></td>
<td></td>
<td>IDP = “B’01”</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set ID bits</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ID = “B’110110”</td>
</tr>
</tbody>
</table>
5.1.4 Wake-up transmission

- Specification
  LIN master sends Wake-up signal with specified width.
  Table 5.5 lists the communication settings

Table 5.5 Communication Settings Example (Wake-up Transmission)

<table>
<thead>
<tr>
<th>Item</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud rate</td>
<td>9600bps</td>
</tr>
<tr>
<td>Wake-up signal width</td>
<td>3Tbits</td>
</tr>
</tbody>
</table>
### Register being used description

#### Table 5.6 Register being used

<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN Baud Rate Prescaler 0 Register (LBRP0)</td>
<td>H'FF61 0002</td>
<td>H'50</td>
<td>Set LIN baud rate prescaler</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LBRP0 = “H’50”: The peripheral clock is divided by (80+1)</td>
</tr>
<tr>
<td>LIN2 Mode Register (L2MD)</td>
<td>H'FF61 0048</td>
<td>H'04</td>
<td>Select LIN system clock</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LCKS = “B’01”: “fb” is selected</td>
</tr>
<tr>
<td>LIN2 Wake-up Setting Register</td>
<td>H'FF61 004B</td>
<td>H'20</td>
<td>Set Wake-up transmission low time pulse width</td>
</tr>
<tr>
<td>(L2WUP)</td>
<td></td>
<td></td>
<td>WUTL = “B’0010”: 3 Tbits</td>
</tr>
<tr>
<td>LIN2 Interrupt Enable Register</td>
<td>H'FF61 004C</td>
<td>H'00</td>
<td>Set LIN interrupt enable bits</td>
</tr>
<tr>
<td>(L2IE)</td>
<td></td>
<td></td>
<td>ERRIE = “B’0”: Disable Error detection interrupt</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FRCIE = “B’0”: Disable Frame/Wake-up receive completion interrupt</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FTCIE = “B’0”: Disable Frame/Wake-up transmit completion interrupt</td>
</tr>
<tr>
<td>LIN2 Error Detection Enable</td>
<td>H'FF61 004D</td>
<td>H'0F</td>
<td>Set LIN Error detection enable bits</td>
</tr>
<tr>
<td>Register (L2EDE)</td>
<td></td>
<td></td>
<td>FERE = “B’1”: Enable Framing error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FTERE = “B’1”: Enable Frame timeout error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PBERE = “B’1”: Enable Physical bus error detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BERE = “B’1”: Enable Bit error detection</td>
</tr>
<tr>
<td>LIN2 Response Field Setting</td>
<td>H'FF61 0054</td>
<td>H'10</td>
<td>Set Frame separate mode</td>
</tr>
<tr>
<td>Register (L2RFC)</td>
<td></td>
<td></td>
<td>FSM = “B’0”: Non-frame separate mode is selected(*) not valid here</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Check sum mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CSM = “B’0”: Classic check sum is selected (* not valid here)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Response field Transmit/Receive direction</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RFT = “B’1”: Transmission direction is set</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set Response field data length</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RFDL = “B’0000”: 8 bytes + check sum is set (* not valid here)</td>
</tr>
</tbody>
</table>
5.1.5 Wake-up reception

- Specification
  LIN master waits Wake-up signal input with specified width.
  LIN slave must send signal low time to be 125μs or more in this example.

- Register being used description
Table 5.7 Register being used

<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN Wake-up Baud Rate Select Register (LWBR)</td>
<td>H'FF61 0001</td>
<td>H'01</td>
<td>Set Wake-up baud rate&lt;br&gt;  LWBR0 = &quot;B'1&quot;:&lt;br&gt;    When LIN 2.0 and 2.1 used.&lt;br&gt;    This allows the input signal Low time to be measured 125 μs or more.</td>
</tr>
<tr>
<td>LIN Baud Rate Prescaler 0 Register (LBRP0)</td>
<td>H'FF61 0002</td>
<td>H'50</td>
<td>Set LIN baud rate prescaler&lt;br&gt;    LBRP0 = &quot;H'50&quot;: The peripheral clock is divided by (80+1)</td>
</tr>
<tr>
<td>LIN2 Interrupt Enable Register (L2IE)</td>
<td>H'FF61 004C</td>
<td>H'00</td>
<td>Set LIN interrupt enable bits&lt;br&gt;     ERRIE = &quot;B'0&quot;: Disable Error detection interrupt&lt;br&gt;     FRCIE = &quot;B'0&quot;: Disable Frame/Wake-up receive completion interrupt&lt;br&gt;     FTCIE = &quot;B'0&quot;: Disable Frame/Wake-up transmit completion interrupt</td>
</tr>
<tr>
<td>LIN2 Error Detection Enable Register (L2EDE)</td>
<td>H'FF61 004D</td>
<td>H'0F</td>
<td>Set LIN Error detection enable bits&lt;br&gt;     FERE = &quot;B'1&quot;: Enable Framing error detection&lt;br&gt;     FTERE = &quot;B'1&quot;: Enable Frame timeout error detection&lt;br&gt;     PBERE = &quot;B'1&quot;: Enable Physical bus error detection&lt;br&gt;     BERE = &quot;B'1&quot;: Enable Bit error detection</td>
</tr>
<tr>
<td>LIN2 Response Field Setting Register (L2RFC)</td>
<td>H'FF61 0054</td>
<td>H'00</td>
<td>Set Frame separate mode&lt;br&gt;     FSM = &quot;B'0&quot;: Non-frame separate mode is selected(* not valid here)&lt;br&gt;     Set Check sum mode&lt;br&gt;     CSM = &quot;B'0&quot;: Classic check sum is selected (* not valid here)&lt;br&gt;     Set Response field Transmit/Receive direction&lt;br&gt;     RFT = &quot;B'0&quot;: Reception direction is set&lt;br&gt;     Set Response field data length&lt;br&gt;     RFDL = &quot;B'0000&quot;: 8 bytes + check sum is set (* not valid here)</td>
</tr>
</tbody>
</table>
5.2 Required Memory Size

Table 5.8 lists the required memory size.

<table>
<thead>
<tr>
<th>Memory Used</th>
<th>Size</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROM</td>
<td>2449 Bytes</td>
<td>SH RISC Compiler 9.4.0.0, Optimization OFF</td>
</tr>
<tr>
<td>RAM</td>
<td>4 Bytes</td>
<td></td>
</tr>
<tr>
<td>Maximum user stack</td>
<td>1024 Bytes</td>
<td></td>
</tr>
</tbody>
</table>

The required memory size varies depending on the C compiler version and compiler options.

5.3 Invariable Table

Table 5.9 lists the invariables used in the sample code.

<table>
<thead>
<tr>
<th>Invariable Name</th>
<th>Setting Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FALSE</td>
<td>0</td>
<td>False</td>
</tr>
<tr>
<td>TRUE</td>
<td>1</td>
<td>True</td>
</tr>
<tr>
<td>CUR_LIN_CH</td>
<td>0x02</td>
<td>Current LIN channel number</td>
</tr>
</tbody>
</table>

5.4 Function Table

Table 5.10 lists the functions.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>initSysClock</td>
<td>Initialize system clock</td>
</tr>
<tr>
<td>LINInit</td>
<td>Initialize LIN communication source</td>
</tr>
<tr>
<td>LINSetClockSource</td>
<td>Set LIN communication clock source</td>
</tr>
<tr>
<td>LINSetPins</td>
<td>Set LIN pins used</td>
</tr>
<tr>
<td>LINResetMode</td>
<td>Set specified LIN channel to Reset mode</td>
</tr>
<tr>
<td>LINOperateMode</td>
<td>Set specified LIN channel to Operate mode</td>
</tr>
<tr>
<td>LINWakeupMode</td>
<td>Set specified LIN channel to Wake-up mode</td>
</tr>
<tr>
<td>LINSetBaudrate</td>
<td>Set LIN communication baud rate</td>
</tr>
<tr>
<td>LINFrameTxNotSeparate</td>
<td>Frame transmission (not-frame separate mode)</td>
</tr>
<tr>
<td>LINFrameRx</td>
<td>Frame reception</td>
</tr>
<tr>
<td>LINWupTx</td>
<td>Wake-up transmission</td>
</tr>
<tr>
<td>LINWupRx</td>
<td>Wake-up reception</td>
</tr>
<tr>
<td>main</td>
<td>main function</td>
</tr>
</tbody>
</table>
### 5.5 Function Specifications

The following tables list the sample code function specifications.

<table>
<thead>
<tr>
<th>Function</th>
<th>Outline</th>
<th>Header</th>
<th>Declaration</th>
<th>Description</th>
<th>Argument</th>
<th>Returned value</th>
</tr>
</thead>
<tbody>
<tr>
<td>initSysClock</td>
<td>Initialize system clock</td>
<td>None</td>
<td>static void initSysClock(void)</td>
<td>Initialize system clock as 100MHz</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>LINInit</td>
<td>Initialization for LIN communication</td>
<td>None</td>
<td>static void LINInit (void)</td>
<td>Initialize LIN communication clock source, pins, baud rate</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>LINSetClockSource</td>
<td>Set LIN communication clock source</td>
<td>None</td>
<td>static void LINSetClockSource(void)</td>
<td>Set LIN communication clock source f(LIN) = f(SYS)/4 = 100 MHz / 4 = 25 MHz,</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>LINSetPins</td>
<td>Set LIN pins used</td>
<td>None</td>
<td>static void LINSetPins(void)</td>
<td>PD05 is as for LRX2 PD04 is as for LTX2</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
### LINResetMode

- **Outline**: Set specified LIN channel to Reset mode
- **Header**: None
- **Declaration**: `static void LINResetMode(U8 LINchannel)`
- **Description**: Set specified LIN channel to Reset mode
- **Argument**: LINchannel, specified LIN channel
- **Returned value**: None

### LINOperateMode

- **Outline**: Set specified LIN channel to Operate mode
- **Header**: None
- **Declaration**: `static void LINOperateMode(U8 LINchannel)`
- **Description**: Set specified LIN channel to Operate mode
- **Argument**: LINchannel, specified LIN channel
- **Returned value**: None

### LINWakeupMode

- **Outline**: Set specified LIN channel to Wake-up mode
- **Header**: None
- **Declaration**: `static void LINWakeupMode(U8 LINchannel)`
- **Description**: Set specified LIN channel to Wake-up mode
- **Argument**: LINchannel, specified LIN channel
- **Returned value**: None

### LINSetBaudrate

- **Outline**: Set LIN communication baud rate
- **Header**: None
- **Declaration**: `static void LINSetBaudrate(U8 LINchannel)`
- **Description**: Set LIN communication baud rate as 9600bps
- **Argument**: LINchannel, specified LIN channel
- **Returned value**: None

### LINFrameTxNotSeparate

- **Outline**: Frame transmission (not-frame separate mode)
- **Header**: None
- **Declaration**: `static void LINFrameTxNotSeparate(void)`
- **Description**: LIN frame transmission (not-frame separate mode)
- **Argument**: None
- **Returned value**: None
### LINFrameRx

<table>
<thead>
<tr>
<th>Outline</th>
<th>Frame reception</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>static void LINFrameRx(void)</td>
</tr>
<tr>
<td>Description</td>
<td>LIN frame reception</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
</tbody>
</table>

### LINWupTx

<table>
<thead>
<tr>
<th>Outline</th>
<th>Wake-up transmission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>static void LINWupTx(void)</td>
</tr>
<tr>
<td>Description</td>
<td>LIN Wake-up transmission</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
</tbody>
</table>

### LINWupRx

<table>
<thead>
<tr>
<th>Outline</th>
<th>Wake-up reception</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>static void LINWupRx(void)</td>
</tr>
<tr>
<td>Description</td>
<td>LIN Wake-up reception</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
</tbody>
</table>

### main

<table>
<thead>
<tr>
<th>Outline</th>
<th>main function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>void main(void)</td>
</tr>
<tr>
<td>Description</td>
<td>main function</td>
</tr>
<tr>
<td></td>
<td>call initSysClock()</td>
</tr>
<tr>
<td></td>
<td>call LINInit()</td>
</tr>
<tr>
<td></td>
<td>call LINFrameTxNotSeparate()</td>
</tr>
<tr>
<td></td>
<td>call LINFrameRx()</td>
</tr>
<tr>
<td></td>
<td>call LINWupTx()</td>
</tr>
<tr>
<td></td>
<td>call LINWupRx()</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
</tbody>
</table>
5.6 Flowchart

5.6.1 Main Processing

Figure 5.2 shows the main processing.

* LIN communication process is selected based on different compile macro definition.

Figure 5.2 Main Processing
5.6.2 System Clock Initialization Processing

Figure 5.3 shows the system clock initialization processing.

Start

Protect off
- System protect
- Clock protect

Set division
- CPU clock
- Bus clock
- LIN bus clock

Supply LIN bus clock

Protect on
- System protect
- Clock protect

End

- System Protection unlocked
  System Protect Register 0 (SPR0) = H'F1
- Clock Protection unlocked
  Clock Protect Register (CPR) = H'F1

- Set CPU Clock Divide Register (CCDR)
  CPUD = “B’000”: CPU Clock Division by 1
  BUSD = “B’001”: Bus Clock Division by 2

- Set Peripheral Bus Clock Divide Register (PBCDR)
  PBBD = “B’010”: Peripheral Bus Clock B Division by 4

- Set Peripheral Bus Clock Control Register (PBCCR)
  STPLIN = “B’0”: LIN bus clock supplied

- Clock Protection locked
  Clock Protect Register (CPR) = H'F0
- System Protection locked
  System Protect Register 0 (SPR0) = H'F0

Figure 5.3 System Clock Initialization Processing
5.6.3 LIN Initialization Processing

Figure 5.4 shows the LIN initialization processing.
Figure 5.5 shows the LIN communication pins setting processing.
Figure 5.6 shows the LIN communication clock source setting processing.
Figure 5.7 shows the LIN communication baud rate setting processing.

![Diagram](image-url)

**Figure 5.4 LIN Communication Initialization Processing**
Figure 5.5 LIN Communication Pins Setting Processing

- Set Port Protect Register (PPR)
  PPR = "H'AA": Write enabled

- Set Port K Direction Register (PKDR)
  PKD00 = "B'1": output direction
  PKD01 = "B'1": output direction

- Set Port K0 Function Select Register (PK0S)
  PK0S = "B'00": IO port

- Set Port K1 Function Select Register (PK1S)
  PK1S = "B'00": IO port

- Set Port K Register (PKR)
  PK00 = "B'1": output High
  PK01 = "B'1": output High

- Set Port D Direction Register (PDDDR)
  PDD05 = "B'0": input direction
  PDD04 = "B'1": output direction

- Set Pull-up Control Register 0 (PUR0)
  PUR00 = "B'1": PD00 to PD07 pulled up

- Set Input Threshold Value Select Register 0 (PVSS0)
  PVSS0 = "B'10": PD00 to PD07 Input Threshold value is 0.70VCC

- Set Port Function Select Register 3 (PFS3)
  LN2S = "B'0": LRX2 input is set to the PD05 pin

- Set Port D04 Function Select Register (PD04S)
  PD04S = "B'0010": LIN function

- Set Port D05 Function Select Register (PD05S)
  PD05S = "B'0010": LIN function

- Set Port Function Enable Register 0 (PFENO)
  PDFEN = "B'1": PD00 to PD15 functions selected by bits PD00S to PD15S are enabled

- Set Port Protect Register (PPR)
  PPR = "H'55": Write disabled
Figure 5.6 LIN Communication Clock Source Setting Processing
- Set to LIN Reset mode
  Set LIN2 Control Register (L2C)
  OM0 = "B'0": LIN reset mode
  and, check until LIN2 Mode Status Register (L2MST)’s
  OMM0 == "B'0" (LIN2 is reset mode)

- Set LIN2 Mode Register (L2MD)
  LCKS = "B'01": select "fb" as LIN System Clock

- Set LIN Baud Rate Prescaler 0 Register (LBRP0)
  LBRP0 = "H'50": The peripheral clock is divided
  by (80+1)

Figure 5.7 LIN Communication Baud rate Setting Processing
5.6.4 Frame Transmission (Non-frame separate mode) Processing

Figure 5.8 shows Frame transmission (Non–frame separate mode) Processing.

- Set to LIN Reset mode
  Set LIN2 Control Register (L2C) OMO = “B’0”: LIN reset mode and, check until LIN2 Mode Status Register (L2MST)’s OMMO = “B’0” (LIN2 is reset mode)

- Set LIN2 Break Field Setting Register (L2BFRK) BLT = “B’0000”: 13 Tbits Break Transmission BDT = “B’00”: 1 Tbit Break Delimiter Transmission

- Set LIN2 Space Setting Register (L2SPC) IBSH = “B’000”: 0 Tbit Interbyte Space (Header) IBS = “B’00”: 0 Tbit Interbyte Space

- Set LIN2 Response Field Setting Register (L2RFC) RFDL = “B’1000”: 8 bytes + checksum RFT = “B’1”: Transmission CSM = “B’0”: Classic checksum FSM = “B’0”: Non-frame separate mode

- Set LIN2 ID Buffer Register (LIN2IDB) IDP = “B’11” ID = “B’110101”


- Set to LIN Operation mode Set LIN2 Control Register (L2C) OMO = “B’1” and, check until LIN2 Mode Status Register (L2MST)’s OMMO = “B’1”

- Set LIN2 Transmission Control Register (L2TC) FT3 = “B’1”: Start frame transmission

- Check LIN2 Status Register (L2ST) until FTC = “B’1” (Frame transmission has been completed)

- Clear Frame Transmit Complete Flag LIN2 Status Register (L2ST) FTC = “B’0”
5.6.5 Frame Reception Processing

Figure 5.9 shows the Frame reception Processing.

- Set to LIN Reset mode
  - Set LIN2 Control Register (L2C)
    - OM0 = "B'0": LIN reset mode
    - and, check until LIN2 Mode Status Register (L2MST)’s
      - OMM0 == "B'0" (LIN2 is reset mode)
- Set LIN2 Break Field Setting Register (L2BRK)
  - BLT = "B'0000": 13 Tbits Break Transmission
  - BDT = "B'00": 1 Tbit Break Delimiter Transmission
- Set LIN2 Space Setting Register (L2SPC)
  - IBSH = "B'000": 0 Tbit Interbyte Space (Header)
  - /Response Space
- Set LIN2 Reponse Field Setting Register (L2RFC)
  - RFDL = "B'1000": 8 bytes + checksum
  - RFT = "B'0": Reception
  - CSM = "B'0": Classic check sum
- Set LIN2 ID Buffer Register (LIN2IDB)
  - IDP = "B'01"
  - ID = "B'1101110"
- Set LIN2 Error Detection Enable Register (L2EDE)
  - FERE = "B'1": Enable Framing Error Detection
  -FTERE = "B'1": Enable Frame Timeout Error Detection
  - PBERE = "B'1": Enable Physical Bus Error Detection
  - BERE = "B'1": Enable Bit Error Detection
- Set to LIN Operation mode
  - Set LIN2 Control Register (L2C)
    - OM0 = "B'1"
    - and, check until LIN2 Mode Status Register (L2MST)’s
      - OMM0 == "B'1"
    - OM1 = "B'1"
    - and, check until LIN2 Mode Status Register (L2MST)’s
      - OMM1 == "B'1"
- Set LIN2 Transmission Control Register (L2TC)
  - FTS = "B'1": Start frame reception
- Check LIN2 Status Register (L2ST)
  - until FRC == "B'1" (Frame Reception has been completed)
- Clear Frame Receive Complete Flag
  - LIN2 Status Register (L2ST)
  - FRC = "B'0"

* LIN slave should check ID & IDP from master and returns response, 8 data bytes and classic checksum is required in this example.

Figure 5.9 Frame Reception Processing
5.6.6 Wake-up Transmission Processing

Figure 5.10 shows the Wake-up transmission Processing.

- Set to LIN Reset mode
  Set LIN2 Control Register (L2C)
  \( \text{OM0} = \text{"B'}0\text{"}, \text{LIN reset mode} \)
  and, check until LIN2 Mode Status Register (L2MST)'s
  \( \text{OMM0} = \text{"B'}0\text{"} \) (LIN2 is reset mode)

- Set LIN2 Reponse Field Setting Register (L2RFC)
  \( \text{RFT} = \text{"B'}1\text{"}, \text{Transmission} \)

- Set LIN2 Wake-up Setting Register (L2WUP)
  \( \text{WUTL} = \text{"B'}0010\text{"}, \text{3 Tbits Wake-up Transmission Low Time Pulse} \)

- Set LIN2 Error Detection Enable Register (L2EDE)
  \( \text{FERE = "B'}1\text{"}, \text{Enable Framing Error Detection} \)
  \( \text{FTERE = "B'}1\text{"}, \text{Enable Frame Timeout Error Detection} \)
  \( \text{PBERE = "B'}1\text{"}, \text{Enable Physical Bus Error Detection} \)
  \( \text{BERE = "B'}1\text{"}, \text{Enable Bit Error Detection} \)

- Set to LIN Wake-up mode
  Set LIN2 Control Register (L2C)
  \( \text{OM0} = \text{"B'}1\text{"} \)
  and, check until LIN2 Mode Status Register (L2MST)'s
  \( \text{OMM0} = \text{"B'}1\text{"} \)
  \( \text{OM1} = \text{"B'}0\text{"} \)
  and, check until LIN2 Mode Status Register (L2MST)'s
  \( \text{OMM1} = \text{"B'}0\text{"} \)

- Set LIN2 Transmission Control Register (L2TC)
  \( \text{FTS = "B'}1\text{"}, \text{Start Wake-up transmission} \)

- Check LIN2 Status Register (L2ST)
  until \( \text{FTC} = \text{"B'}1\text{"} \) (Wake-up Transmission has been completed)

- Clear Wake-up Transmit Complete Flag
  LIN2 Status Register (L2ST)
  \( \text{FTC = "B'}0\text{"} \)

Figure 5.10 Wake-up Transmission Processing
5.6.7 Wake-up Reception Processing

Figure 5.11 shows the Wake-up reception Processing.

- Set to LIN Reset mode
  - Set LIN2 Control Register (L2C)
  - OM0 = "B'0": LIN reset mode
  - and, check until LIN2 Mode Status Register (L2MST)'s
    - OMM0 == "B'0" (LIN2 is reset mode)

- Set LIN Wake-up Baud Rate Select Register (LWBR)
  - LWBR0 = "B'1": When LIN 2.0 and 2.1 used.
  - This allows the input signal Low time to be measured
    - 125us or more.

- Set LIN2 Reponse Field Setting Register (L2RFC)
  - RFT = "B'0": Reception

- Set LIN2 Error Detection Enable Register (L2EDE)
  - FERE = "B'1": Enable Framing Error Detection
  - FTERE = "B'1": Enable Frame Timeout Error Detection
  - PBERE = "B'1": Enable Physical Bus Error Detection
  - BERE = "B'1": Enable Bit Error Detection

- Set to LIN Wake-up mode
  - Set LIN2 Control Register (L2C)
  - OM0 = "B'1"
  - and, check until LIN2 Mode Status Register (L2MST)'s
    - OMM0 == "B'1"
    - OM1 = "B'0"
  - and, check until LIN2 Mode Status Register (L2MST)'s
    - OMM1 == "B'0"

- Set LIN2 Transmission Control Register (L2TC)
  - FTS = "B'1": Start Wake-up reception

- Check LIN2 Status Register (L2ST)
  - until FRC == "B'1" (Wake-up Reception has been completed)

- Clear Wake-up Receive Complete Flag
  - LIN2 Status Register (L2ST)
  - FRC = "B'0"

* LIN slave should send specified width wake-up signal. 125 μs or more is required in this example.

Figure 5.11 Wake-up Reception Processing
<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Page</th>
<th>Description</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>May. 15, 2012</td>
<td>—</td>
<td>First edition issued</td>
<td></td>
</tr>
</tbody>
</table>

All trademarks and registered trademarks are the property of their respective owners.
General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on
   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.
Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products described herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.

2. 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.

3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.

4. 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.

5. When exporting the 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. You should not use Renesas Electronics products or the 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. 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.

6. 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.

7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. 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 categorized as “Specific” without the prior written consent of Renesas Electronics. 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 an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics.

8. 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.

9. 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 malfunction under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard 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 system 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. 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.

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.

(Notice 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-held subsidiaries.

(Notice 2) “Renesas Electronics products” means any product developed or manufactured by or for Renesas Electronics.