Introduction

This application note describes a method about how to use the key matrix input and the 4-digit 8-segment LED display with the R7F0C809 microcontroller.

Target Device

R7F0C809

When applying the sample program covered in this application note to another microcomputer with the same SFR (Special Function Register), modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.
Contents

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

2. Operating Conditions ..................................................................................................... 4

3. Related Application Note ............................................................................................... 4

4. Description of the Hardware .......................................................................................... 5
   4.1 Hardware Configuration Example ............................................................................... 5
   4.2 List of Pins to be Used ............................................................................................... 6
   4.3 High Current I/O Port Function .................................................................................. 6

5. Description of Software .................................................................................................. 7
   5.1 Operation Overview .................................................................................................... 7
   5.2 4-Digit 8-Segment LED Display controlling ............................................................. 8
   5.3 Key Scan controlling .................................................................................................. 8
   5.4 Timing Diagram ......................................................................................................... 9
   5.5 List of Option Byte Settings ....................................................................................... 10
   5.6 List of Constants ........................................................................................................ 10
   5.7 List of Variables ........................................................................................................ 11
   5.8 List of Functions ........................................................................................................ 12
   5.9 Function Specifications ............................................................................................. 12
   5.10 Flowcharts ............................................................................................................... 14
      5.10.1 System Initialization Function ............................................................................. 14
      5.10.2 I/O Port Setup .................................................................................................... 15
      5.10.3 Timer Array Unit Setup ...................................................................................... 19
      5.10.4 Main Processing ................................................................................................ 26
      5.10.5 LED Display Processing .................................................................................... 27
      5.10.6 Key Scan Processing ........................................................................................ 28
      5.10.7 LED Mode Processing ....................................................................................... 29

6. Sample Code ................................................................................................................ 30

7. Reference Documents .................................................................................................. 30
1. Specifications

This application note describes how the high current I/O ports of R7F0C809 MCU control the LED COM pins and are used as key scan output pins. The channel 0 of Timer Array Unit 0 (TAU0) is used to generate a time interval to switch the COM pins and the channel 1 of Timer Array Unit 0 is used to generate a time delay to wait for the stability of the COM port voltage when it is changed from high level to low level. P00 and P16 pins are used as key scan input ports.

Table 1.1 lists the peripheral functions and their applications.

<table>
<thead>
<tr>
<th>Peripheral Function</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAU0 channel 0</td>
<td>Generate a time interval to switch the COM pins (Control the time interval of LED display scanning).</td>
</tr>
<tr>
<td>TAU0 channel 1</td>
<td>Generate a time delay to wait for the stability of COM voltage.</td>
</tr>
<tr>
<td>P02 to P05</td>
<td>Control the LED COM ports and be used as key scan output ports.</td>
</tr>
<tr>
<td>P06, P07, P10 to P15</td>
<td>Control the LED SEG ports.</td>
</tr>
<tr>
<td>P00, P16</td>
<td>Key scan input ports.</td>
</tr>
</tbody>
</table>
2. Operating Conditions

The sample code contained in this application note has been tested 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>R7F0C809</td>
</tr>
<tr>
<td>Operating frequency</td>
<td>• High-speed on-chip oscillator clock (f_{HOCO}): 20 MHz (typ.)</td>
</tr>
<tr>
<td></td>
<td>• CPU/peripheral hardware clock (f_{CLK}): 20 MHz</td>
</tr>
<tr>
<td>Operating voltage</td>
<td>5.0 V (operation enabled from 4.5 to 5.5 V)</td>
</tr>
<tr>
<td></td>
<td>SPOR detection operation (V_{SPOR}): rising edge 4.28V (typ.), falling edge 4.00V (min.)</td>
</tr>
<tr>
<td>Integrated development environment</td>
<td>Renesas Electronics Corporation</td>
</tr>
<tr>
<td></td>
<td>CubeSuite+ V2.01.00</td>
</tr>
<tr>
<td>C compiler</td>
<td>Renesas Electronics Corporation</td>
</tr>
<tr>
<td></td>
<td>CA78K0R V1.60</td>
</tr>
</tbody>
</table>

3. Related Application Note

The application notes that are related to this application note are listed below for reference.

- R7F0C809 6-Digit 8-Segment LED Display (R01AN2005E) Application Note
- R7F0C809 4-Digit 8-Segment LED Display with A/D Key Read (R01AN2007E) Application Note
4. Description of the Hardware

4.1 Hardware Configuration Example

Figure 4.1 shows an example of hardware configuration that is used for this application note.

Notes:
1. The purpose of this circuit is only to provide the connection outline and the circuit is simplified accordingly. When designing and implementing an actual circuit, provide proper pin treatment and make sure that the hardware's electrical specifications are met (connect the input-dedicated ports separately to VDD or VSS via a resistor).
2. VDD must be held at not lower than the reset release voltage (VSPOR) that is specified as SPOR.
Table 4.1  Keys to be Used and Their Functions

<table>
<thead>
<tr>
<th>Key Name</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>K1</td>
<td>LED all blink mode</td>
<td>After the key pressed, the LEDs will start blinking. The interval time between LEDs on and LEDs off is constant.</td>
</tr>
<tr>
<td>K2</td>
<td>LED ring shift right mode</td>
<td>After the key pressed, the display data rotate right at the same interval time.</td>
</tr>
<tr>
<td>K3</td>
<td>LED ring shift left mode</td>
<td>After the key pressed, the display data rotate left at the same interval time.</td>
</tr>
<tr>
<td>K4</td>
<td>Reserved</td>
<td>None.</td>
</tr>
<tr>
<td>K5</td>
<td>Update display data</td>
<td>After the key pressed, the display data will be updated.</td>
</tr>
<tr>
<td>K6</td>
<td>Reset display data</td>
<td>After the key pressed, the LEDs will display the initial data.</td>
</tr>
<tr>
<td>K7</td>
<td>Reserved</td>
<td>None.</td>
</tr>
<tr>
<td>K8</td>
<td>Reserved</td>
<td>None.</td>
</tr>
</tbody>
</table>

4.2  List of Pins to be Used

Table 4.2  Pins to be Used and Their Function

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>I/O</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P02</td>
<td>Output</td>
<td>COM output 1 and key scan output.</td>
</tr>
<tr>
<td>P03</td>
<td>Output</td>
<td>COM output 2 and key scan output.</td>
</tr>
<tr>
<td>P04</td>
<td>Output</td>
<td>COM output 3 and key scan output.</td>
</tr>
<tr>
<td>P05</td>
<td>Output</td>
<td>COM output 4 and key scan output.</td>
</tr>
<tr>
<td>P06</td>
<td>Output</td>
<td>SEG output dp.</td>
</tr>
<tr>
<td>P07</td>
<td>Output</td>
<td>SEG output g.</td>
</tr>
<tr>
<td>P10</td>
<td>Output</td>
<td>SEG output f.</td>
</tr>
<tr>
<td>P11</td>
<td>Output</td>
<td>SEG output e.</td>
</tr>
<tr>
<td>P12</td>
<td>Output</td>
<td>SEG output d.</td>
</tr>
<tr>
<td>P13</td>
<td>Output</td>
<td>SEG output c.</td>
</tr>
<tr>
<td>P14</td>
<td>Output</td>
<td>SEG output b.</td>
</tr>
<tr>
<td>P15</td>
<td>Output</td>
<td>SEG output a.</td>
</tr>
<tr>
<td>P00</td>
<td>Input</td>
<td>Key scan input.</td>
</tr>
<tr>
<td>P16</td>
<td>Input</td>
<td>Key scan input.</td>
</tr>
</tbody>
</table>

4.3  High Current I/O Port Function

R7F0C809 microcontroller has 6 ports (P-ch open-drain output) to control the LED digits (COM pins), and 8 ports (N-ch open-drain) to control the LED segments (SEG pins). The highest output current of the pin which connects to the COM pin can reach 120mA when this pin is in the output mode of P-ch open-drain, and the highest output current of the pin which connects to SEG pin can reach 15mA in the output mode of N-ch open-drain. Additionally, please pay attention to that the number of pins which connect to COM pins that output "1" at the same time is not more than 1.
5. Description of Software

5.1 Operation Overview

This application note introduces how to use P-ch/N-ch open-drain ports (high current ports) of R7F0C809 MCU to control the LED display and using the external keypad to set the display mode.

(1) Initialize port.
- Set P00,P16 to digital input mode(key scan input ports).
- Set P02 to P05 to P-ch open-drain output mode (control the LED COM ports).
- Set P06,P07,P10 to P15 to N-ch open-drain output mode (control the LED SEG ports).

(2) Initialize the TAU.
- Set the count clock of TAU0 channel 0 and channel 1 to $f_{MCLK} = f_{CLK}/16=1.25MHz$ ($f_{CLK}=20MHz$).
- Set the operation mode of TAU0 channel 0 and channel 1 to interval timer mode.
- Set the value of timer data register 00 (TDR00H, TDR00L) so that the interval time is 4.17ms.
- Set the value of timer data register 01 (TDR01H, TDR01L) so that the interval time is 2ms.

(3) Set TAU0 channel 0 to start counting.

(4) Wait for the interrupt request flag (TMIF00) of TAU0 channel 0 becomes "1".

(5) After the interrupt request flag (TMIF00) of TAU0 channel 0 becomes "1", clear the flag then enter the LED display processing.

(6) Start the TAU0 channel 1 and wait for the interrupt request flag (TMIF00) becomes "1".

(7) After the interrupt request flag (TMIF01) of TAU0 channel 1 becomes "1", clear the flag then stop the TAU0 channel 1 and enter the key scan processing then set the value of mode flag defined as "g_Mode" according to the key value.

(8) Execute the subroutine of LED mode processing and enter the corresponding display mode processing according to the mode flag "g_Mode".

(9) Return Step (4).
5.2 4-Digit 8-Segment LED Display controlling
This application note explains how to control the 4-digit 8-segment LED to display by R7F0C809 MCU. The 4-digit 8-segment LED uses common anode connection.

Use 60Hz as the scan frequency of 4-digit LED. The scan period is: \(1/60\text{Hz}/4 = 4.17\text{ms}\).

5.3 Key Scan controlling
In this application note, 2×4 matrix keypad is used. According to the pressed key, execute the corresponding subroutine.

Figure 5.1 shows an example of key scan hardware configuration that is used for this application note.

The pattern of how to locate the pressed key is shown as below:

1. The interval time of each key scanning is 4.17ms.
2. When switch the COM pins, in order to avoid the misreading caused by the voltage dropping slowly, we must wait for 2ms before starting the key scanning.
3. The method of locating the pressed key is by the detection of a key pressed through the state of the key scan input pins (P00 and P06), then sampling the value of P0 (the state of key scan output pins) 3 times. If the 3 sampling values are the same, we can recognize the pressed key by the value. (The total time of 3 times sampling is \(4.17\text{ms} \times 4 \times 3 = 50\text{ms}\)).

Note: In order to avoid the generation of reverse current, each key must be in series with a diode.
5.4 Timing Diagram

Figure 5.2 shows the timing diagram of digit LED display controlling and key scanning that is used for this application

Note:
1. \textit{g\_DisplayData[6][4]} is a display data array which is defined on RAM and is used as storing the display data of LED.
2. Set TAU0 channel 0 to interval timer mode which generates the INTTM00 timer interrupt at T0 intervals (T0 is 4.17ms in this sample).
3. Set TAU0 channel 1 to interval timer mode which generates the INTTM01 timer interrupt at T1 intervals (T1 is 2ms in this sample).
5.5 **List of Option Byte Settings**
Table 5.1 summarizes the settings of the option bytes.

<table>
<thead>
<tr>
<th>Address</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000C0H</td>
<td>11101110B</td>
<td>Watchdog timer operation is stopped. (Count is stopped after reset.)</td>
</tr>
<tr>
<td>000C1H</td>
<td>11110011B</td>
<td>SPOR detection voltage: rising edge 4.28V(typ.), falling edge 4.00V(min.) P125/KR1/RESET pin: RESET input</td>
</tr>
<tr>
<td>000C2H</td>
<td>11111001B</td>
<td>HOCO: 20 MHz</td>
</tr>
<tr>
<td>000C3H</td>
<td>10000101B</td>
<td>On-chip debugging is enabled.</td>
</tr>
</tbody>
</table>

5.6 **List of Constants**
Table 5.2 lists the constants that are used in this sample program.

<table>
<thead>
<tr>
<th>Type</th>
<th>Constant Name</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>uint8_t</td>
<td>c_COM_Data[4]</td>
<td>0x20, 0x10, 0x08, 0x04</td>
<td>Control the state of COM ports</td>
</tr>
</tbody>
</table>
### 5.7 List of Variables

Table 5.3 lists the global variables.

#### Table 5.3 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_DisplayData[6][4]</td>
<td>Store the display data</td>
<td>Key_Scan() LED_Display() LED_Mode()</td>
</tr>
<tr>
<td>uint8_t</td>
<td>g_Mode</td>
<td>Key mode</td>
<td>Key_Scan() LED_Mode()</td>
</tr>
<tr>
<td>uint8_t</td>
<td>g_TimeCount</td>
<td>Blink time counter</td>
<td>Key_Scan() LED_Mode()</td>
</tr>
<tr>
<td>uint8_t</td>
<td>g_Row</td>
<td>A variable stores the row value of the two-dimensional array.</td>
<td>Key_Scan() LED_Display() LED_Mode()</td>
</tr>
<tr>
<td>uint8_t</td>
<td>g_RowRecord</td>
<td>A variable records the value of &quot;g_Row&quot;</td>
<td>Key_Scan() LED_Mode()</td>
</tr>
</tbody>
</table>

Table 5.4 lists the static variables.

#### Table 5.4 Static Variables

<table>
<thead>
<tr>
<th>Type</th>
<th>Variable Name</th>
<th>Contents</th>
<th>Function Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>static uint8_t</td>
<td>s_COM</td>
<td>Variable of COM port value.</td>
<td>LED_Display()</td>
</tr>
<tr>
<td>static uint8_t</td>
<td>s_ComNRecord</td>
<td>Save the COM port value at the first key sampling.</td>
<td>Key_Scan()</td>
</tr>
<tr>
<td>static uint8_t</td>
<td>s_Count</td>
<td>Record how many times are the same key pressed (debounce count).</td>
<td>Key_Scan()</td>
</tr>
<tr>
<td>static uint8_t</td>
<td>s_Flag</td>
<td>Key processed flag that records the state whether the key process is completed.</td>
<td>Key_Scan()</td>
</tr>
</tbody>
</table>
5.8 List of Functions

Table 5.5 summarizes the functions that are used in this sample program.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Outline</th>
</tr>
</thead>
<tbody>
<tr>
<td>main</td>
<td>Main processing</td>
</tr>
<tr>
<td>System_Init</td>
<td>Initial setting of peripheral functions</td>
</tr>
<tr>
<td>PORT_Init</td>
<td>Initial setting of the I/O port</td>
</tr>
<tr>
<td>TAU0_Init</td>
<td>Initial setting of the TAU0</td>
</tr>
<tr>
<td>LED_Display</td>
<td>LED display data processing</td>
</tr>
<tr>
<td>Key_Scan</td>
<td>Key scan processing</td>
</tr>
<tr>
<td>LED_Mode</td>
<td>LED display mode processing</td>
</tr>
</tbody>
</table>

5.9 Function Specifications

This section describes the specifications for the functions that are used in the sample code.

[Function Name] main

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Main processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>main.h, system.h, led.h, key_scan.h</td>
</tr>
<tr>
<td>Declaration</td>
<td>void main (void)</td>
</tr>
<tr>
<td>Explanation</td>
<td>Perform main processing.</td>
</tr>
<tr>
<td>Arguments</td>
<td>None</td>
</tr>
<tr>
<td>Return value</td>
<td>None</td>
</tr>
<tr>
<td>Remarks</td>
<td>None</td>
</tr>
</tbody>
</table>

[Function Name] System_Init

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Initial setting of peripheral functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>main.h, system.h</td>
</tr>
<tr>
<td>Declaration</td>
<td>void System_Init (void)</td>
</tr>
<tr>
<td>Explanation</td>
<td>Perform the initial setting of peripheral functions used in this application note.</td>
</tr>
<tr>
<td>Arguments</td>
<td>None</td>
</tr>
<tr>
<td>Return value</td>
<td>None</td>
</tr>
<tr>
<td>Remarks</td>
<td>None</td>
</tr>
</tbody>
</table>

[Function Name] PORT_Init

<table>
<thead>
<tr>
<th>Synopsis</th>
<th>Initial setting of the I/O port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>main.h, system.h</td>
</tr>
<tr>
<td>Declaration</td>
<td>void PORT_Init (void)</td>
</tr>
<tr>
<td>Explanation</td>
<td>Perform the initial setting of I/O port used in this application note.</td>
</tr>
<tr>
<td>Arguments</td>
<td>None</td>
</tr>
<tr>
<td>Return value</td>
<td>None</td>
</tr>
<tr>
<td>Remarks</td>
<td>None</td>
</tr>
<tr>
<td>Function Name</td>
<td>Synopsis</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
</tr>
<tr>
<td>TAU0_Init</td>
<td>Initial setting of the TAU0</td>
</tr>
<tr>
<td>LED_Display</td>
<td>LED display data processing</td>
</tr>
<tr>
<td>Key_Scan</td>
<td>Key scan processing</td>
</tr>
<tr>
<td>LED_Mode</td>
<td>LED display mode processing</td>
</tr>
</tbody>
</table>
5.10 Flowcharts

5.10.1 System Initialization Function

Figure 5.3 shows the flowchart for the system initialization function.

```plaintext
System_Init()

Disable maskable interrupts
IE ← 0

I/O Ports initialization
PORT_Init()

TAU module initialization
TAU0_Init()

Enable maskable interrupts
IE ← 1

Return
```

Figure 5.3 System Initialization Function
5.10.2 I/O Port Setup

Figure 5.4 shows the flowchart for setting up the I/O ports.

```
PORT_Init()

Set LED control ports to digital input/output mode (P07, P10 to P16)

Set the ports which control the COM pins to P-ch open-drain output mode (P02 to P05)

Set the ports which control the SEG pins to N-ch open-drain output mode (P06, P07, P10 to P15)

Set the initial value of COM pins control ports to “0” (P02 to P05)
Set the initial value of SEG pins control ports to “1” (P06, P07, P15 to P10)

Set LED control ports to output mode (P02 to P07, P10 to P15)

Return
```

Setting up I/O ports
- Port Mode Control Register 0 (PMC0)
  Set the port to digital I/O or analog input

Symbol: PMC0

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMC07</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Bit 7

<table>
<thead>
<tr>
<th>PMC07</th>
<th>P07 pin digital I/O and analog input selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Digital I/O (alternate function other than analog input)</td>
</tr>
<tr>
<td>1</td>
<td>Analog input</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:
- x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
• Port Mode Control Register 1 (PMC1)
  Set the port to digital I/O or analog input

Symbol: PMC1

```
<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PMC16</td>
<td>PMC15</td>
<td>PMC14</td>
<td>PMC13</td>
<td>PMC12</td>
<td>PMC11</td>
<td>PMC10</td>
</tr>
<tr>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```

Bits 6 to 0

<table>
<thead>
<tr>
<th>PMC1n</th>
<th>P1n pin digital I/O and analog input selection(n = 0 to 6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Digital I/O (alternate function other than analog input)</td>
</tr>
<tr>
<td>1</td>
<td>Analog input</td>
</tr>
</tbody>
</table>

• Port Output Mode Register 0 (POM0)
  Set the port to normal output mode or N-ch/P-ch open drain output mode

Symbol: POM0

```
<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>POM07</td>
<td>POM06</td>
<td>POM05</td>
<td>POM04</td>
<td>POM03</td>
<td>POM02</td>
<td>POM01</td>
<td>POM00</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>x</td>
<td>1</td>
</tr>
</tbody>
</table>
```

Bits 7 to 2

<table>
<thead>
<tr>
<th>POM0n</th>
<th>P0n pin output mode selection(n = 2 to 7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Normal output mode</td>
</tr>
</tbody>
</table>
| 1     | N-ch open-drain output (VDD tolerance) mode(n = 6,7)  
      | P-ch open-drain output (VDD tolerance) mode(n = 2 to 5) |

• Port Output Mode Register 1 (POM1)
  Set the port to normal output mode or N-ch open drain output mode

Symbol: POM1

```
<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>POM15</td>
<td>POM14</td>
<td>POM13</td>
<td>POM12</td>
<td>POM11</td>
<td>POM10</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
```

Bits 5 to 0

<table>
<thead>
<tr>
<th>POM1n</th>
<th>P1n pin output mode selection(n = 0 to 5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Normal output mode</td>
</tr>
<tr>
<td>1</td>
<td>N-ch open-drain output (VDD tolerance) mode</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:

x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
• Port Register 0 (P0)
  Set the output latch value of a port

Symbol: P0

<table>
<thead>
<tr>
<th></th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

Bits 5 to 2

<table>
<thead>
<tr>
<th>P0n</th>
<th>Output data control (in output mode) (n = 2 to 5)</th>
<th>Input data read (in input mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Output &quot;0&quot; (in output mode).</td>
<td>Input low level (in input mode).</td>
</tr>
<tr>
<td>1</td>
<td>Output &quot;1&quot; (in output mode).</td>
<td>Input high level (in input mode).</td>
</tr>
</tbody>
</table>

Bits 7 to 6

<table>
<thead>
<tr>
<th>P0n</th>
<th>Output data control (in output mode) (n = 6 to 7)</th>
<th>Input data read (in input mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Output &quot;0&quot; (in output mode).</td>
<td>Input low level (in input mode).</td>
</tr>
<tr>
<td>1</td>
<td>Output &quot;1&quot; (in output mode).</td>
<td>Input high level (in input mode).</td>
</tr>
</tbody>
</table>

• Port Register 1 (P1)
  Set the output latch value of a port

Symbol: P1

<table>
<thead>
<tr>
<th></th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Bits 5 to 0

<table>
<thead>
<tr>
<th>P1n</th>
<th>Output data control (in output mode) (n = 0 to 5)</th>
<th>Input data read (in input mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Output &quot;0&quot; (in output mode).</td>
<td>Input low level (in input mode).</td>
</tr>
<tr>
<td>1</td>
<td>Output &quot;1&quot; (in output mode).</td>
<td>Input high level (in input mode).</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:
- x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
• Port Mode Register 0 (PM0)
  Select I/O mode for the port

Symbol: PM0

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PM07</td>
<td>PM06</td>
<td>PM05</td>
<td>PM04</td>
<td>PM03</td>
<td>PM02</td>
<td>PM01</td>
<td>PM00</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

Bits 7 to 2

<table>
<thead>
<tr>
<th>PM0n</th>
<th>PM0n pin I/O mode selection (n = 2 to 7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Output mode (output buffer on).</td>
</tr>
<tr>
<td>1</td>
<td>Input mode (output buffer off).</td>
</tr>
</tbody>
</table>

• Port Mode Register 1 (PM1)
  Select I/O mode for the port

Symbol: PM1

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PM10</td>
</tr>
<tr>
<td>-</td>
<td>PM16</td>
<td>PM15</td>
<td>PM14</td>
<td>PM13</td>
<td>PM12</td>
<td>PM11</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Bits 5 to 0

<table>
<thead>
<tr>
<th>PM1n</th>
<th>PM1n pin I/O mode selection (n = 0 to 5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Output mode (output buffer on).</td>
</tr>
<tr>
<td>1</td>
<td>Input mode (output buffer off).</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:

x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
5.10.3 Timer Array Unit Setup

Figure 5.5 shows the flowchart for setting up the timer array unit.

![Flowchart](image)

**Figure 5.5 Timer Array Unit Setup**
Enabling supply of clock to the timer array unit 0

- Peripheral enable register 0 (PER0)
  Start supplying clock to the timer array unit 0

Symbol: PER0

<table>
<thead>
<tr>
<th>Bit</th>
<th>TMKAEN</th>
<th>R0EN</th>
<th>ADCEN</th>
<th>0</th>
<th>0</th>
<th>SAU0EN</th>
<th>0</th>
<th>TAU0EN</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>-</td>
<td>-</td>
<td>x</td>
<td>-</td>
<td>1</td>
</tr>
</tbody>
</table>

Bit 0

<table>
<thead>
<tr>
<th>TAU0EN</th>
<th>Control of timer array unit 0 input clock supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Stops input clock supply.</td>
</tr>
<tr>
<td></td>
<td>·SFR used by timer array unit cannot be written.</td>
</tr>
<tr>
<td></td>
<td>·Timer array unit is in the reset status.</td>
</tr>
<tr>
<td>1</td>
<td>Enables input clock supply.</td>
</tr>
<tr>
<td></td>
<td>·SFR used by timer array unit can be read and written.</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:

x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
Configuring the timer clock frequency

- Timer clock select register 0 (TPS0)
  - Select an operation clock for timer array unit 0

Symbol: TPS0

<table>
<thead>
<tr>
<th>Bits 3 to 0</th>
<th>Selection of operation clock (CK0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>f_{CLK} = 1.25 MHz</td>
</tr>
<tr>
<td>0 0 0 0 0</td>
<td>f_{CLK}</td>
</tr>
<tr>
<td>0 0 0 0 1</td>
<td>f_{CLK} / 2</td>
</tr>
<tr>
<td>0 0 1 0 0</td>
<td>f_{CLK} / 2^2</td>
</tr>
<tr>
<td>0 0 1 1 1</td>
<td>f_{CLK} / 2^3</td>
</tr>
<tr>
<td>0 1 0 0 0</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 0 0 1</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 1 0 0</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 1 1 1</td>
<td>f_{CLK} / 2^5</td>
</tr>
<tr>
<td>1 0 0 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 0 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 1 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 1 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 0 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 0 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 1 0 1</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>f_{CLK} / 2^6</td>
</tr>
</tbody>
</table>

Setting up TAU0 channel n operation mode (n = 0, 1)
- Timer mode register 0n (TMR0nH, TMR0nL) (n = 0, 1)
  - Select an operation clock (f_{MCK}).
  - Select a count clock.
  - Select the software trigger start.
  - Set up the operation mode.

Symbol: TMR00H

<table>
<thead>
<tr>
<th>Bits</th>
<th>Selection of operation clock (CK0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>f_{CLK} = 1.25 MHz</td>
</tr>
<tr>
<td>0 0 0 0 0</td>
<td>f_{CLK}</td>
</tr>
<tr>
<td>0 0 0 0 1</td>
<td>f_{CLK} / 2</td>
</tr>
<tr>
<td>0 0 1 0 0</td>
<td>f_{CLK} / 2^2</td>
</tr>
<tr>
<td>0 0 1 1 1</td>
<td>f_{CLK} / 2^3</td>
</tr>
<tr>
<td>0 1 0 0 0</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 0 0 1</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 1 0 0</td>
<td>f_{CLK} / 2^4</td>
</tr>
<tr>
<td>0 1 1 1 1</td>
<td>f_{CLK} / 2^5</td>
</tr>
<tr>
<td>1 0 0 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 0 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 1 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 0 1 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 0 0 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 0 1 0</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 1 0 1</td>
<td>f_{CLK} / 2^6</td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>f_{CLK} / 2^6</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:
- x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
### Symbol: TMR01H

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>CKS0n1</th>
<th>Channel n operation clock (fMCK) selection (n = 0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Operation clock CK00 set by timer clock select register 0 (TPS0).</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Operation clock CK01 set by timer clock select register 0 (TPS0).</td>
</tr>
</tbody>
</table>

### Bit 4

<table>
<thead>
<tr>
<th>Bit 0</th>
<th>CCS0n</th>
<th>Channel n count clock (fTCLK) selection (n = 0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Operation clock (fMCK) specified by the CKS0n1 bits.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Valid edge of input signal input from the TI0n pin.</td>
</tr>
</tbody>
</table>

### Bit 3

<table>
<thead>
<tr>
<th>Bit 0</th>
<th>SPLIT</th>
<th>Selection of 8 or 16-bit timer operation for channel 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Operates as 16-bit timer.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Operates as 8-bit timer.</td>
</tr>
</tbody>
</table>

### Bits 2 to 0

<table>
<thead>
<tr>
<th>Bit 0</th>
<th>STS0n2</th>
<th>STS0n1</th>
<th>STS0n0</th>
<th>Setting of start trigger or capture trigger of channel n (n = 0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Only software trigger start is valid (other trigger sources are unselected).</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Valid edge of the TI0n pin input is used as the start trigger and capture trigger.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Both the edges of the TI0n pin input are used as the start trigger and capture trigger.</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>When the channel is used as a slave channel with the one-short pulse output, PWM output function or multiple PWM output function: The Interrupt request signal of the master channel (INTTM0n) is used as the start trigger.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>When the channel is used as a slave channel in two-channel input with one-short pulse output function: The Interrupt request signal of the master channel (INTTM0n) is used as the start trigger. A valid edge of the TI03 pin input of the slave channel is used as the end trigger.</td>
</tr>
</tbody>
</table>

**Note:** For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

**Legend symbol:**

- x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
Symbol: TMR0nL ( n = 0, 1 )

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIS0n1</td>
<td>CIS0n0</td>
<td>0</td>
<td>0</td>
<td>MD0n3</td>
<td>MD0n2</td>
<td>MD0n1</td>
<td>MD0n0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Bits 3 to 0

<table>
<thead>
<tr>
<th>MD0n3</th>
<th>MD0n2</th>
<th>MD0n1</th>
<th>MD0n0</th>
<th>Operation mode of channel n ( n = 0, 1 )</th>
<th>Corresponding function</th>
<th>Count operation of TCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1/0</td>
<td>Interval timer mode</td>
<td>Interval timer / Square wave output / Divider function / PWM output (master)</td>
<td>Down count</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1/0</td>
<td>Capture mode</td>
<td>Input pulse interval measurement / Two channel input with one-shot pulse output function (slave)</td>
<td>Up count</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Event counter mode</td>
<td>External event counter</td>
<td>Down count</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1/0</td>
<td>One-count mode</td>
<td>Delay counter / One-shot pulse output / Two-channel input with one-shot pulse output function (master) / PWM output (slave)</td>
<td>Down count</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Capture &amp; one-count mode</td>
<td>Measurement of high- / low-level width of input signal</td>
<td>Up count</td>
</tr>
</tbody>
</table>

Other than above Setting prohibited

The operation of each mode changes depending on the operation of MD0n0 bit (see the table below).

<table>
<thead>
<tr>
<th>Operation mode (Value set by the MD0n3 to MD0n1 bits (see table above)) ( n = 0, 1 )</th>
<th>MD0n0 ( n = 0, 1 )</th>
<th>Setting of starting counting and interrupt</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Interval timer mode (0, 0, 0)</td>
<td>0</td>
<td>Timer interrupt is not generated when counting is started (timer output does not change, either).</td>
</tr>
<tr>
<td>• Capture mode (0, 1, 0)</td>
<td>1</td>
<td>Timer interrupt is generated when counting is started (timer output also changes).</td>
</tr>
<tr>
<td>• Event counter mode (0, 1, 1)</td>
<td>0</td>
<td>Timer interrupt is not generated when counting is started (timer output does not change, either).</td>
</tr>
<tr>
<td>• One-count mode (1, 0, 0)</td>
<td>0</td>
<td>Start trigger is invalid during counting operation. At that time, interrupt is not generated, either.</td>
</tr>
<tr>
<td>• Capture &amp; one-count mode (1, 1, 0)</td>
<td>1</td>
<td>Start trigger is valid during counting operation. At that time, interrupt is also generated.</td>
</tr>
<tr>
<td>Other than above</td>
<td>Setting prohibited</td>
<td></td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:
x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
Setting up the interval timer cycle time

- Timer data register 0n (TDR0nH, TDR0nL) (n = 0, 1)
  Specify the interval timer compare value.

Symbol: TDR00H

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Symbol: TDR00L

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Timer interrupt (INTTM00) occurrence = Count clock cycle time × (TDR00 setting + 1)

=\frac{1}{1.25\text{MHz}} \times (0x142a + 1) = 4.17\text{ms}

Symbol: TDR01H

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Symbol: TDR01L

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Timer interrupt (INTTM01) occurrence = Count clock cycle time × (TDR01 setting + 1)

=\frac{1}{1.25\text{MHz}} \times (0x09c4 + 1) = 0.15\text{ms}

Setting up timer interrupt

- Interrupt request flag register (IF0H, IF0L)
  Set interrupt request flag

Symbol: IF0H

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMIF02</td>
<td>0</td>
<td>TMIF03H</td>
<td>PIF3</td>
<td>PIF2</td>
<td>KRIF</td>
<td>ADIF</td>
<td>TMIF01</td>
</tr>
<tr>
<td>x</td>
<td>-</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>0</td>
</tr>
</tbody>
</table>

Bit 0

<table>
<thead>
<tr>
<th>TMIF01</th>
<th>Interrupt request flag</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No interrupt request signal is generated</td>
</tr>
<tr>
<td>1</td>
<td>Interrupt request signal is generated, interrupt request status</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:

x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
Symbol: IF0L

<table>
<thead>
<tr>
<th></th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMIF00</td>
<td>TMIF01H</td>
<td>SREIF0</td>
<td>SRIF0</td>
<td>STIF0</td>
<td>CSIIF0</td>
<td>PIF1</td>
<td>PIF0</td>
<td>WDTIF</td>
</tr>
<tr>
<td>0</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

Bit 7

<table>
<thead>
<tr>
<th>TMIF00</th>
<th>Interrupt request flag</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No interrupt request signal is generated</td>
</tr>
<tr>
<td>1</td>
<td>Interrupt request signal is generated, interrupt request status</td>
</tr>
</tbody>
</table>

Starting timer channel operation

- Timer channel start register 0 (TS0)
  Start the timer

Symbol: TS0

<table>
<thead>
<tr>
<th></th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>TS03</td>
<td>TS02</td>
<td>TS01</td>
<td>TS00</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>x</td>
<td>x</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Bits 1 to 0

<table>
<thead>
<tr>
<th>TS0n</th>
<th>Operation enable (start) trigger of channel n (n = 0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No trigger operation</td>
</tr>
<tr>
<td>1</td>
<td>The TE0n bit is set to &quot;1&quot; and the count operation becomes enabled.</td>
</tr>
</tbody>
</table>

Stopping timer channel operation

- Timer channel stop register 0 (TT0)
  Stop the timer

Symbol: TT0

<table>
<thead>
<tr>
<th></th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>TT03</td>
<td>TT02</td>
<td>TT01</td>
<td>TT00</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>x</td>
<td>x</td>
<td>1</td>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

Bit 1

<table>
<thead>
<tr>
<th>TT01</th>
<th>Operation stop trigger of channel 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No trigger operation</td>
</tr>
<tr>
<td>1</td>
<td>TE01 bit is cleared to &quot;0&quot; and the count operation is stopped.</td>
</tr>
</tbody>
</table>

Note: For details on the procedure for setting up the registers, refer to R7F0C806-809 User’s Manual: Hardware.

Legend symbol:
x: Unused bit; blank cell: unchanged bit; -: reserved bit or unallocated bit
5.10.4 **Main Processing**

Figure 5.6 shows the flowchart for the main processing.

![Flowchart for main processing](image)

The option bytes are referenced before the initialization function is called.

- `TS00 bit ← 1`
- `TMIF00 bit ← 0`
- `TS01 bit ← 1`
- `TT01 bit ← 1`
- `TMIF01 bit ← 0`
5.10.5 LED Display Processing

Figure 5.7 shows the flowchart for the LED display processing.

![Flowchart for LED Display Processing](image)

- **LED_Display()**
  - s_COM + 1
  - **s_COM > 3?**
    - Yes: s_COM = 0
    - No
  - Close all the COM ports
  - Close all the SEG ports
  - **g_DisplayData**
    - [g_Row][s_COM]
  - 0x30: display"0"
  - 0x31: display"1"
  - 0x32: display"
  - 0x33: display"
  - 0x34: display"
  - 0x35: display"
  - 0x36: display"
  - 0x37: display"
  - 0x38: display"
  - 0x39: display"
  - Open corresponding port according to s_COM
  - Return

**g_Row**: A variable stores the row value of the two-dimensional array

**s_COM**: Variable of COM port value
5.10.6 Key Scan Processing

Figure 5.8 shows the flowchart for the key scan processing.

Figure 5.8 Key Scan Processing

Note: Make sure that the key will not be multiple processed during this key is pressed in the key processing routine. After recognizing the pressed key, perform the key processing and set the key processed flag to "1". If this key is not released, keep the value of key processed flag unchanged and make sure the key will not be reprocessed. If this key is released, clear the key processed flag and perform the next key processing.
5.10.7 LED Mode Processing

Figure 5.9 shows the flowchart for the LED mode processing.

Figure 5.9 LED Mode Processing
6. **Sample Code**

The sample code is available on the Renesas Electronics Website.

7. **Reference Documents**

User's Manual:
- R7F0C806-809 User's Manual: Hardware (R01UH0481E)

The latest versions of the documents are available on the Renesas Electronics Website.

Technical Updates/Technical News
- The latest information can be downloaded from the Renesas Electronics website.

**Website and Support**

Renesas Electronics Website
- [http://www.renesas.com/](http://www.renesas.com/)

Inquiries
- [http://www.renesas.com/contact/](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>Page</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>Sep. 30, 2014</td>
<td>30</td>
<td>First edition issued</td>
</tr>
</tbody>
</table>
# 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 document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

<table>
<thead>
<tr>
<th>1. Handling of Unused Pins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.</td>
</tr>
<tr>
<td>— 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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Processing at Power-on</th>
</tr>
</thead>
<tbody>
<tr>
<td>The state of the product is undefined at the moment when power is supplied.</td>
</tr>
<tr>
<td>— 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.</td>
</tr>
<tr>
<td>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.</td>
</tr>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Prohibition of Access to Reserved Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access to reserved addresses is prohibited.</td>
</tr>
<tr>
<td>— 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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Clock Signals</th>
</tr>
</thead>
<tbody>
<tr>
<td>After applying a reset, only release the reset line after the operating clock signal has become stable.</td>
</tr>
<tr>
<td>When switching the clock signal during program execution, wait until the target clock signal has stabilized.</td>
</tr>
<tr>
<td>— 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.</td>
</tr>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Differences between Products</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.</td>
</tr>
<tr>
<td>— The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the 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.</td>
</tr>
</tbody>
</table>
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 must 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 (vertical life support devices or systems, surgical implantation 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 application except for the military units or products that are 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, dispenses, 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 products" means any product developed or manufactured by or for Renesas Electronics.