Abstract
This document describes a method for outputting PWM using phase shift waveform output mode in the intelligent I/O of the R32C/100 Series MCU.

Products
MCUs: R32C/120 Group, R32C/121 Group, R32C/151 Group, R32C/152 Group, R32C/153 Group, R32C/156 Group, R32C/157 Group, R32C/160 Group, R32C/161 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 Notes ..................................................................................... 4
4. Hardware ....................................................................................................................... 4
   4.1 Pin Used .................................................................................................................. 4
   4.2 Notes ...................................................................................................................... 4
5. Software ......................................................................................................................... 5
   5.1 Operation Overview ............................................................................................... 5
   5.2 Notes ...................................................................................................................... 5
   5.3 Function Table ....................................................................................................... 6
   5.4 Function Specifications .......................................................................................... 6
   5.5 Flowcharts ............................................................................................................. 7
      5.5.1 Main Processing .............................................................................................. 7
      5.5.2 Intelligent I/O Interrupt Handling ..................................................................... 10
6. Sample Code ................................................................................................................ 11
7. Reference Documents ............................................................................................... 11
8. Website and Support ................................................................................................. 11
1. Specifications

In phase shift waveform output mode, PWM output is phase-shifted at every channel. This mode enables functions that help to reduce switching noise and instantaneous power consumption.

Table 1.1 lists the Peripheral Function and Its Application with the sample code. Figure 1.1 shows Phase Shift Waveform Output Mode (i = 0, 1).

Table 1.1 Peripheral Function and Its Application

<table>
<thead>
<tr>
<th>Peripheral Function</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligent I/O group 0 (IIO0)</td>
<td>PWM pulse output</td>
</tr>
</tbody>
</table>

Figure 1.1 Phase Shift Waveform Output Mode (i = 0, 1)
2. Operation Confirmation Conditions

The sample code accompanying this application note has been run and confirmed under the conditions below.

<table>
<thead>
<tr>
<th>Table 2.1 Operation Confirmation Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
</tr>
<tr>
<td>MCU used</td>
</tr>
<tr>
<td>Operating frequencies</td>
</tr>
<tr>
<td>Main clock</td>
</tr>
<tr>
<td>PLL clock</td>
</tr>
<tr>
<td>Base clock</td>
</tr>
<tr>
<td>CPU clock</td>
</tr>
<tr>
<td>Peripheral bus clock</td>
</tr>
<tr>
<td>Peripheral function clock source</td>
</tr>
<tr>
<td>Operating voltage</td>
</tr>
<tr>
<td>Integrated development environment</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>C compiler</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Compile options</td>
</tr>
<tr>
<td>-D STACKSIZE =0X300 -D STACKSIZE =0X300</td>
</tr>
<tr>
<td>-DVECTOR_ADR=0xFFFFFBDC -c -info -dir &quot;$(CONFIGDIR)&quot;</td>
</tr>
<tr>
<td>(Default setting is used in the integrated development environment.)</td>
</tr>
<tr>
<td>Operating mode</td>
</tr>
<tr>
<td>Sample code version</td>
</tr>
</tbody>
</table>

3. Reference Application Notes

The application notes associated with this application note are listed below. Refer to the following application notes for additional information.

- R32C/100 Series Configuring PLL Mode (REJ05B1221-0100)
- R32C/100 Series How to Use Intelligent I/O Interrupt (REJ05B1416-0100)

4. Hardware

4.1 Pin Used

Table 4.1 lists the Pins Used and Their Functions.

<table>
<thead>
<tr>
<th>Table 4.1 Pins Used and Their Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin Name</td>
</tr>
<tr>
<td>P0_0/AN0_0/D0</td>
</tr>
<tr>
<td>P1_0/SS00/IIO0_0/IIO1_0</td>
</tr>
<tr>
<td>P1_1/SSCK0/IIO0_1/IIO1_1</td>
</tr>
<tr>
<td>P1_2/SS00/IIO0_2/IIO1_2</td>
</tr>
</tbody>
</table>

4.2 Notes

When using the base timer, the bit in intelligent I/O interrupt request register i becomes 1 when the base timer overflows (i = 0 to 11).
5. Software

5.1 Operation Overview

A PWM pulse is output from the IIO0_i pin in phase shift waveform output mode (i = 0 to 2). The P0_0 pin is held high when the group 0 base timer interrupt is generated.

(1) Intelligent I/O settings

Set group 0 to phase shift waveform output mode and select output channels 0 to 2.

Figure 5.1 shows the Waveform Output Timing Using the Sample Code.

---

5.2 Notes

In the sample code, the BT0R bit in intelligent I/O interrupt request register 7 becomes 1 (interrupt requested) when the base timer overflows. When the BT0E bit in intelligent I/O interrupt request register 7 is 0 (disabled), the BT0R bit does not need to be set to 0 (no interrupt requested).
5.3 Function Table

Table 5.1 lists the Functions.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Outline</th>
</tr>
</thead>
<tbody>
<tr>
<td>iio_init</td>
<td>Intelligent I/O initialization</td>
</tr>
<tr>
<td>_intelligent_io_int7</td>
<td>Intelligent I/O interrupt 7 handling</td>
</tr>
</tbody>
</table>

5.4 Function Specifications

The following tables list the sample code function specifications.

**iio_init**

<table>
<thead>
<tr>
<th>Outline</th>
<th>Intelligent I/O initialization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>void iio_init(void)</td>
</tr>
<tr>
<td>Explanation</td>
<td>Initializes the intelligent I/O.</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
<tr>
<td>Remark</td>
<td></td>
</tr>
</tbody>
</table>

**_intelligent_io_int7**

<table>
<thead>
<tr>
<th>Outline</th>
<th>Intelligent I/O interrupt 7 handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>None</td>
</tr>
<tr>
<td>Declaration</td>
<td>void _intelligent_io_int7(void)</td>
</tr>
<tr>
<td>Explanation</td>
<td>Enable the intelligent I/O interrupt for channel 7</td>
</tr>
<tr>
<td>Argument</td>
<td>None</td>
</tr>
<tr>
<td>Returned value</td>
<td>None</td>
</tr>
<tr>
<td>Remark</td>
<td>Set the P0_0 pin high when a base timer 0 overflow interrupt is generated.</td>
</tr>
</tbody>
</table>
5.5 Flowcharts

5.5.1 Main Processing

Figure 5.2 shows the Main Processing.

```
main

(1) Disable maskable interrupts
    I flag ← 0
    Set PLL clock
    SetPLLClock()

(2) Set P0_0 pin
    P0 register ← 00h
    PD0 register ← 01h
    PD0_0 = 1 : Set P0_0 pin as output port.

(3) Initialize intelligent I/O
    iio_init()

(4) Enable maskable interrupts
    I flag ← 1

(5) Base timer count starts
    G0BCR1 register
    BTS bit ← 1 : Count starts.
```

Figure 5.2 Main Processing
Figure 5.3 and Figure 5.4 show Intelligent I/O Initialization.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Register Initializations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Set group 0 base timer</td>
<td>G0BCR0 register $\leftarrow 7Fh$&lt;br&gt;Bits BCK0 and BCK1 = 11b&lt;br&gt;Bits DIV0 to DIV4 = 11111b&lt;br&gt;IT bit = 0 &lt;br&gt;Set count source to f1.&lt;br&gt;Set count source divide ratio to no division. &lt;br&gt;Set base timer interrupt source select bit to 0 (overflow of bit 15 or bit 9).</td>
</tr>
<tr>
<td>2</td>
<td>Group 0 base timer reset</td>
<td>G0BCR1 register $\leftarrow 00h$&lt;br&gt;BTS bit = 0 &lt;br&gt;Reset base timer.</td>
</tr>
<tr>
<td>3</td>
<td>Set group 0 waveform generation control register</td>
<td>G0POCR0 register $\leftarrow 00h$&lt;br&gt;Bits MOD0 to MOD2 = 000b&lt;br&gt;IVL bit = 0 &lt;br&gt;RLD bit = 0 &lt;br&gt;BTR bit = 0 &lt;br&gt;INV bit = 0 &lt;br&gt;Single-phase waveform output mode&lt;br&gt;Output low as default value. &lt;br&gt;Reload the value into the G0POj register on a write access. &lt;br&gt;Reset the base timer when bit 15 overflows. &lt;br&gt;Do not invert the output level.</td>
</tr>
<tr>
<td>4</td>
<td>Set group 0 waveform generation register</td>
<td>G0PO0 register $\leftarrow 200$&lt;br&gt;G0PO1 register $\leftarrow 400$&lt;br&gt;G0PO2 register $\leftarrow 600$ &lt;br&gt;Set channel 0 output waveform.&lt;br&gt;Set channel 1 output waveform. &lt;br&gt;Set channel 2 output waveform.</td>
</tr>
<tr>
<td>5</td>
<td>Set group 0 waveform generation control register</td>
<td>G0POCR0 register $\leftarrow E0h$&lt;br&gt;Bits MOD0 to MOD2 = 000b&lt;br&gt;IVL bit = 0 &lt;br&gt;RLD bit = 1 &lt;br&gt;BTR bit = 1 &lt;br&gt;INV bit = 1 &lt;br&gt;Single-phase waveform output mode&lt;br&gt;Output low as default value. &lt;br&gt;Reload the value into the G0POj register when the base timer is reset. &lt;br&gt;Reset the base timer when bit 9 overflows. &lt;br&gt;Invert the output level.</td>
</tr>
</tbody>
</table>

Figure 5.3 Intelligent I/O Initialization (1/2) ($j = 0$ to 2)
**Figure 5.4 Intelligent I/O Initialization (2/2)**

1. **Set group 0 waveform generation**
   - G0FS register ← 00h
   - FSC0 bit = 0
   - FSC1 bit = 0
   - FSC2 bit = 0
   : Waveform generation channel 0
   : Waveform generation channel 1
   : Waveform generation channel 2

2. **Set group 0 function enable register**
   - G0FE register ← 07h
   - IFE0 bit = 1
   - IFE1 bit = 1
   - IFE2 bit = 1
   : Enable the function for channel 0
   : Enable the function for channel 1
   : Enable the function for channel 2

3. **Set phase shift waveform output mode**
   - G0SDR register ← 63h
   - G0PSR register ← 01h
   - Bits PSS0 and PSS1 = 00b
   : Set phase shift clock as a base timer count source (fBT0) divided by 100.
   : Enable phase shift waveform output mode.
   : Set pins IIO0_0 to IIO0_4.

4. **Wait two or more fBT0 clock cycles**

5. **Clear intelligent I/O interrupt request register 7**
   - IIO7IR register ← 00h
   : Clear BT0R interrupt request.

6. **Set intelligent I/O interrupt enable register 7 (1)**
   - IIO7IE register ← 01h
   - IRLT bit = 1
   : Use interrupt requests for interrupt. (1)

7. **Set intelligent I/O interrupt enable register 7 (2)**
   - IIO7IE register ← 11h
   - IRLT bit = 1
   - BT0R bit = 1
   : Use interrupt requests for interrupt. (1)
   : Enable BT0R interrupt. (1)

8. **Set interrupt request level**
   - IIO7IC register ← 01h
   - Bits ILVL0 to ILVL2 = 001b
   : Set interrupt request level to level 1.

9. **Set input port**
   - P1 register ← 00h
   - P1_0S register ← 05h
   - P1_1S register ← 05h
   - P1_2S register ← 05h
   : Set P1_0 as IIO0_0 output port.
   : Set P1_1 as IIO0_1 output port.
   : Set P1_2 as IIO0_2 output port.
   - PD1 register ← 07h
     - PD1_0 bit = 1
     - PD1_1 bit = 1
     - PD1_2 bit = 1
     : Set P1_0 as output port.
     : Set P1_1 as output port.
     : Set P1_2 as output port.

---

**Note:**
1. To use interrupt requests for interrupt, the IRLT bit should be set to 1, then bit 1 should be set to 1.
5.5.2 Intelligent I/O Interrupt Handling

Figure 5.5 shows Intelligent I/O Interrupt Handling.

```
(1) Is the BT0R bit 1?
   Yes
   Set output port
   P0_0 bit ← 1
   return

(2) Clear BT0R bit
   BT0R bit ← 0
   : Clear interrupt request.
   Set output port
   P0_0 bit ← 0
```

Figure 5.5 Intelligent I/O Interrupt Handling
6. **Sample Code**
   
   Sample code can be downloaded from the Renesas Electronics website.

7. **Reference Documents**
   
   - **R32C/156 Group User's Manual: Hardware Rev.1.03**

   The latest versions can be downloaded from the Renesas Electronics website.

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

   - **C Compiler Manual**
     
     - **R32C/100 Series C Compiler Package V.1.02**
     - **C Compiler User’s Manual Rev.2.00**

     The latest version can be downloaded from the Renesas Electronics website.

8. **Website and Support**
   
   - **Renesas Electronics website**
     
     http://www.renesas.com/

   - **Inquiries**
     
     http://www.renesas.com/inquiry
<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>Mar. 15, 2011</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 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 part number, confirm that the change will not lead to problems.
   - The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part 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 listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to any additional and different information to be disclosed by Renesas Electronics 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 such 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 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. Further, you may not use any Renesas Electronics product for any application for which it is not intended 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 failures 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 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 recomputor 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.

(Notes)

* "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

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