

To our customers,

---

## Old Company Name in Catalogs and Other Documents

---

On April 1<sup>st</sup>, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1<sup>st</sup>, 2010  
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

## 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 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. 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. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
  - “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.
  - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
  - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
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 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 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.

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

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

# Manual for Using Sample Program Functions Timer Q (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)

This manual explains the sample program functions of the 16-bit timer/event counter Q (TMQ) for the V850E/IA4 microcontroller.

The explanations are based on usage with the V850E/IA4 microcontroller. Refer to this manual when using the V850E/IA3, V850ES/IK1, and V850ES/IE2 microcontrollers.

## Caution

**This sample program is provided for reference purposes only and operations are therefore not subject to guarantee by NEC Electronics Corporation. When using this sample program, customers are kindly advised to sufficiently evaluate this product based on their system before usage.**

### ① VOLTAGE APPLICATION WAVEFORM AT INPUT PIN

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between  $V_{IL}$  (MAX) and  $V_{IH}$  (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between  $V_{IL}$  (MAX) and  $V_{IH}$  (MIN).

### ② HANDLING OF UNUSED INPUT PINS

Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to  $V_{DD}$  or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

### ③ PRECAUTION AGAINST ESD

A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

### ④ STATUS BEFORE INITIALIZATION

Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

### ⑤ POWER ON/OFF SEQUENCE

In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current.

The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

### ⑥ INPUT OF SIGNAL DURING POWER OFF STATE

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

These commodities, technology or software, must be exported in accordance with the export administration regulations of the exporting country. Diversion contrary to the law of that country is prohibited.

- **The information in this document is current as of July, 2006. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC 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 NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.

"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.

"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).

"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

- (1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
- (2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

## INTRODUCTION

- Cautions**
1. Download the program used in this manual from the NEC Electronics Website (<http://www.necel.com/>).
  2. When using this sample program, reference the following startup file and link directive file and adjust them if as necessary.
    - Startup file: IA4\_start.s
    - Link directive file: IA4\_link.dir

**Conventions**      The function lists are structured as follows.

### Hardware name (symbol)

[Function(s)]	Function description
[Function name]	Name of sample function
[Argument(s)]	Type and overview of argument(s)
[Processing content]	Processing content of sample function
[Starting method]	Conditions for calling a function
[SFR(s) used]	Register name and setting content
[call function(s)]	Name and function of call function(s)
[Variable(s)]	Type, name, and overview of variable(s) used in sample function
[Interrupt(s)]	Name of function
[Interrupt source(s)]	Name
[File name]	Name of corresponding sample program file
[Caution(s)]	Caution(s) up on function usage

### Interrupt function

[Function name]	Name of interrupt function
[Overview]	Processing content
[Factor(s)]	Name of interrupt and conditions for occurrence
[call function(s)]	None
[Variable(s)]	Name of variable, function
[File name]	Name of corresponding sample program file
[Caution(s)]	None

**Product Differences**

The differences between the V850E/IA4 and the V850E/IA3, V850ES/IK1, and V850ES/IE2 related to the 16-bit timer/event counter Q (TMQ) are shown below.

Item	V850E/IA4	V850E/IA3	V850ES/IK1	V850ES/IE2
TOQ10 pin	Provided	Not provided	Provided	
TRGQ0 pin	Provided	Provided	Not provided	
TOQH01 to TOQH03 pins	Not provided	Not provided	Provided	
TOQ00 pin	TOQ00	TOQ00	TOQ00 (CLMER)	
Count clock	$f_{xx}/2$ , $f_{xx}/4$ , $f_{xx}/8$ , $f_{xx}/16$ , $f_{xx}/32$ , $f_{xx}/64$ , $f_{xx}/128$ , $f_{xx}/256$	$f_{xx}/2$ , $f_{xx}/4$ , $f_{xx}/8$ , $f_{xx}/16$ , $f_{xx}/32$ , $f_{xx}/64$ , $f_{xx}/128$ , $f_{xx}/256$	$f_{xx}$ , $f_{xx}/2$ , $f_{xx}/4$ , $f_{xx}/8$ , $f_{xx}/16$ , $f_{xx}/32$ , $f_{xx}/64$ , $f_{xx}/128$	

**Remark** fxx: Peripheral clock frequency



**Related Documents** The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

**Documents related to V850E/IA3, V850E/IA4, V850ES/IK1, and V850ES/IE2**

Document Name	Document No.
V850E1 Architecture User's Manual	U14559E
V850E/IA3, V850E/IA4 Hardware User's Manual	U16543E
V850ES Architecture User's Manual	U15943E
V850ES/IK1 Hardware User's Manual	U16910E
V850ES/IE2 Hardware User's Manual	U17716E
Inverter Control by V850 Series Vector Control by Hole Sensor Application Note	U17338E
Inverter Control by V850 Series Vector Control by Encoder Application Note	U17324E
Inverter Control by V850 Series 120° Excitation Method Control by Zero-Cross Detection Application Note	U17209E
Manual for Using Sample Program Functions Serial Communication (UARTA) (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18233E
Manual for Using Sample Program Functions Serial Communication (CSIB) (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18234E
Manual for Using Sample Program Functions DMA Functions (V850E/IA3, V850E/IA4) Application Note	U18235E
Manual for Using Sample Program Functions Timer M (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18236E
Manual for Using Sample Program Functions Watchdog Timer (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18237E
Manual for Using Sample Program Functions Timer P (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18238E
Manual for Using Sample Program Functions Timer Q (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	This manual
Manual for Using Sample Program Functions Timer ENC (V850E/IA3, V850E/IA4) Application Note	U18240E
Manual for Using Sample Program Functions Port Functions (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18241E
Manual for Using Sample Program Functions Clock Generator (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18242E
Manual for Using Sample Program Functions Standby Functions (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18243E
Manual for Using Sample Program Functions Interrupt Functions (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18244E
Manual for Using Sample Program Functions A/D Converters 0 and 1 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2) Application Note	U18245E
Manual for Using Sample Program Functions A/D Converter 2 (V850E/IA3, V850E/IA4) Application Note	U18246E

## CONTENTS

16-bit timer/event counter Q (TMQn) (n = 0, 1)	
Interval timer mode .....	9
16-bit timer/event counter Q (TMQ0)	
External event count mode .....	12
16-bit timer/event counter Q (TMQ0)	
External trigger pulse output mode .....	15
16-bit timer/event counter Q (TMQ0)	
One-shot pulse output mode.....	19
16-bit timer/event counter Q (TMQ0)	
PWM output mode .....	23
16-bit timer/event counter Q (TMQn) (n = 0, 1)	
Free-running timer mode .....	27
16-bit timer/event counter Q (TMQ0)	
Pulse width measurement mode.....	32

## 16-bit timer/event counter Q (TMQn) (n = 0, 1)

### Interval timer mode

[Functions]	<p>Outputs a PWM waveform with a duty factor of 50% whose half cycle is equal to the interval set by the TQ0CCR0 register.</p> <p>Outputs a PWM waveform, from the TOQ01 pin, whose half cycle is equal to the interval with a duty factor of 50% upon a match between the count values of the TQ0CCR1 register and the 16-bit counter.</p> <p>Can be implemented with TMQ0 and TMQ1.</p>
[Function name]	timerq_interval
[Argument]	None
[Processing content]	Performs count operation of an f <sub>xx</sub> /32 count clock, generates an interrupt by inverting the TOQ00 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, and clears the counter. Generates an interrupt by inverting the TOQ01 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR1 register. The TOQ00 and TOQ01 pins start output at high level.
[Starting method]	Starts by calling timerq_interval_st function.
[SFRs used]	<p>TQ0CTL0     Selects the count clock.</p> <p>TQ0CTL1     Selects the timer mode.</p> <p>TQ0IOC0     • Sets the TOQ00 to TOQ03 register pin outputs.               • Sets the TOQ00 to TOQ03 pin output levels.</p> <p>TQ0CCR0     Compare register of the 16-bit counter</p> <p>TQ0CCR1     Compare register of the 16-bit counter</p>
[call function]	main    main function
[Variable]	None
[Interrupts]	timerq_TQ0CC0_int timerq_TQ0CC1_int
[Interrupt sources]	INTTQ0CC0 INTTQ0CC1
[File name]	timerq_interval\timerq_1.c, timerq_interval\MAIN.C
[Caution]	<ul style="list-style-type: none"><li>• The following care must be exercised for setting the registers.       TQnCTL1           Bits 5 and 6: Only TMQ0 is settable. TMQ1 must be set to 0.</li></ul>

The interval time can be calculated by the following formula.

$$\text{Interval} = (\text{Set value of TQ0CCR0 register} + 1) \times \text{Count clock cycle}$$

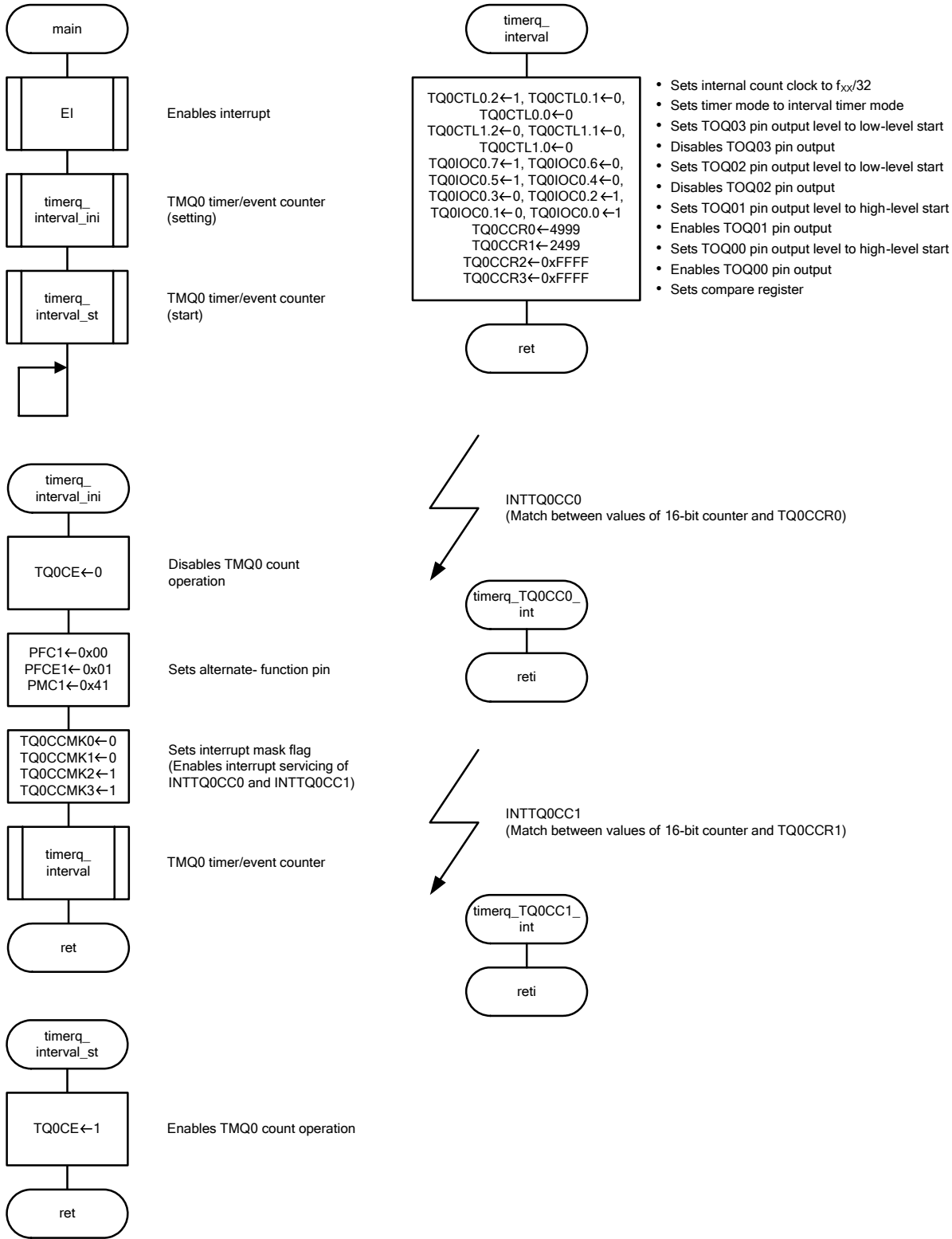
[Function name]	timerq_interval_st
[Argument]	None
[Processing content]	Starting function of timerq_interval
[Starting method]	Call this function after calling the timerq_interval function.
[SFR used]	TQ0CTL0.TQ0CE                      Controls TMQ0 operation.
[call function]	None
[Variable]	None
[File name]	timerq_interval\timerq_1.c
[Caution]	None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0              Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_interval\timerq_1.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1              Match between the count value of the 16-bit counter and TQ0CCR1
[call function]	None
[Variable]	None
[File name]	timerq_interval\timerq_1.c
[Caution]	None

16-bit timer/event counter Q (TMQn)  
Interval timer mode



## 16-bit timer/event counter Q (TMQ0)

### External event count mode

[Functions]	<p>Counts the valid edge of the external event count input (EVTQ0 pin) and generates an interrupt request signal (INTTQ0CC0) for each count set to the TQ0CCR0 register. (Clears the 16-bit counter simultaneously.)</p> <p>Generates an interrupt request signal (INTTQ0CCR1) upon a compare match between the count value of the 16-bit counter and the value of the TQ0CCR1 register.</p> <p>Can be implemented with TMQ0.</p>
[Function name]	timerq_event_count
[Argument]	None
[Processing content]	<p>Counts the valid edge of the external event count input, generates an interrupt upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, and clears the counter.</p> <p>Generates an interrupt upon the count subsequent to the count whose value matches the value of the TQ0CCR1 register.</p>
[Starting method]	Starts by calling the timerq_event_count_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1 Selects the timer mode.</p> <p>TQ0IOC2</p> <ul style="list-style-type: none"><li>• Sets the valid edge of the external event count input signal (EVTQ0 pin).</li><li>• Sets the valid edge of the external trigger input signal (TRGQ0 pin).</li></ul> <p>TQ0CCR0 Compare register of the 16-bit counter</p> <p>TQ0CCR1 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	timerq_TQ0CC0_int timerq_TQ0CC1_int
[Interrupt sources]	INTTQ0CC0 INTTQ0CC1
[File name]	timerq_event_count\timerq_2.c, timerq_event_count\MAIN.C
[Cautions]	<ul style="list-style-type: none"><li>• The TQ0CCR0 to TQ0CCR3 registers must not be set to 0x0000.</li><li>• The following care must be exercised for setting the registers. TQ0IOC0 Set to 0x00.</li></ul>

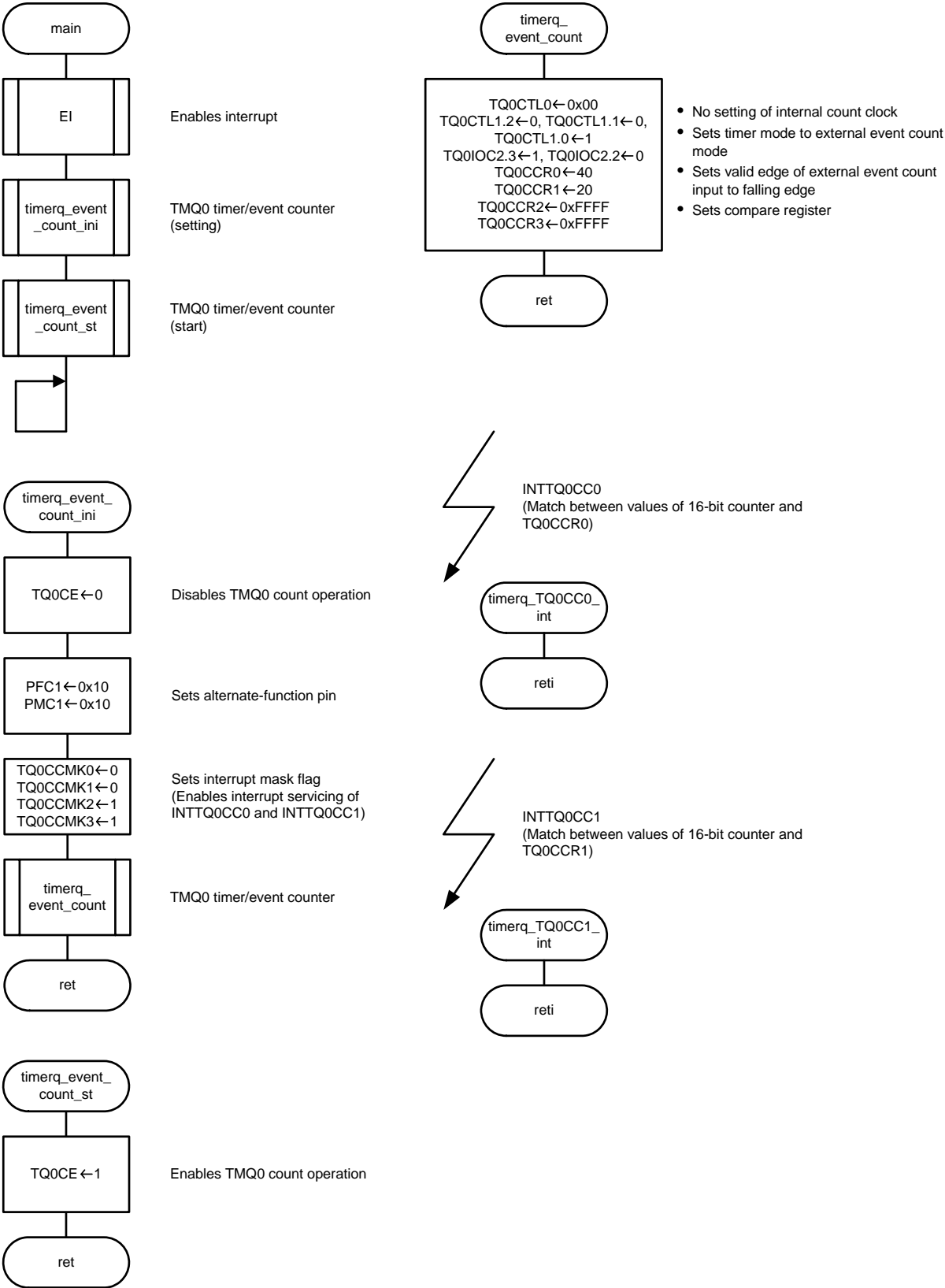
[Function name]	timerq_event_count_st
[Argument]	None
[Processing content]	Starting function of timerq_event_count
[Starting method]	Call this function after calling the timerq_event_count function.
[SFR used]	TQ0CTL0.TQ0CE                      Controls TMQ0 operation.
[call function]	None
[Variable]	None
[File name]	timerq_event_count\timerq_2.c
[Caution]	None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0              Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_event_count\timerq_2.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1              Match between the count value of the 16-bit counter and TQ0CCR1
[call function]	None
[Variable]	None
[File name]	timerq_event_count\timerq_2.c
[Caution]	None

16-bit timer/event counter Q (TMQ0)  
External event count mode





# 16-bit timer/event counter Q (TMQ0)

## External trigger pulse output mode

(1/2)

[Functions]	<p>Starts count operation of the valid edge of the external event count input (EVTQ0 pin) when the valid edge of the external trigger input (TRGQ0 pin) is detected.</p> <p>Inverts the TOQ00 pin output by clearing the count value of the 16-bit counter upon a compare match with the TQ0CCR0 register.</p> <p>Inverts the TOQ01 pin output upon a match between the value set by the TQ0CCR1 register and the count value of the 16-bit counter.</p> <p>Inverts the TOQ01 pin output after the 16-bit counter is cleared.</p> <p>Can be implemented with TMQ0.</p>
[Function name]	timerq_trigger_pulse
[Argument]	None
[Processing content]	<p>Starts count operation of the valid edge of the external event count input when the valid edge of the external trigger input is detected, generates an interrupt by inverting the TOQ00 and TOQ01 pin outputs upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, and clears the counter.</p> <p>Generates an interrupt by inverting the TOQ01 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR1 register.</p> <p>TOQ00 and TOQ01 pins starts output at high level.</p>
[Starting method]	Starts by calling the timerq_trigger_pulse_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1 Selects the timer mode.</p> <p>TQ0IOC0</p> <ul style="list-style-type: none"> <li>• Sets the TOQ00 and TOQ03 pin outputs.</li> <li>• Sets the TOQ00 and TOQ03 pin output levels.</li> </ul> <p>TQ0IOC2</p> <ul style="list-style-type: none"> <li>• Sets the valid edge of the external event count input signal (EVTQ0 pin).</li> <li>• Sets the valid edge of the external trigger input signal (TRGQ0 pin).</li> </ul> <p>TQ0CCR0 Compare register of the 16-bit counter</p> <p>TQ0CCR1 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	<p>timerq_TQ0CC0_int</p> <p>timerq_TQ0CC1_int</p>
[Interrupt sources]	<p>INTTQ0CC0</p> <p>INTTQ0CC1</p>

[File name]	timerq_trigger_pulse\timerq_3.c, timerq_trigger_pulse\MAIN.C
[Caution]	<ul style="list-style-type: none"> <li>The compare register is written in batch write mode.</li> </ul> <p>When changing the value of the compare register during timer operation, change the value of the TQ0CCR1 register at the end.</p>

The active level width, cycle and duty factor of the PWM waveform output from the TOQ01 pin can be calculated by the following formula.

Active level width = (Set value of TQ0CCR1 register) × Count clock cycle
Cycle = (Set value of TQ0CCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TQ0CCR1 register)/(Set value of TQ0CCR0 register + 1)

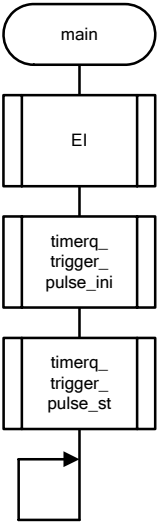
[Function name]	timerq_trigger_pulse_st
[Argument]	None
[Processing content]	Starting function of timerq_trigger_pulse
[Starting method]	Call this function after calling the timerq_trigger_pulse function.
[SFR used]	TQ0CTL0.TQ0CE                      Controls TMQ0 operation.
[call function]	None
[Variable]	None
[File name]	timerq_trigger_pulse\timerq_3.c
[Caution]	None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0      Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_trigger_pulse\timerq_3.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1      Match between the count value of the 16-bit counter and TQ0CCR1
[call function]	None
[Variable]	None
[File name]	timerq_trigger_pulse\timerq_3.c
[Caution]	None

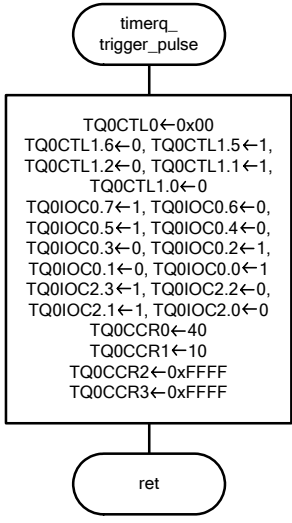
16-bit timer/event counter Q (TMQ0)  
External trigger pulse output mode



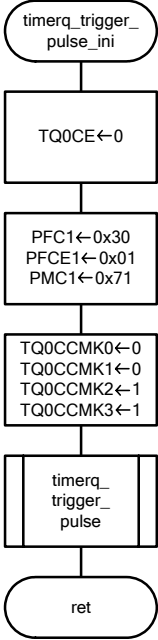
Enables interrupt

TMQ0 timer/event counter (setting)

TMQ0 timer/event counter (start)



- No setting of internal count clock
- Sets software trigger control as unused
- Enables external event count input operation
- Sets timer mode to external trigger pulse output mode
- Sets TOQ03 pin output level to low-level start
- Disables TOQ03 pin output
- Sets TOQ02 pin output level to low-level start
- Disables TOQ02 pin output
- Sets TOQ01 pin output level to high-level start
- Enables TOQ01 pin output
- Sets TOQ00 pin output level to high-level start
- Enables TOQ00 pin output
- Sets valid edge of external event count input to falling edge
- Sets valid edge of external trigger input to falling edge
- Sets compare register

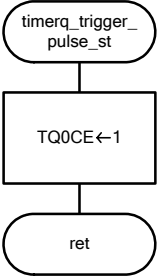
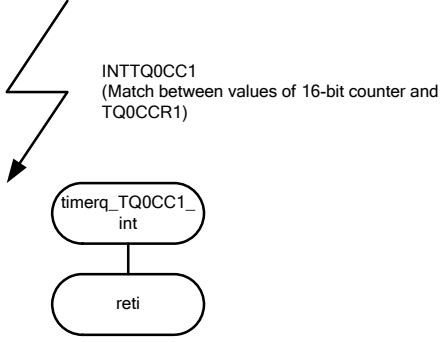
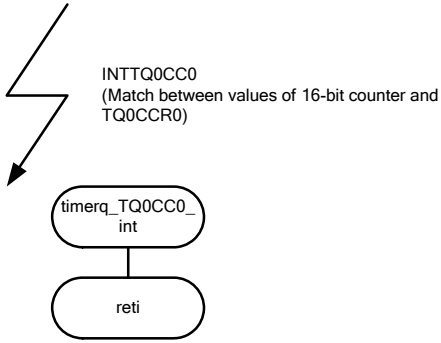


Disables TMQ0 count operation

Sets alternate-function pin

Sets interrupt mask flag (Enables interrupt servicing of INTTQ0CC0 and INTTQ0CC1)

TMQ0 timer/event counter



Enables TMQ0 count operation

## 16-bit timer/event counter Q (TMQ0) One-shot pulse output mode

[Functions]	<p>Starts count operation of the valid edge of the external event count input (EVTQ0 pin) when the valid edge of the external trigger input (TRGQ0 pin) is detected.</p> <p>Stops count operation of the 16-bit counter by inverting the TOQ00 pin output upon a match between the value set by the TQ0CCR0 register and the count value of the 16-bit counter.</p> <p>Inverts the TOQ01 pin output upon a match between the value set by the TQ0CCR1 register and the count value of the 16-bit counter.</p> <p>Can be implemented with TMQ0.</p>
[Function name]	timerq_1shot_pulse
[Argument]	None
[Processing content]	<p>Starts count operation of the valid edge of the external event count input upon valid edge detection of the external trigger input, generates an interrupt by inverting the TOQ00 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, clears the counter, and stops count operation.</p> <p>Generates an interrupt by inverting the TOQ01 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR1 register.</p> <p>TOQ00 and TOQ01 pins start at high level.</p>
[Starting method]	Starts by calling the timerq_1shot_pulse_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1 Selects the timer mode.</p> <p>TQ0IOC0</p> <ul style="list-style-type: none"> <li>• Sets the TOQ00 to TOQ03 pin outputs.</li> <li>• Sets the TOQ00 to TOQ03 pin output levels.</li> </ul> <p>TQ0IOC2</p> <ul style="list-style-type: none"> <li>• Sets the valid edge of the external event count input signal (EVTQ0 pin).</li> <li>• Sets the valid edge of the external trigger input signal (TRGQ0 pin).</li> </ul> <p>TQ0CCR0 Compare register of the 16-bit counter</p> <p>TQ0CCR1 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	<p>timerq_TQ0CC0_int</p> <p>timerq_TQ0CC1_int</p>
[Interrupt sources]	<p>INTTQ0CC0</p> <p>INTTQ0CC1</p>
[File name]	<p>timerq_1shot_pulse\timerq_4.c,</p> <p>timerq_1shot_pulse\MAIN.C</p>
[Caution]	None

The output delay period and the active level width of the one-shot pulse can be calculated from the following formula.

Output delay period = (Set value of TQ0CCR1 register) × Count clock cycle

Active level width = (Set value of TQ0CCR0 register – Set value of TQ0CCR1 register + 1) × Count clock cycle

[Function name] timerq\_1shot\_pulse\_st

[Argument] None

[Processing content] Starting function of timerq\_1shot\_pulse

[Starting method] Call this function after calling the timerq\_1shot\_pulse function.

[SFR used] TQ0CTL0.TQ0CE Controls TMQ0 operation.

[call function] None

[Variable] None

[File name] timerq\_1shot\_pulse\timerq\_4.c

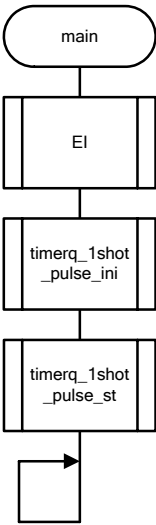
[Caution] None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0      Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_1shot_pulse\timerq_4.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1      Match between the count value of the 16-bit counter and TQ0CCR1
[call function]	None
[Variable]	None
[File name]	timerq_1shot_pulse\timerq_4.c
[Caution]	None

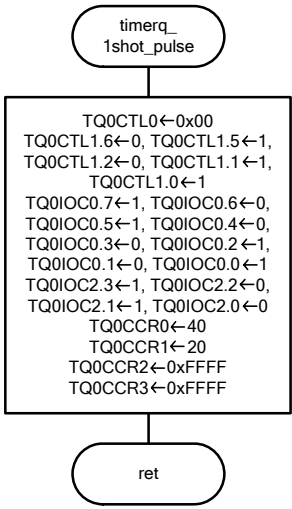
16-bit timer/event counter Q (TMQ0)  
One-shot pulse output mode



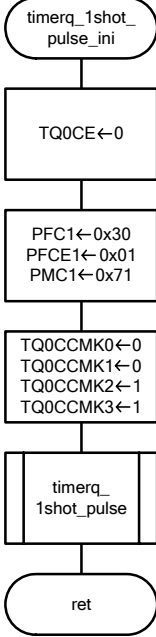
Enables interrupt

TMQ0 timer/event counter (setting)

TMQ0 timer/event counter (start)



- No setting of internal count clock
- Sets software trigger control as unused
- Enables external event count input operation
- Sets timer mode to one-shot pulse output mode
- Sets TOQ03 pin output level to low-level start
- Disables TOQ03 pin output
- Sets TOQ02 pin output level to low-level start
- Disables TOQ02 pin output
- Sets TOQ01 pin output level to high-level start
- Enables TOQ01 pin output
- Sets TOQ00 pin output level to high-level start
- Enables TOQ00 pin output
- Sets valid edge of external event count input to falling edge
- Sets valid edge of external trigger input to falling edge
- Sets compare register

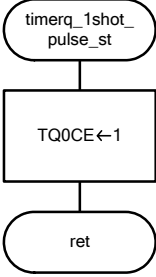
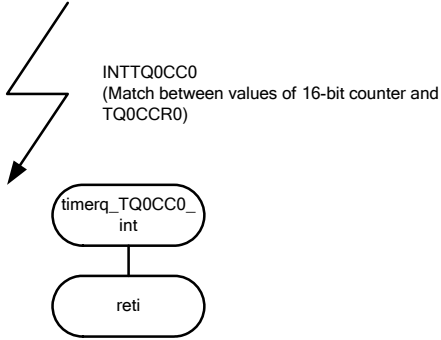


Disables TMQ0 count operation

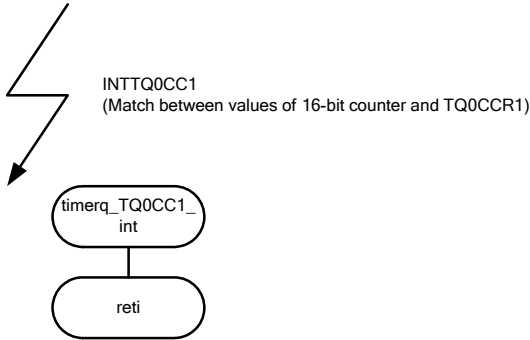
Sets alternate-function pin

Sets interrupt mask flag (Enables interrupt servicing of INTTQ0CC0 and INTTQ0CC1)

TMQ0 timer/event counter



Enables TMQ0 count operation





## 16-bit timer/event counter Q (TMQ0) PWM output mode

[Functions]	<p>Starts operation of the 16-bit counter by setting the TQ0CE bit.</p> <p>Outputs a PWM waveform with a 50% duty factor whose half cycle is equal to the set value of the TQ0CCR0 register + 1, by clearing the 16-bit counter upon a compare match with the TQ0CCR0 register and inverting the TOQ00 pin.</p> <p>Inverts the TOQ01 pin output upon a match between the value set by the TQ0CCR1 register and the count value of the 16-bit counter.</p> <p>Inverts the TOQ01 pin output when the 16-bit counter is cleared.</p> <p>Can be implemented with TMQ0.</p>
[Function name]	timerq_pwm_output
[Argument]	None
[Processing content]	<p>Performs count operation of an fxx/32 count clock, generates an interrupt by inverting the TOQ00 and TOQ01 pin outputs upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, and clears the counter.</p> <p>Generates an interrupt by inverting the TOQ01 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR1 register.</p> <p>The TOQ00 and TOQ01 pins start output at high level.</p>
[Starting method]	Starts by calling the timerq_pwm_output_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1</p> <ul style="list-style-type: none"> <li>• Selects the timer mode.</li> <li>• Selects the internal count clock or external event count input.</li> </ul> <p>TQ0IOC0</p> <ul style="list-style-type: none"> <li>• Sets the TOQ00 to TOQ03 pin outputs.</li> <li>• Sets the TOQ00 to TOQ03 pin output levels.</li> </ul> <p>TQ0IOC2</p> <ul style="list-style-type: none"> <li>• Sets the valid edge of the external event counter input signal (EVTQ0 pin).</li> </ul> <p>TQ0CCR0 Compare register of the 16-bit counter</p> <p>TQ0CCR1 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	<p>timerq_TQ0CC0_int</p> <p>timerq_TQ0CC1_int</p>
[Interrupt sources]	<p>INTTQ0CC0</p> <p>INTTQ0CC1</p>
[File name]	<p>timerq_pwm_output\timerq_5.c,</p> <p>timerq_pwm_output\MAIN.C</p>
[Caution]	<ul style="list-style-type: none"> <li>• The compare register is written in batch write mode.</li> </ul> <p>When changing the value of the compare register during timer operation, change the value of the TQ0CCR1 register at the end.</p>

The active level width, cycle, and duty factor of the PWM waveform output from the TOQ01 pin can be calculated from the following formula.

Active level width = (Set value of TQ0CCR1 register) × Count clock cycle

Cycle = (Set value of TQ0CCR0 register + 1) × Count clock cycle

Duty factor = (Set value of TQ0CCR1 register)/(Set value of TQ0CCR0 register + 1)

[Function name] timerq\_pwm\_output\_st

[Argument] None

[Processing content] Starting function of timerq\_pwm\_output

[Starting method] Call this function after calling the timerq\_pwm\_output function.

[SFR used] TQ0CTL0.TQ0CE Controls TMQ0 operation.

[call function] None

[Variable] None

[File name] timerq\_pwm\_output\timerq\_5.c

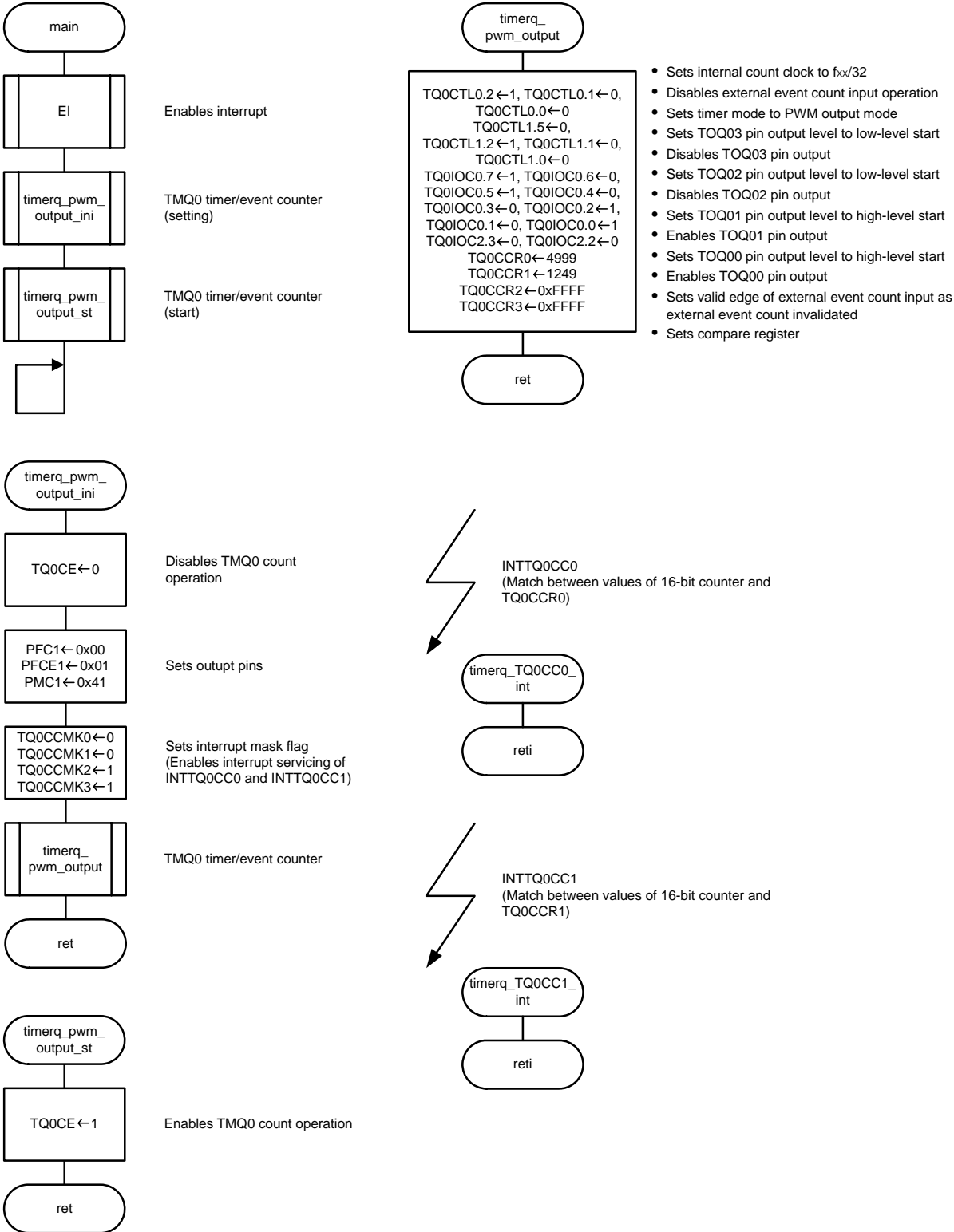
[Caution] None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0      Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_pwm_output\timerq_5.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1      Match between the count value of the 16-bit counter and TQ0CCR1
[call function]	None
[Variable]	None
[File name]	timerq_pwm_output\timerq_5.c
[Caution]	None

16-bit timer/event counter Q (TMQ0)  
 PWM output mode



## 16-bit timer/event counter Q (TMQn) (n = 0, 1)

### Free-running timer mode

(1/2)

[Functions]	<p>Inverts the TOQ00 pin output upon a compare match between the TQ0CCR0 register and the count value of the 16-bit counter (compare function).</p> <p>Stores the count value of the 16-bit counter when the valid edge of the capture trigger input (TIQ01 pin) is detected (capture function).</p> <p>The compare function can be implemented with TMQ0 and TMQ1.</p> <p>The capture function can be implemented with TMQ0.</p>
[Function name]	timerq_free_running
[Argument]	None
[Processing content]	<p>Performs count operation of an f<sub>xx</sub>/32 count clock, generates an interrupt by inverting the TOQ00 pin output upon the count subsequent to the count whose value matches the value of the TQ0CCR0 register, and clears the counter.</p> <p>Generates an interrupt by capturing the count value to the TQ0CCR1 register when the valid edge from the TIQ01 pin is detected.</p> <p>Generates an INTTQ0OV interrupt when a counter overflow is detected.</p> <p>The TOQ00 pin starts output at high level.</p>
[Starting method]	Starts by calling the timerq_free_running_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1 • Selects the timer mode.</p> <ul style="list-style-type: none"> <li>• Selects the internal count clock or external event count input.</li> </ul> <p>TQ0IOC0 • Sets the TOQ00 to TOQ03 pin outputs.</p> <ul style="list-style-type: none"> <li>• Sets the TOQ00 to TOQ03 pin output levels.</li> </ul> <p>TQ0IOC1 Sets the valid edge of the capture trigger input signal (TIQ00 to TIQ03 pins)</p> <p>TQ0IOC2 • Sets the valid edge of the external event count input signal (EVTQ0 pin).</p> <ul style="list-style-type: none"> <li>• Sets the valid edge of the external trigger input signal (TRGQ0 pin).</li> </ul> <p>TQ0OPT0 Selects the capture or compare function of the TQ0CCR0 to TQ0CCR3 registers.</p> <p>TQ0CCR0 Compare register of the 16-bit counter</p> <p>TQ0CCR1 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	<p>timerq_TQ0CC0_int</p> <p>timerq_TQ0CC1_int</p> <p>timerq_TQ0OV_int</p>
[Interrupt sources]	<p>INTTQ0CC0</p> <p>INTTQ0CC1</p> <p>INTTQ0OV</p>

[File name]	timerq_free_running\timerq_6.c, timerq_free_running\MAIN.C
[Caution]	<ul style="list-style-type: none"> <li>The following care must be exercised for setting the registers.  TQnCTL1  Bits 5 and 6: Only TMQ0 is valid. TMQ1 must be set to 0.  TQnOPT0  Bits 7 to 4: Only TMQ0 is valid. TMP1 must be set to 0.</li> </ul>

[Function name]	timerq_free_running_st
[Argument]	None
[Processing content]	Starting function of timerq_free_running
[Starting method]	Call this function after calling the timerq_free_running function.
[SFR used]	TQ0CTL0.TQ0CE Controls TMQ0 operation.
[call function]	None
[Variable]	None
[File name]	timerq_free_running\timerq_6.c
[Caution]	None

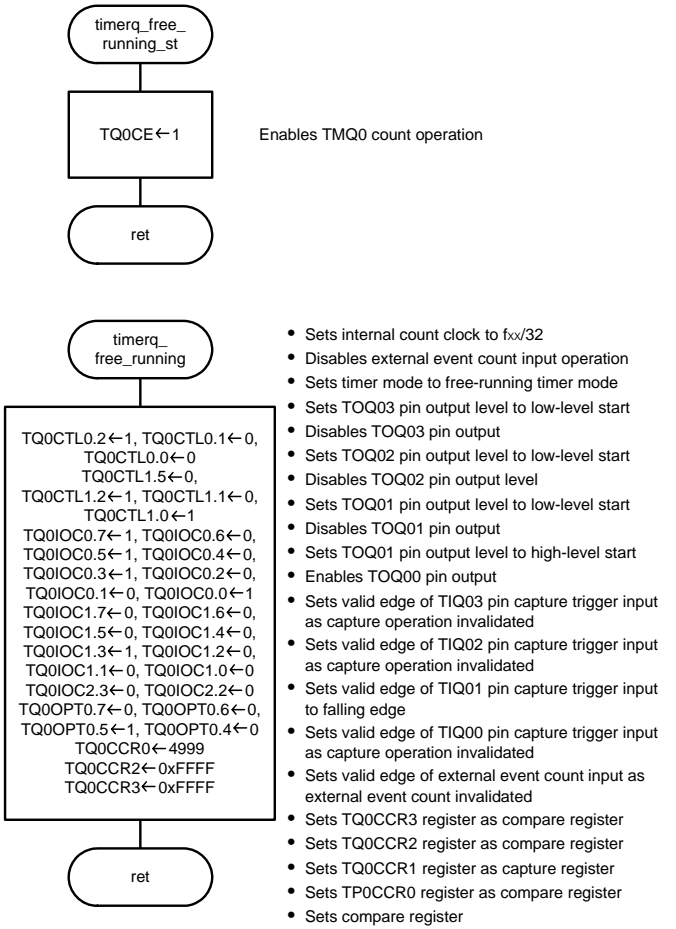
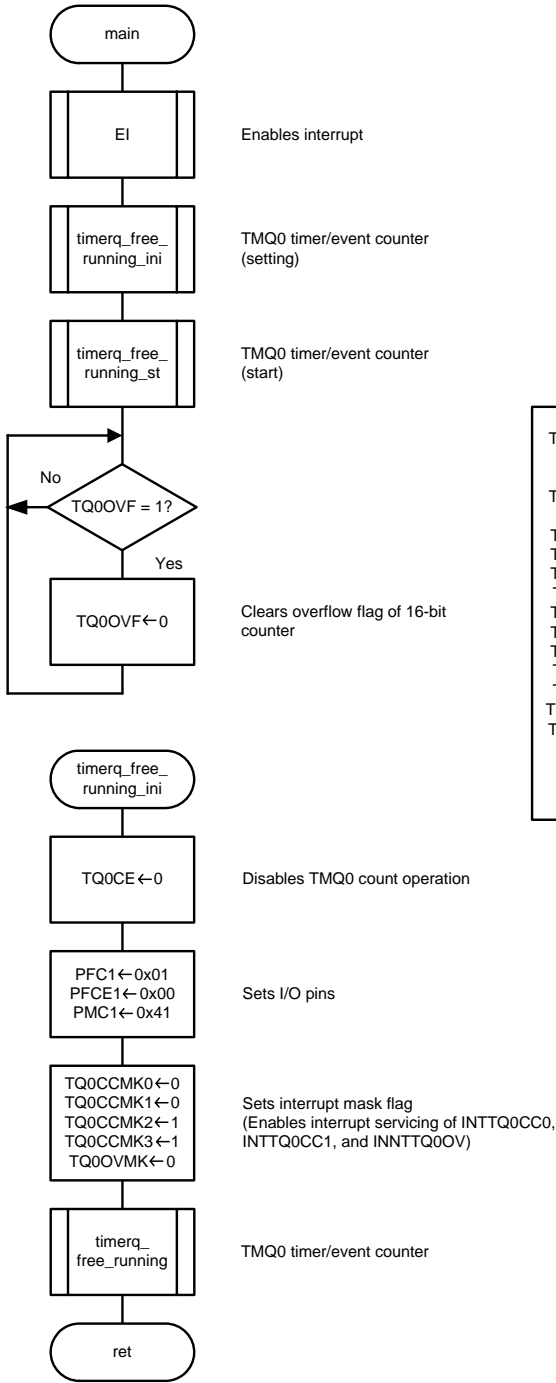
## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0      Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_free_running\timerq_6.c
[Caution]	None

[Function name]	timerq_TQ0CC1_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC1      Valid edge detection of TIQ01 pin input.
[call function]	None
[Variable]	None
[File name]	timerq_free_running\timerq_6.c
[Caution]	None

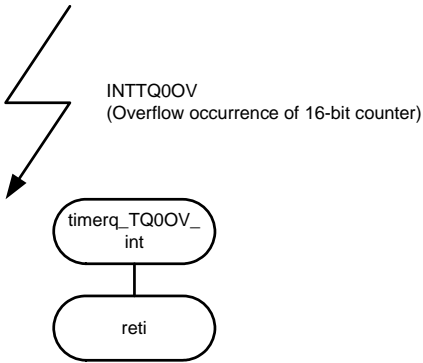
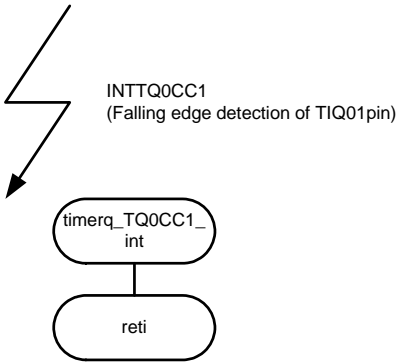
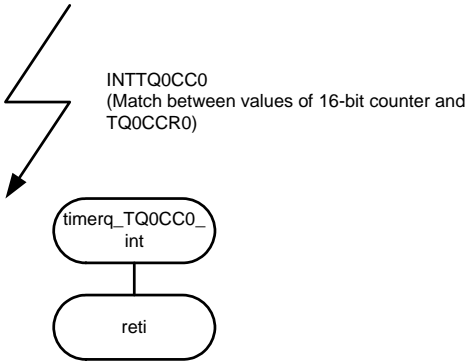
[Function name]	timerq_TQ0OV_int
[Overview]	Defined by the user.
[Factor]	INTTQ0OV      Overflow occurrence of 16-bit counter
[call function]	None
[Variable]	None
[File name]	timerq_free_running\timerq_6.c
[Caution]	None

16-bit timer/event counter Q (TMQn)  
Free-running timer mode (1/2)





16-bit timer/event counter Q (TMQn)  
Free-running timer mode (2/2)



## 16-bit timer/event counter Q (TMQ0)

### Pulse width measurement mode

[Functions]	<p>Clears the 16-bit counter by storing the count value to the TQ0CCR0 register when the valid edge of the capture trigger input (TIQ00 pin) is detected.</p> <p>Measures the valid edge interval of the TIQ00 pin by generating an interrupt when the valid edged of the TIQ00 pin input is detected and reading the TQ0CCR0 register value.</p> <p>Can be implemented with TMQ0.</p>
[Function name]	timerq_pulse_measure
[Argument]	None
[Processing content]	<p>Performs count operation of an fxx/32 count clock, generates an interrupt by storing the count value of the 16-bit counter to the TQ0CCR0 register when the valid edge of the TIQ00 pin input is detected, and clears the counter.</p> <p>Generates an INTTQ0OV interrupt when a counter overflow is detected.</p>
[Starting method]	Starts by calling the timerq_pulse_measure_st function.
[SFRs used]	<p>TQ0CTL0 Selects the count clock.</p> <p>TQ0CTL1</p> <ul style="list-style-type: none"><li>• Selects the timer mode.</li><li>• Selects the internal count clock or external event count input.</li></ul> <p>TQ0IOC1 Sets the valid edge of the capture trigger input signal (TIQ00 to TIQ03 pins).</p> <p>TQ0IOC2</p> <ul style="list-style-type: none"><li>• Sets the valid edge of the external event count input signal (EVTQ0 pin).</li><li>• Sets the valid edge of the external trigger input signal (TRGQ0 pin).</li></ul> <p>TQ0OPT0 Selects the capture or compare function of the TQ0CCR0 to TQ0CCR3 registers.</p> <p>TQ0CCR0 Compare register of the 16-bit counter</p>
[call function]	main main function
[Variable]	None
[Interrupts]	timerq_TQ0CC0_int timerq_TQ0OV_int
[Interrupt sources]	INTTQ0CC0 INTTQ0OV
[File name]	timerq_pulse_measure\timerq_7.c, timerq_pulse_measure\MAIN.C
[Cautions]	<ul style="list-style-type: none"><li>• If a slow clock is selected as the count clock and a capture trigger is input immediately after the TQ0CTL0.TQ0CE bit has been set, 0xFFFF may be captured instead of 0x0000 to the TQ0CCR0 register.</li><li>• The following care must be exercised for setting the registers. TQ0IOC0 Set to 0x00.</li></ul>

The pulse width can be calculated by the following formula.

$$\text{Pulse width} = (\text{Value of TQ0CCR0 register} + 1) \times \text{Count clock cycle}$$

The pulse width if an overflow of the 16-bit counter is detected can be calculated by the following formula.

$$\text{Pulse width} = (\text{Value of TQ0CCR0 register} + 0x10001) \times \text{Count clock cycle}$$

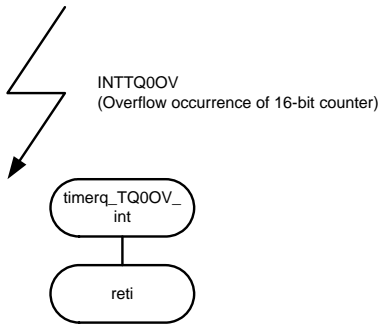
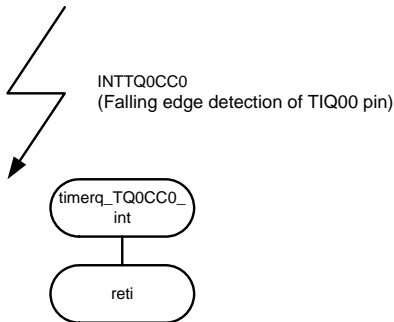
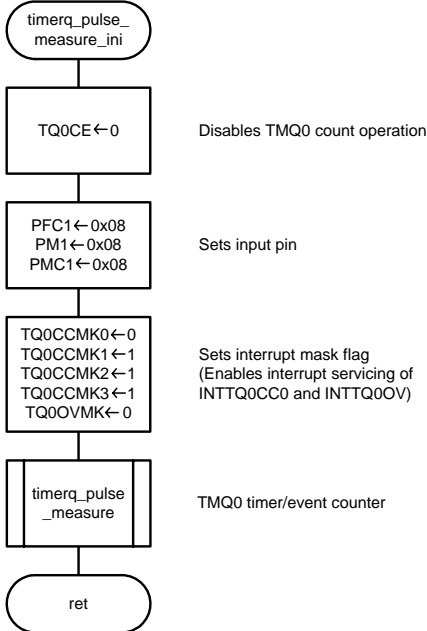
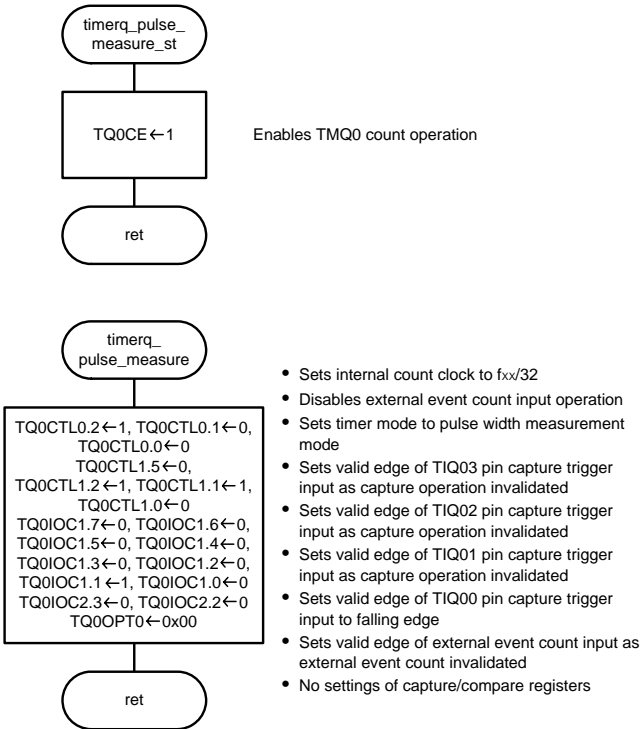
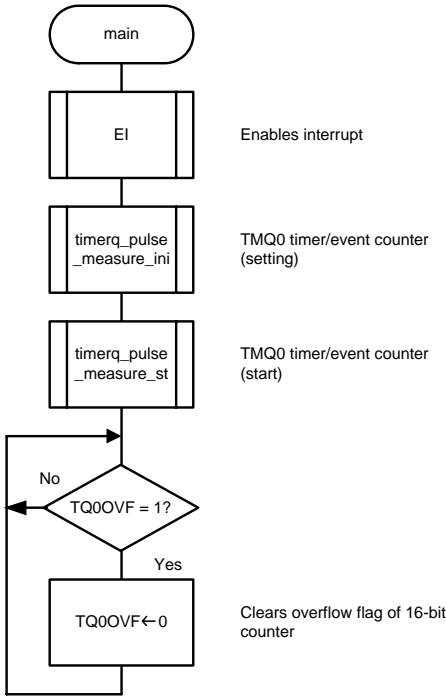
[Function name]	timerq_pulse_measure_st
[Argument]	None
[Processing content]	Starting function of timerq_pulse_measure
[Starting method]	Call this function after calling the timerq_pulse_measure function.
[SFR used]	TQ0CTL0.TQ0CE Controls TMQ0 operation.
[call function]	None
[Variable]	None
[File name]	timerq_pulse_measure\timerq_7.c
[Caution]	None

## Interrupt functions

[Function name]	timerq_TQ0CC0_int
[Overview]	Defined by the user.
[Factor]	INTTQ0CC0      Match between the count value of the 16-bit counter and TQ0CCR0
[call function]	None
[Variable]	None
[File name]	timerq_pulse_measure\timerq_7.c
[Caution]	None

[Function name]	timerq_TQ0OV_int
[Overview]	Defined by the user.
[Factor]	INTTQ0OV      Overflow occurrence of 16-bit counter
[call function]	None
[Variable]	None
[File name]	timerq_pulse_measure\timerq_7.c
[Caution]	None

16-bit timer/event counter Q (TMQ0)  
Pulse width measurement mode



*For further information,  
please contact:*

**NEC Electronics Corporation**

1753, Shimonumabe, Nakahara-ku,  
Kawasaki, Kanagawa 211-8668,  
Japan  
Tel: 044-435-5111  
<http://www.necel.com/>

**[America]**

**NEC Electronics America, Inc.**

2880 Scott Blvd.  
Santa Clara, CA 95050-2554, U.S.A.  
Tel: 408-588-6000  
800-366-9782  
<http://www.am.necel.com/>

**[Europe]**

**NEC Electronics (Europe) GmbH**

Arcadiastrasse 10  
40472 Düsseldorf, Germany  
Tel: 0211-65030  
<http://www.eu.necel.com/>

**Hanover Office**

Podbielskistrasse 166 B  
30177 Hannover  
Tel: 0 511 33 40 2-0

**Munich Office**

Werner-Eckert-Strasse 9  
81829 München  
Tel: 0 89 92 10 03-0

**Stuttgart Office**

Industriestrasse 3  
70565 Stuttgart  
Tel: 0 711 99 01 0-0

**United Kingdom Branch**

Cygnus House, Sunrise Parkway  
Linford Wood, Milton Keynes  
MK14 6NP, U.K.  
Tel: 01908-691-133

**Succursale Française**

9, rue Paul Dautier, B.P. 52  
78142 Velizy-Villacoublay Cédex  
France  
Tel: 01-3067-5800

**Sucursal en España**

Juan Esplandiú, 15  
28007 Madrid, Spain  
Tel: 091-504-2787

**Tyskland Filial**

Täby Centrum  
Entrance S (7th floor)  
18322 Täby, Sweden  
Tel: 08 638 72 00

**Filiale Italiana**

Via Fabio Filzi, 25/A  
20124 Milano, Italy  
Tel: 02-667541

**Branch The Netherlands**

Steijgerweg 6  
5616 HS Eindhoven  
The Netherlands  
Tel: 040 265 40 10

**[Asia & Oceania]**

**NEC Electronics (China) Co., Ltd**

7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian  
District, Beijing 100083, P.R.China  
Tel: 010-8235-1155  
<http://www.cn.necel.com/>

**NEC Electronics Shanghai Ltd.**

Room 2511-2512, Bank of China Tower,  
200 Yincheng Road Central,  
Pudong New Area, Shanghai P.R. China P.C.:200120  
Tel: 021-5888-5400  
<http://www.cn.necel.com/>

**NEC Electronics Hong Kong Ltd.**

12/F., Cityplaza 4,  
12 Taikoo Wan Road, Hong Kong  
Tel: 2886-9318  
<http://www.hk.necel.com/>

**NEC Electronics Taiwan Ltd.**

7F, No. 363 Fu Shing North Road  
Taipei, Taiwan, R. O. C.  
Tel: 02-8175-9600  
<http://www.tw.necel.com/>

**NEC Electronics Singapore Pte. Ltd.**

238A Thomson Road,  
#12-08 Novena Square,  
Singapore 307684  
Tel: 6253-8311  
<http://www.sg.necel.com/>

**NEC Electronics Korea Ltd.**

11F., Samik Lavied'or Bldg., 720-2,  
Yeoksam-Dong, Kangnam-Ku,  
Seoul, 135-080, Korea  
Tel: 02-558-3737  
<http://www.kr.necel.com/>

G06.11A