M4K Group (1)
Application Note
32-bit Timer Event Counter
(T32A-B)
Interval Timer Function

Outlines

This application note is a reference material for developing products using the interval timer function of the 32-bit timer event counter (T32A) of M4K Group (1).
This document helps the user check operation of the product and develop its program.

Target sample program: Timer_LED

Table of Contents

1. Preface

This sample program blinks an LED with a constant interval using the Timer A.
Push switches are used to change the stop and the re-start of the blink of the LED.

2. Reference Document

1. Datasheet
   TMPM4K Group (1) datasheet Rev2.0 (Japanese edition)
2. Reference manual
   32-bit Timer Event Counter (T32A-B) Rev3.0 (Japanese edition)
   Input/Output Ports (PORT-M4K(1)) Rev2.0 (Japanese edition)
   Asynchronous Serial Communication Circuit (UART-C) Rev3.0 (Japanese edition)
3. Application note
   M4K Group (1) Application Note Startup (CMSIS System & Clock Configuration) Rev1.0
4. Other reference document
   TMPM4KxA Group Peripheral Driver User Manual (Doxygen) V1.0.4.0

3. Function to Use

IP	Channel	Port	Function/Operation mode
32-bit Timer Event Counter	ch0	-	Interval timer
Input/Output Ports	-	PE2 (Input Port) PE3 (Input Port)	Input
	-	PJ0 (Output Port) PJ2 (Output Port) PJ4 (Output Port) PL4 (Output Port)	Output
Asynchronous Serial Communication Circuit	ch0	PK0 (UT0RXD) PK1 (UT0TXDA)	UART mode

4. Target Device

The target devices of this application note are as follows;

TMPM4K4FYAUG	TMPM4K4FWAUG	TMPM4K4FUAUG	TMPM4K4FSAUG
TMPM4K4FYAFG	TMPM4K4FWAFG	TMPM4K4FUAFG	TMPM4K4FSAFG
TMPM4K2FYADUG	TMPM4K2FWADUG	TMPM4K2FUADUG	TMPM4K2FSADUG
TMPM4K1FYAUG	TMPM4K1FWAUG	TMPM4K1FUAUG	TMPM4K1FSAUG
			TMPM4K0FSADUG

* This sample program operates on the evaluation board of TMPM4K4FYAUG.
If other function than the TMPM4K4 one is checked, it is necessary that CMSIS Core related files (the startup file and I/O header file) should be changed properly.
Additionally, the name of microcontroller which is set to the project should be changed.
The BSP related file is dedicated to the evaluation board (TMPM4K4FYAUG). If other function than the TMPM4K4 one is checked, the BSP related file should be changed properly.

5. Operation Confirmation Condition

Used microcontroller	TMPM4K4FYAUG
Used board	TMPM4K4 evaluation board (Product of ESP-kikaku Co. Ltd.)
Integrated development environment	IAR Embedded Workbench for ARM 8.22.2
Integrated development environment	Arm® Keil® MDK Version 5.24.2.0
Terminal software	Tera Term V4.96
Sample program	v1.0.0

6. Evaluation Board Operation

Board function	Microcontroller pin name	Function
Push switch (SW1)	PE2	Port function stop
Push switch (SW2)	PE3	Port function re-start (Blink operation)

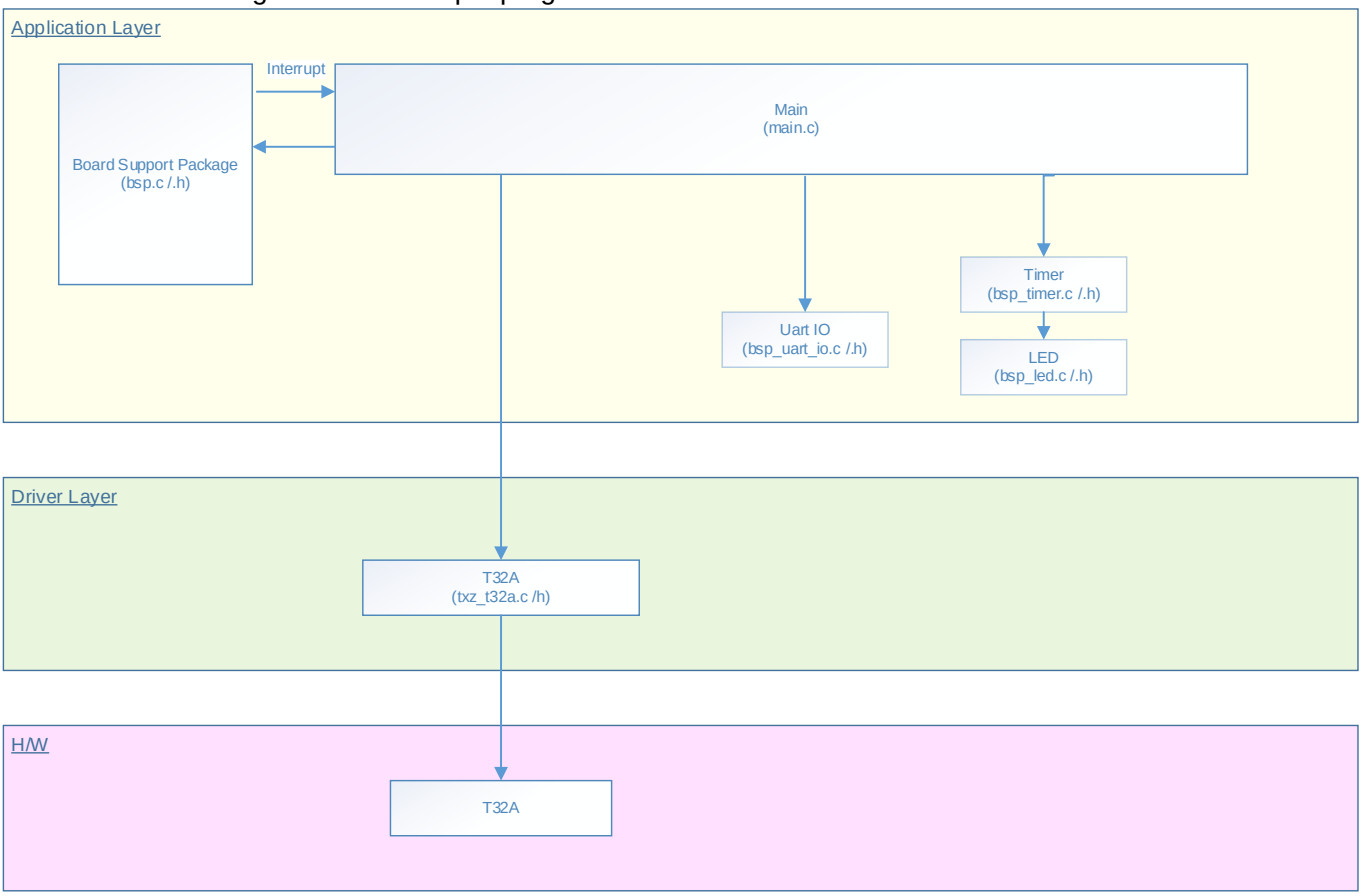
Corresponding port	LED on the board
PJ0	LED1
PJ2	LED2
PJ4	LED3
PL4	Unconnected.

The change of the operation is done by the push switch.
A PC should be connected with a USB_UART connector to communicate with the terminal software.
When the sample program is started up, the timer start-up interval is displayed on the terminal software (Tera Term).
When the push switch (SW1) is pushed down, the LED stops blinking. When the push switch (SW2) is pushed down, the LED re-starts to blink.

7. Sample Program

7.1. Structure Diagram of Sample Program

The structure diagram of the sample program is shown below.



7.2. Startup Routine

The following initialization is done after power is supplied.
The initialization of each clock setting and the initialization of the watchdog timer setting are done.

7.3. Main Operation

The initialization of the BSP should be done.
The initialization of the variables should be done.
The timer interval should be set as the initialization of the Timer driver.
The initialization of the USB_UART, the initialization of the LED's, and the initialization of the push switches should be done as the initialization of the application software.

The value of the timer interval is output to the terminal software.
The Timer is started.
Whenever the timer interval elapses, the LED lights and lights out alternatively.

When the push switch (SW1) is pushed down, the Timer stops and the status of the LED remains.
When the push switch (SW2) is pushed down, the Timer re-starts and the LED blinks.

7.4. Change of Timer Setting

The timer setting time can be changed by either of the followings;

- Change of the time of the Timer
- Change of the count-up value

Change of the time of the Timer
”static TXZ_Result driver_initialize (void)” in the “main” function should be used.

```
p_timer->init.interval = 1000
```

The setting value above is 1 ms.
When “1000 (1 μs*1000)” is modified, the blink time is changed.

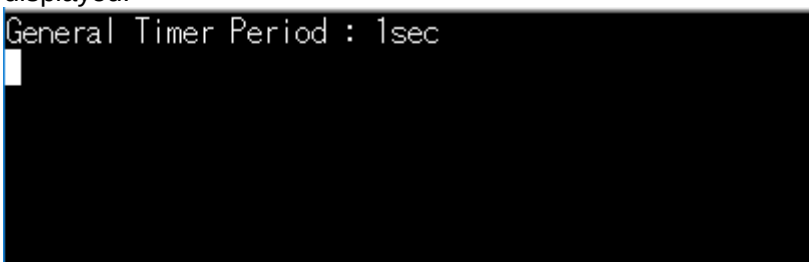
Change of the count-up value
”static void timer_interval_handler (uint32_t id)” in the “main” function should be used.

```
if (count < 1000) {     /* 1ms * 1000 = 1sec LED on */
    count++;
    on = 1;
}else if ( (count >= 1000) && (count < 2000) ) {    /* 1ms * 1000 = 1sec LED off */
    count++;
    on = 0;
```

The blink time can be changed by modifying the count value above.

7.5. Output Example of Terminal Software

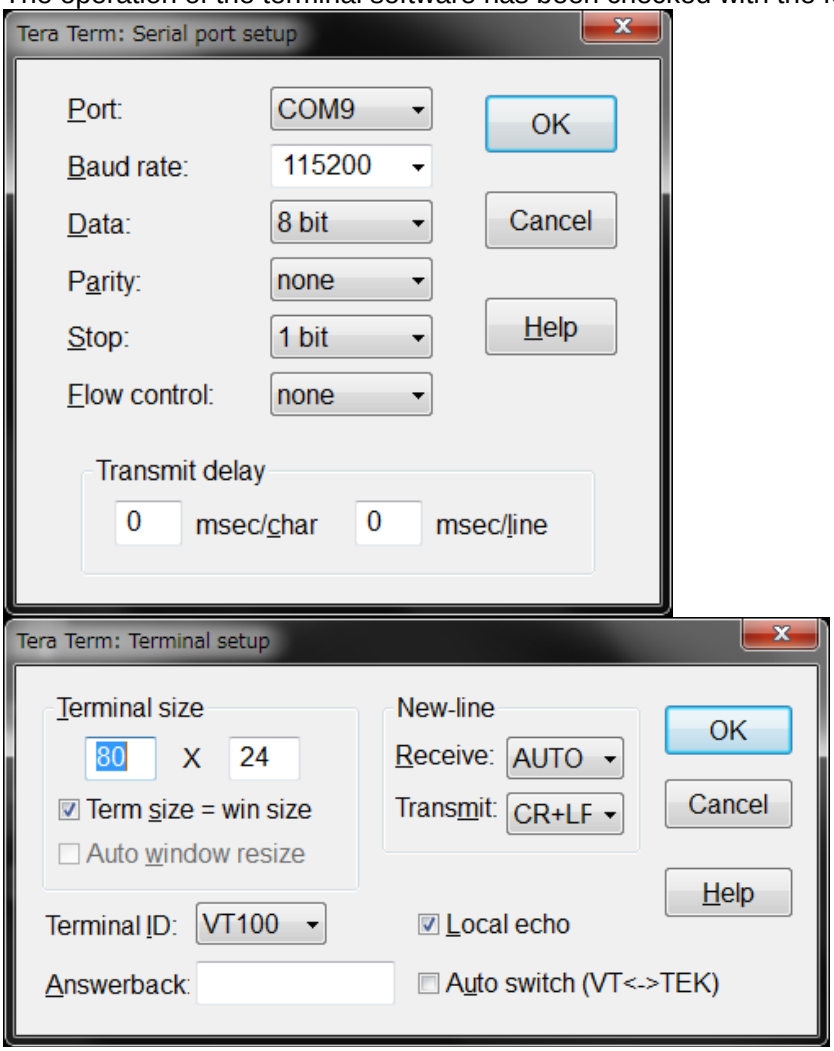
After the sample program is executed, the lighting interval and the definition of the push switch are displayed.



The display above is defined by “printf”.
If “General Timer Period” is changed, the setting time is different from the displayed result.

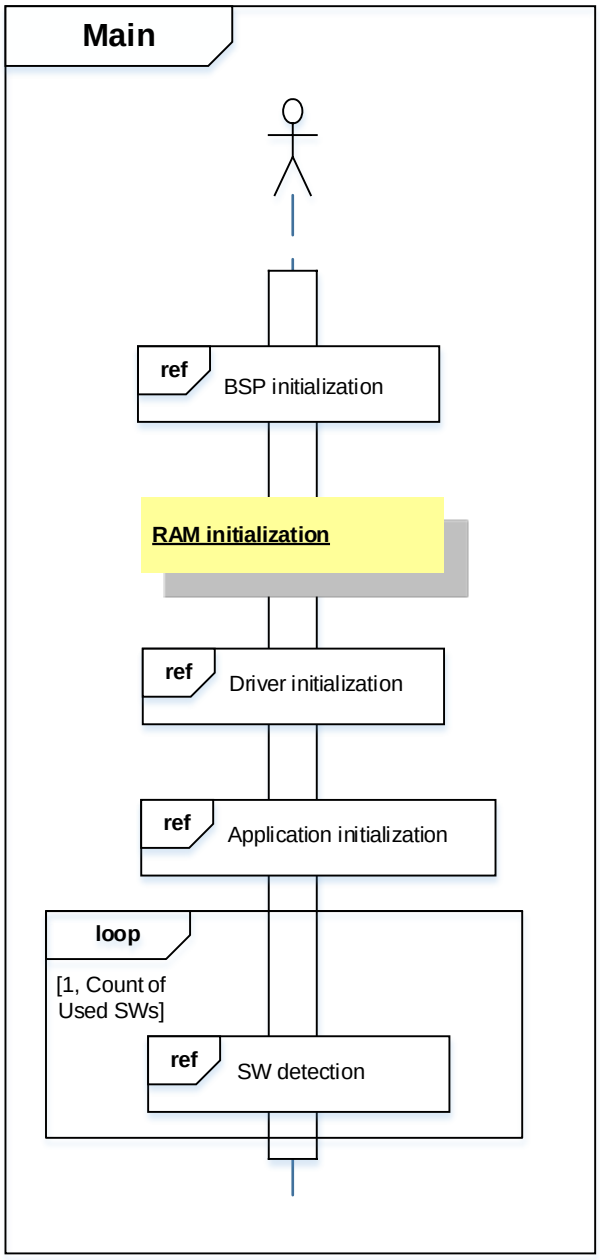
7.5.1. Setting Example of Terminal Software

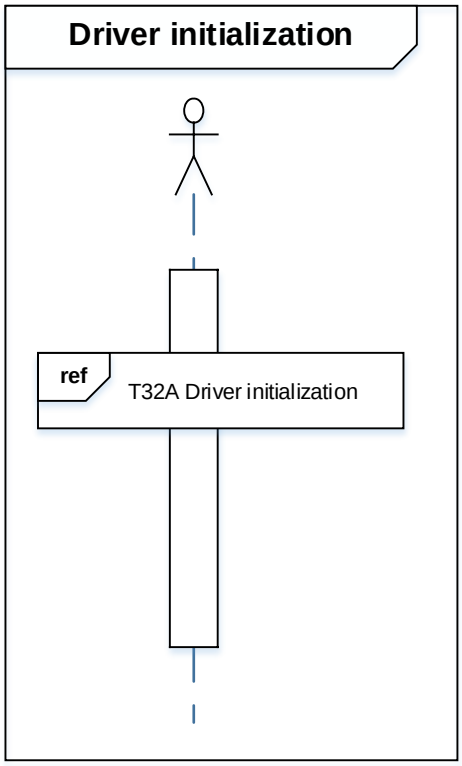
The operation of the terminal software has been checked with the following settings.

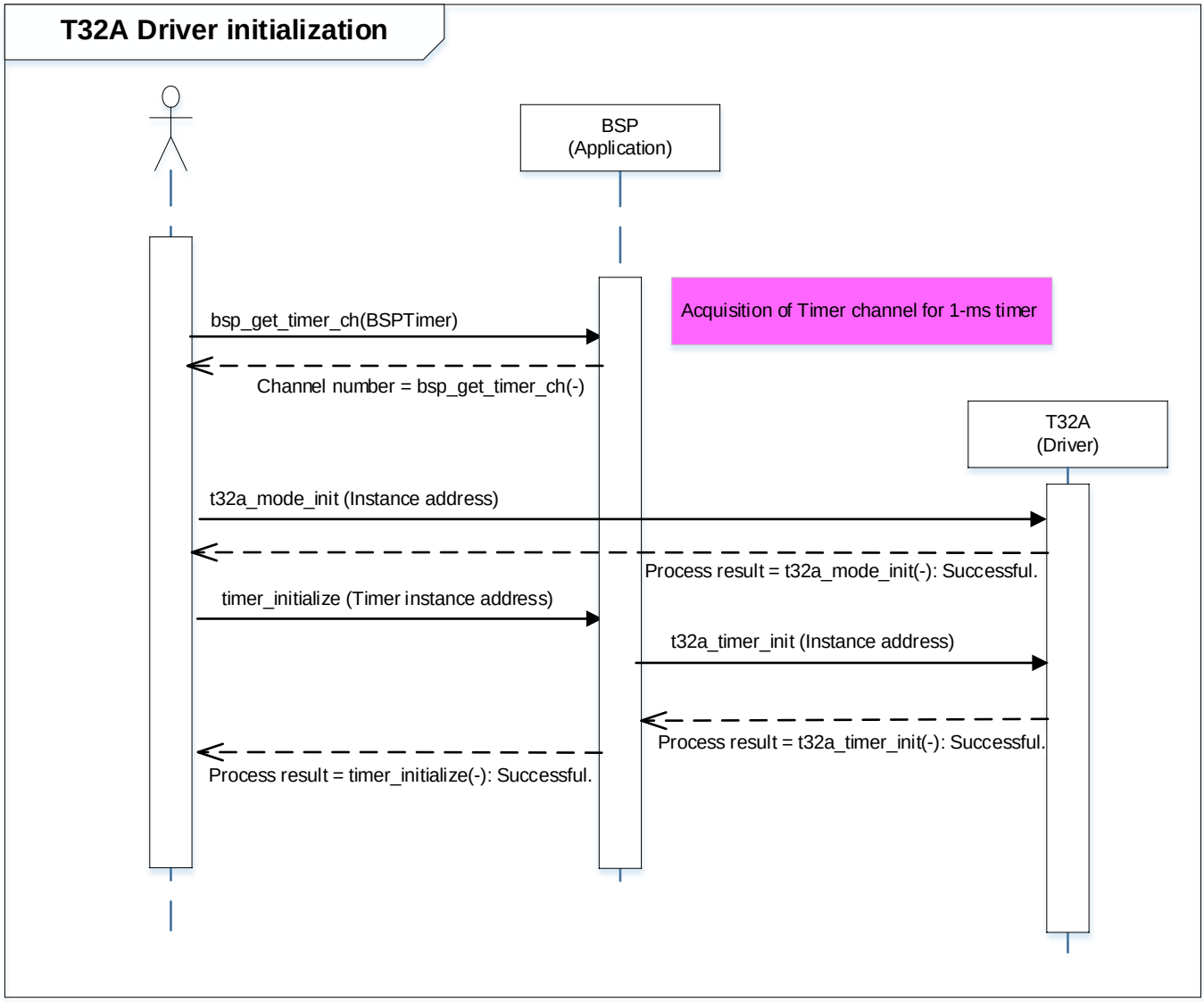


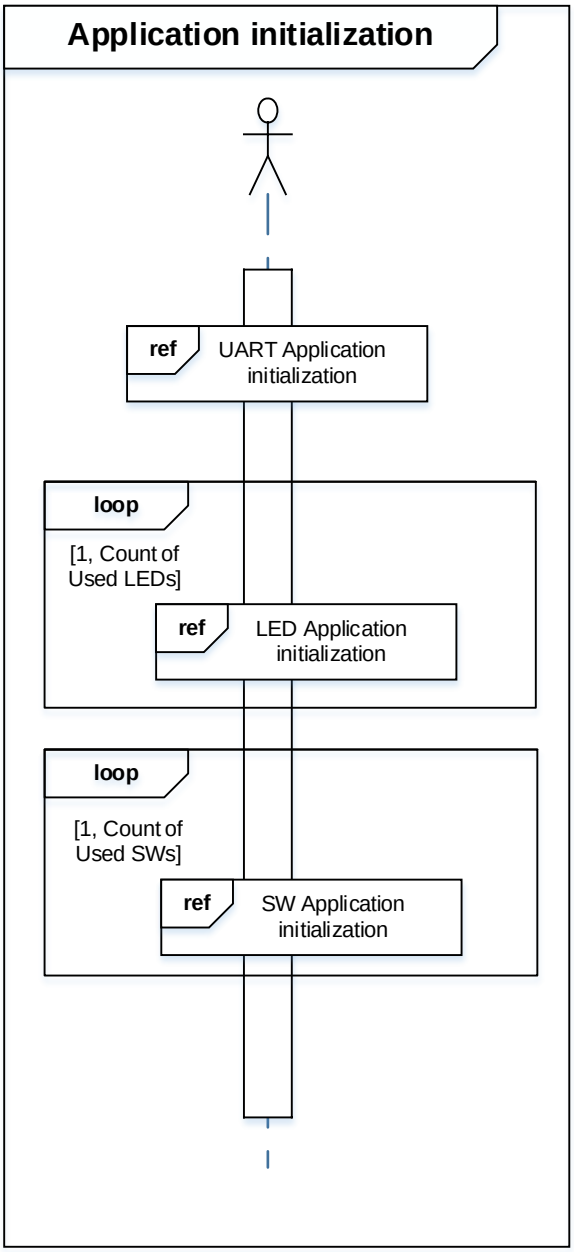
7.6. Operating Flow of Sample Program

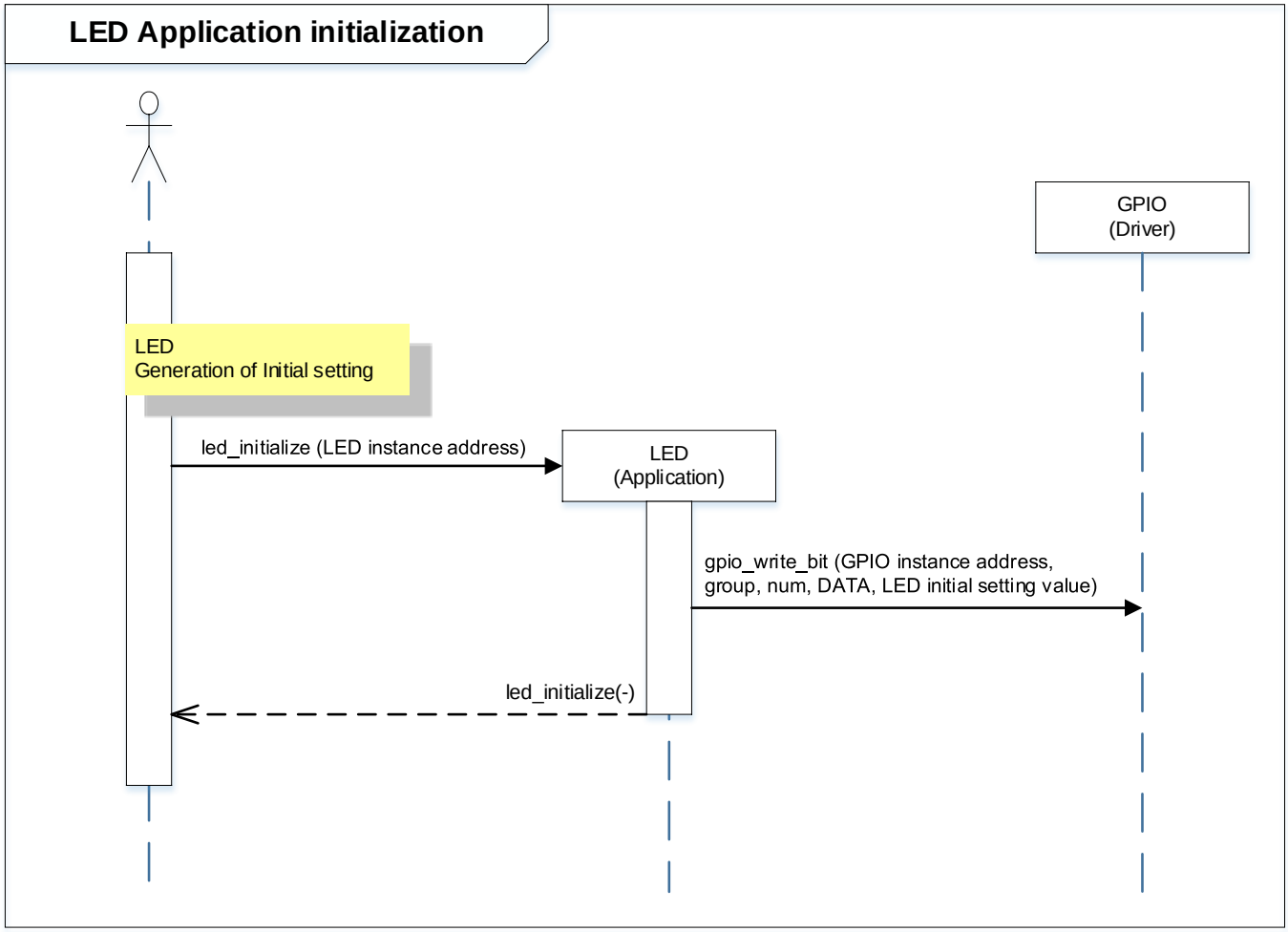
The basic operating flows of the sample program are shown in the following;

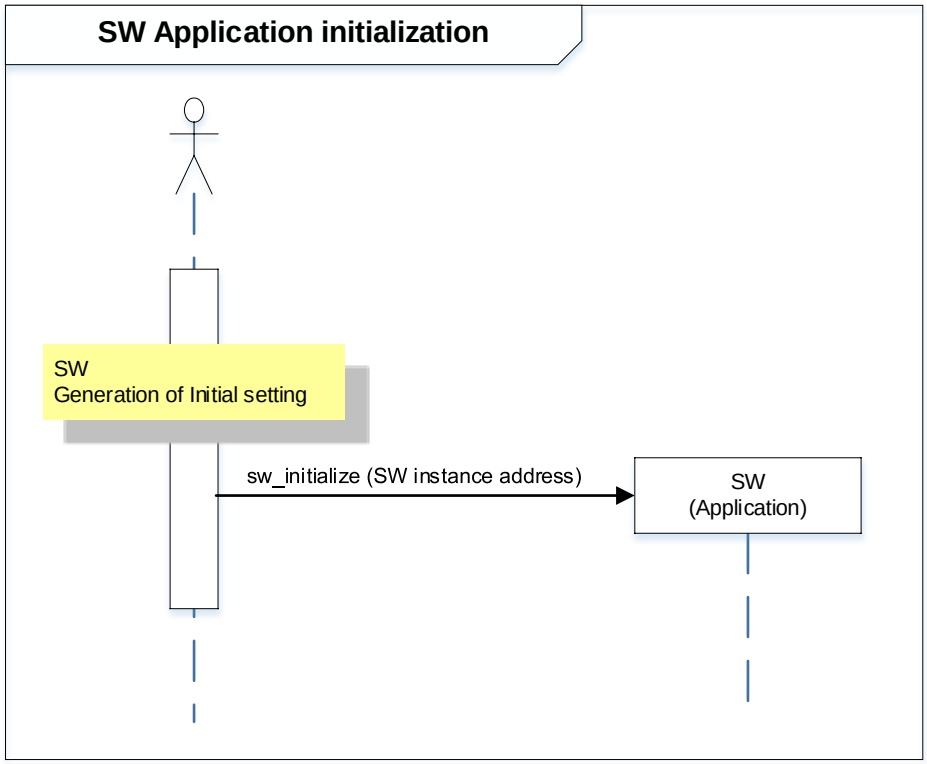


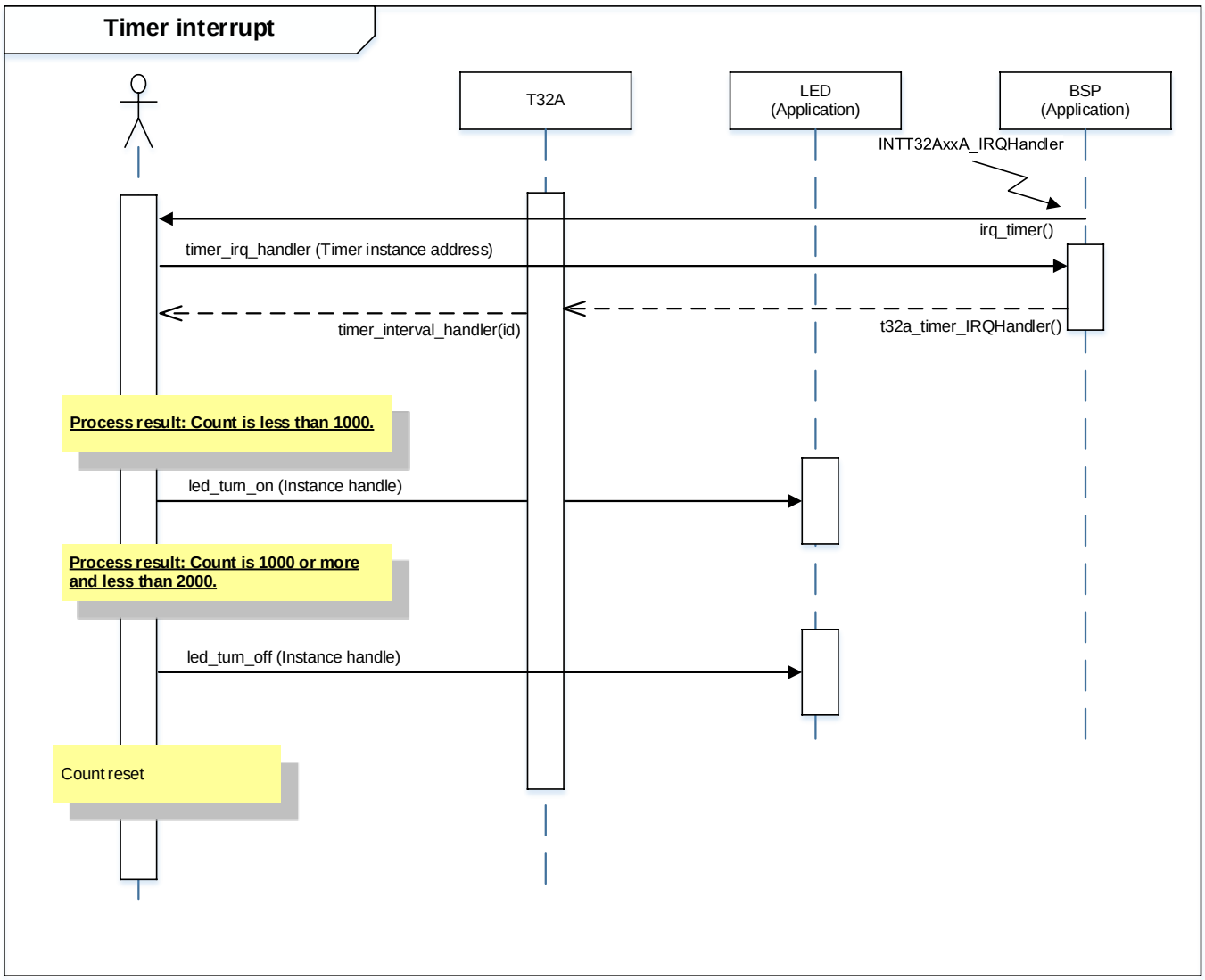

Driver initialization
ref
T32A Driver initialization

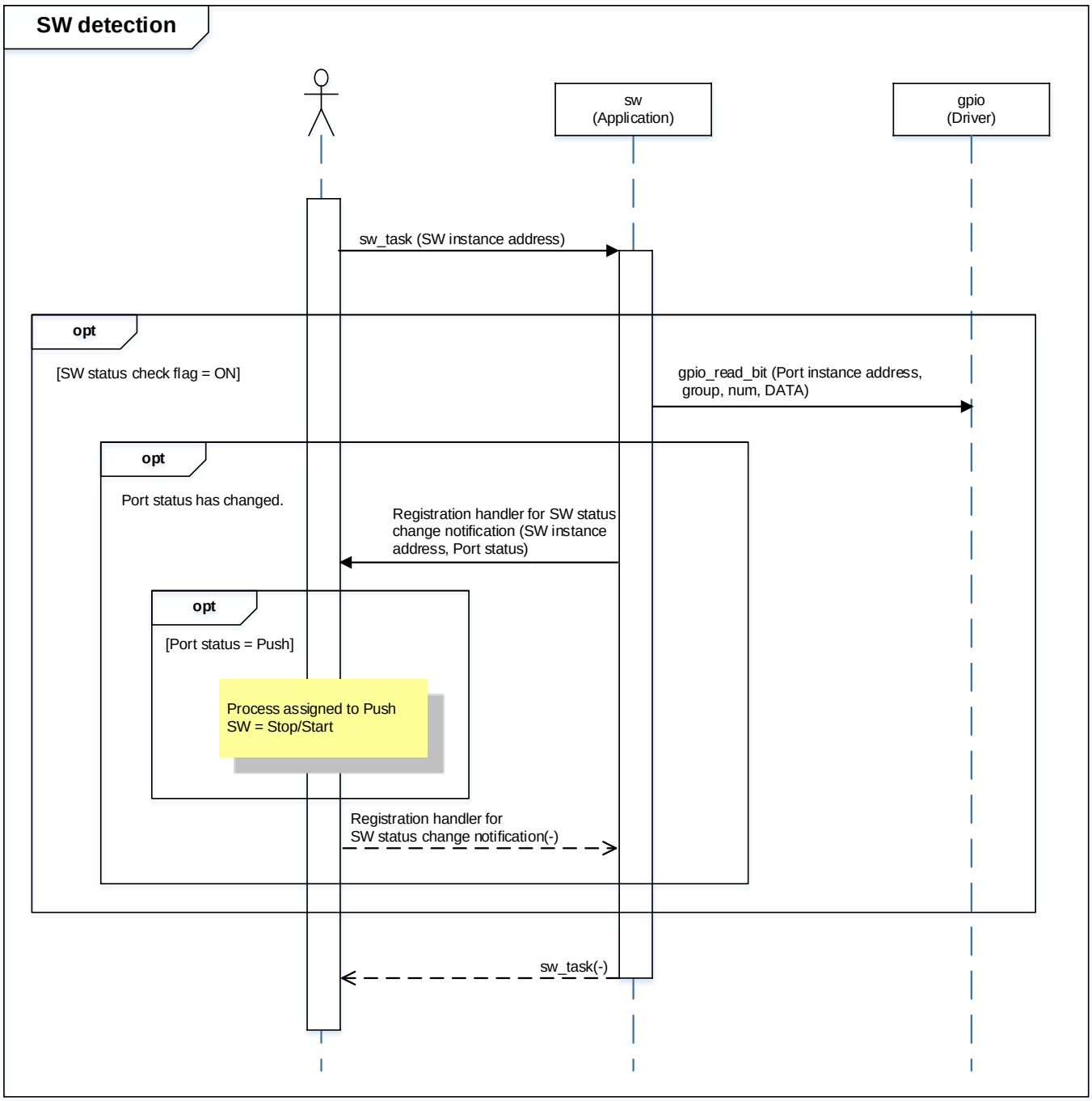

T32A Driver initialization
BSP
(Application)
Acquisition of Timer channel for 1-ms timer
bsp_get_timer_ch(BSPTimer)
Channel number = bsp_get_timer_ch(-)
T32A
(Driver)
t32a_mode_init (Instance address)
Process result = t32a_mode_init(-): Successful.
timer_initialize (Timer instance address)
t32a_timer_init (Instance address)
Process result = t32a_timer_init(-): Successful.
Process result = timer_initialize(-): Successful.


Application initialization
ref
UART Application initialization
loop
[1, Count of Used LEDs]
ref
LED Application initialization
loop
[1, Count of Used SWs]
ref
SW Application initialization


LED Application initialization
GPIO
(Driver)
LED
Generation of Initial setting
led_initialize (LED instance address)
LED
(Application)
gpio_write_bit (GPIO instance address,
group, num, DATA, LED initial setting value)
led_initialize(-)


SW Application initialization
SW
Generation of Initial setting
sw_initialize (SW instance address)
SW
(Application)


Timer interrupt
T32A
LED
(Application)
BSP
(Application)
INTT32AxxA_IRQHandler
irq_timer()
timer_irq_handler (Timer instance address)
timer_interval_handler(id)
t32a_timer_IRQHandler()
Process result: Count is less than 1000.
led_turn_on (Instance handle)
Process result: Count is 1000 or more and less than 2000.
led_turn_off (Instance handle)
Count reset


SW detection
sw
(Application)
gpio
(Driver)
sw_task (SW instance address)
opt
[SW status check flag = ON]
gpio_read_bit (Port instance address,
group, num, DATA)
opt
Port status has changed.
Registration handler for SW status
change notification (SW instance
address, Port status)
opt
[Port status = Push]
Process assigned to Push
SW = Stop/Start
Registration handler for
SW status change notification(-)
sw_task(-)

8. Points to Remember on Handling of Sample Programs

When using the sample program with other than “Operation Confirmation Condition“, please check the operation sufficiently.

9. Revision History

Revision	Date	Description
1.0	2019-10-18	First release

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as “TOSHIBA”.
Hardware, software and systems described in this document are collectively referred to as “Product”.

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**