Application Note

TSPI_MASTER_RECEIVE (TSPI-E)

Table of Contents

1. Preface

This application note describes the sample software for TSPI_MASTER_RECEIVE using Serial Peripheral Interface (SPI).
This document helps the user check operation of a product under development and develop its program.

2. Technical Term

Term/Abbreviation	Definition
BSP	Board Support Package
CG	Clock Control and Operation Mode
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
Timer	T32A:32-bit Timer Event Counter
TSPI	TOSHIBA Serial Peripheral Interface

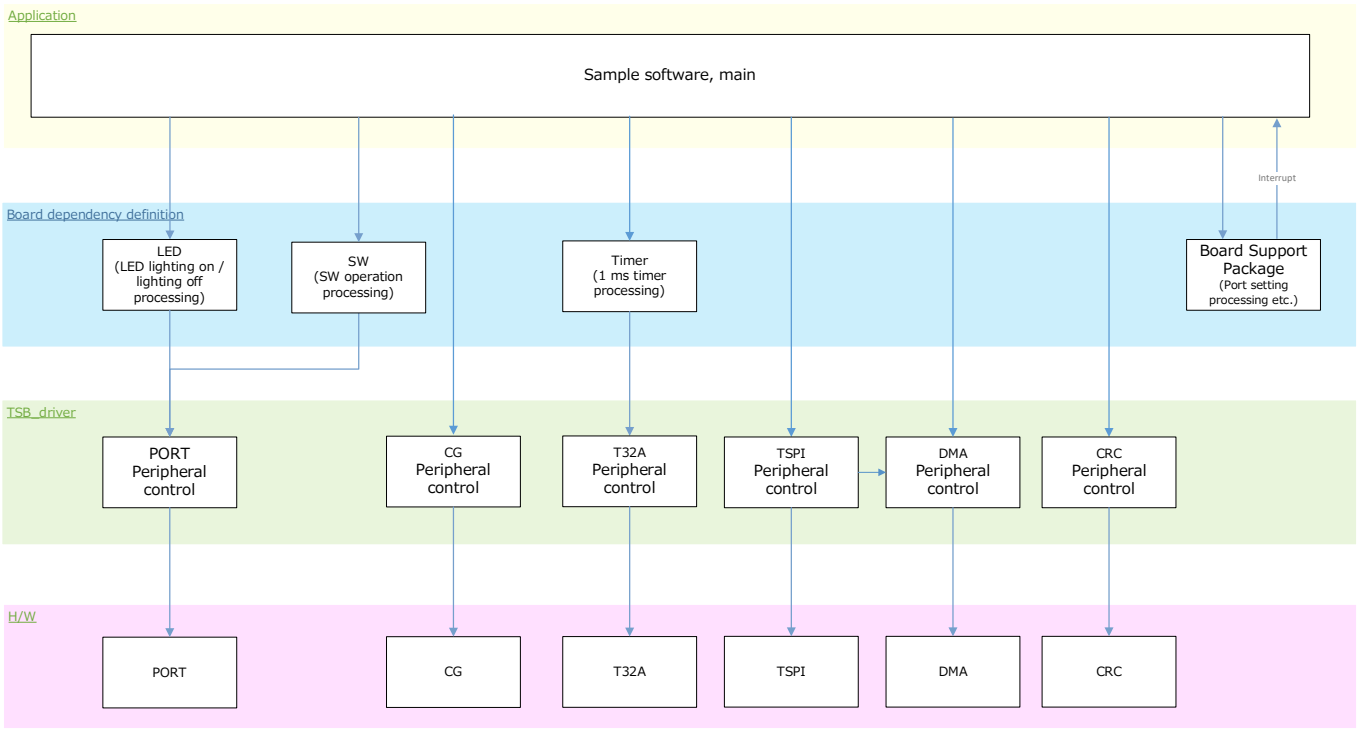
3. Reference Document

Document	Notes
Data sheet	Refer to the data sheet of MCU to be used.
Reference manual	Refer to the reference manual of each IP to be used.
Application note MCU User Guide	Refer to the MCU user guide to be used.

4. Target Sample Program

Sample Program	Outline
TSPI_MASTER_RECEIVE	Sample program of SPI function (Master Receive)

5. Configuration Diagram



6. Sample Program:TSPI_MASTER_RECEIVE

This sample software that uses the Master receive processing function of the SPI communication function to enter the reception waiting state when the switch is pressed, and switches the LED turn on / turn off each time data receive is completed.
This sample software allows you to select FIFO_MODE or DMA_MODE.

6.1. Outlines of Operation

Turns off BSP_LED_1, BSP_LED_3, and BSP_LED_4.
When BSP_PSW_1 is pressed, BSP_LED_3 and BSP_LED_4 are turned off and data for the data size is received. Switches the lighting status (turn on / turn off) of BSP_LED_1.
When an SPI read error occurs, BSP_LED_3 is turn on.
When CRC does not match, BSP_LED_4 is turn on.

6.2. Function to Use

The functions to use are as follows:
For the Port assignment of each BSP channel, refer to the MCU user guide.

IP	Channel	Objective
TSPI	BSP_TSPI_1	SPI Communication
T32A	BSP_T32A_TIMER_1	Interval timer
PORT(Push-Switch)	BSP_PSW_1	Event Trigger
PORT(LED)	BSP_LED_1	For operation check
	BSP_LED_3	For operation check
	BSP_LED_4	For operation check

6.3. Interrupt to Use

Interrupt	Outlines
INTT32A00A	T32A Timer_A Timer counter increment every 1ms for SW processing
*1	SPI receive interrupt
*2	SPI error interrupt
INTDMAATC	DMA transmit end interrupt
INTDMAAERR	DMA error interrupt

*1 For SBK-M4KN, "INTSC0RX", for AdBun-M3HQF10, "INTT1RX"
*2 For SBK-M4KN, "INTSC0ERR", for AdBun-M3HQF10, "INTT1ERR"

6.4. Configuration

“main.c” configuration setting.

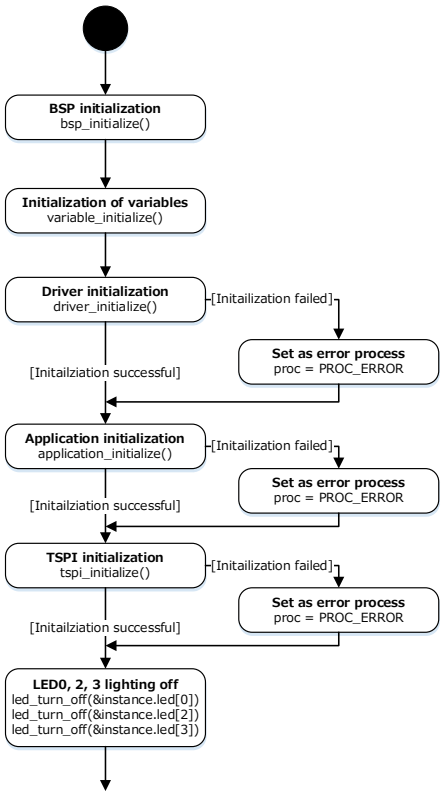
Configuration	Current Value	Description
DATA_LENGTH	10	Data size (Unit: byte) Set to 16 by setting the compile switch CHK_CODE.
CHK_CODE	CHK_CODE_CRC16	CHK_CODE_CRC16 and CHK_CODE_CRC32 can be switched
RX_MODE	FIFO_MODE	FIFO_MODE and DMA_MODE can be switched
RX_FILL_LEVEL	4	Receive Fill level setting

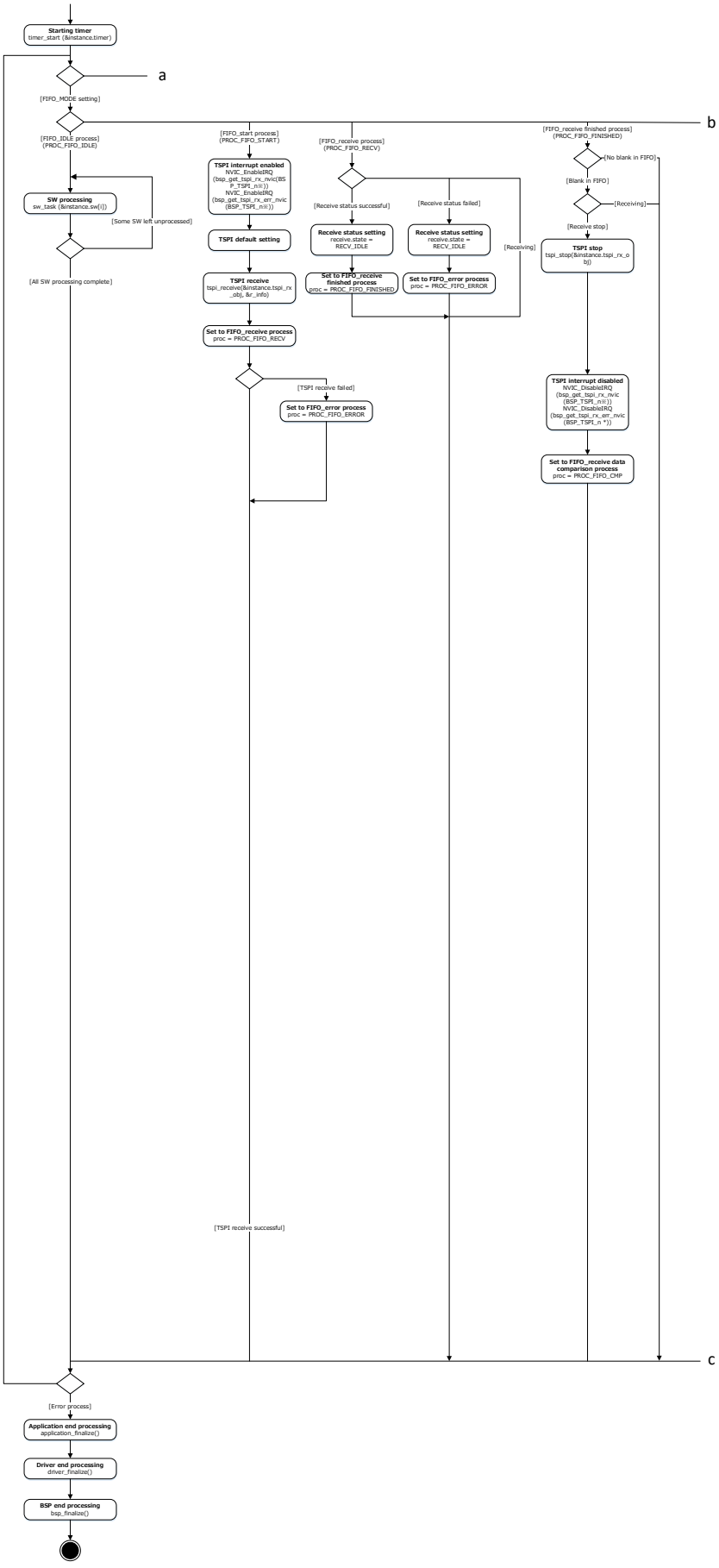
6.5. Example of Terminal Emulator Output

Nothing.

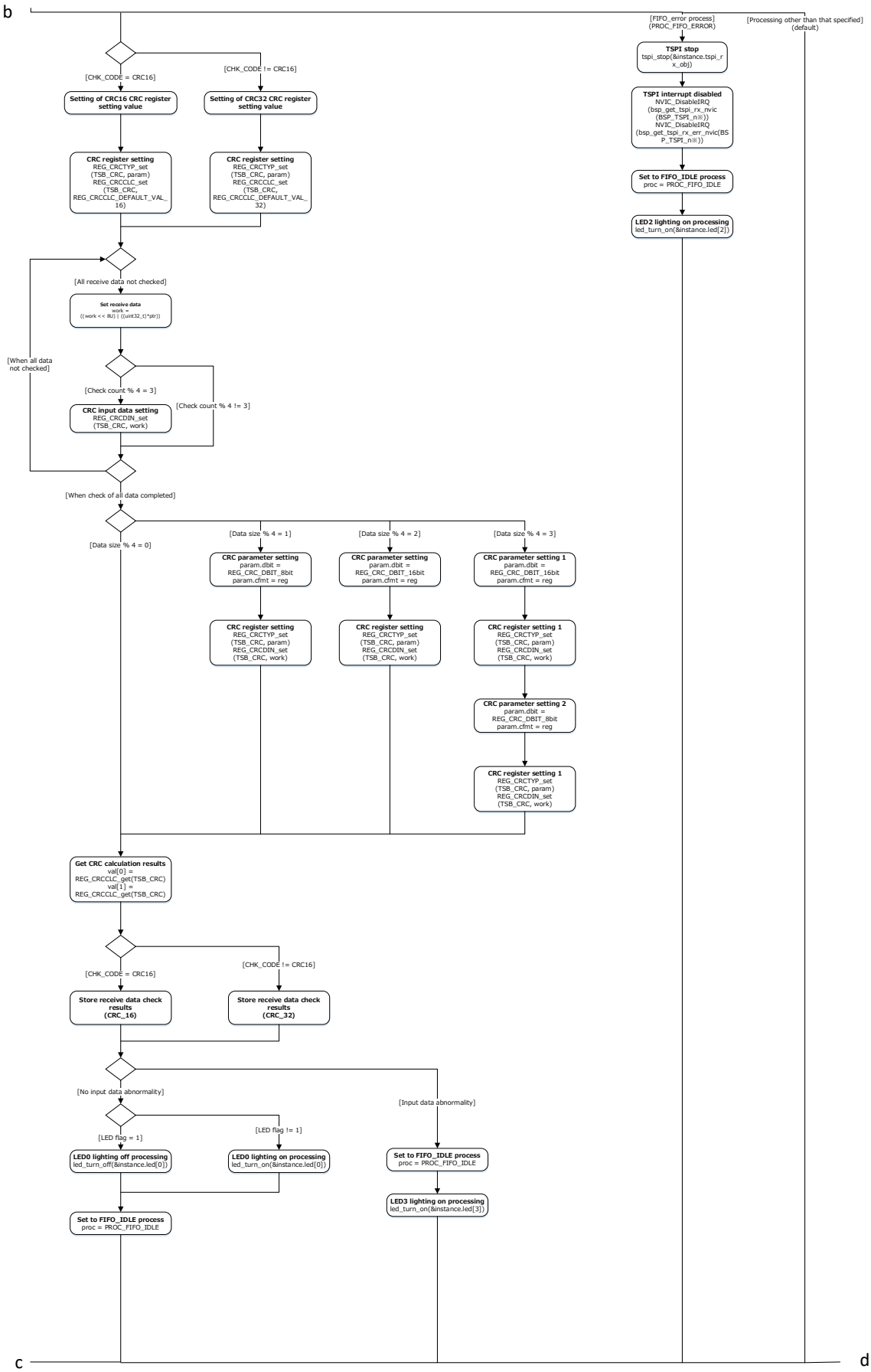
7. Activity diagram

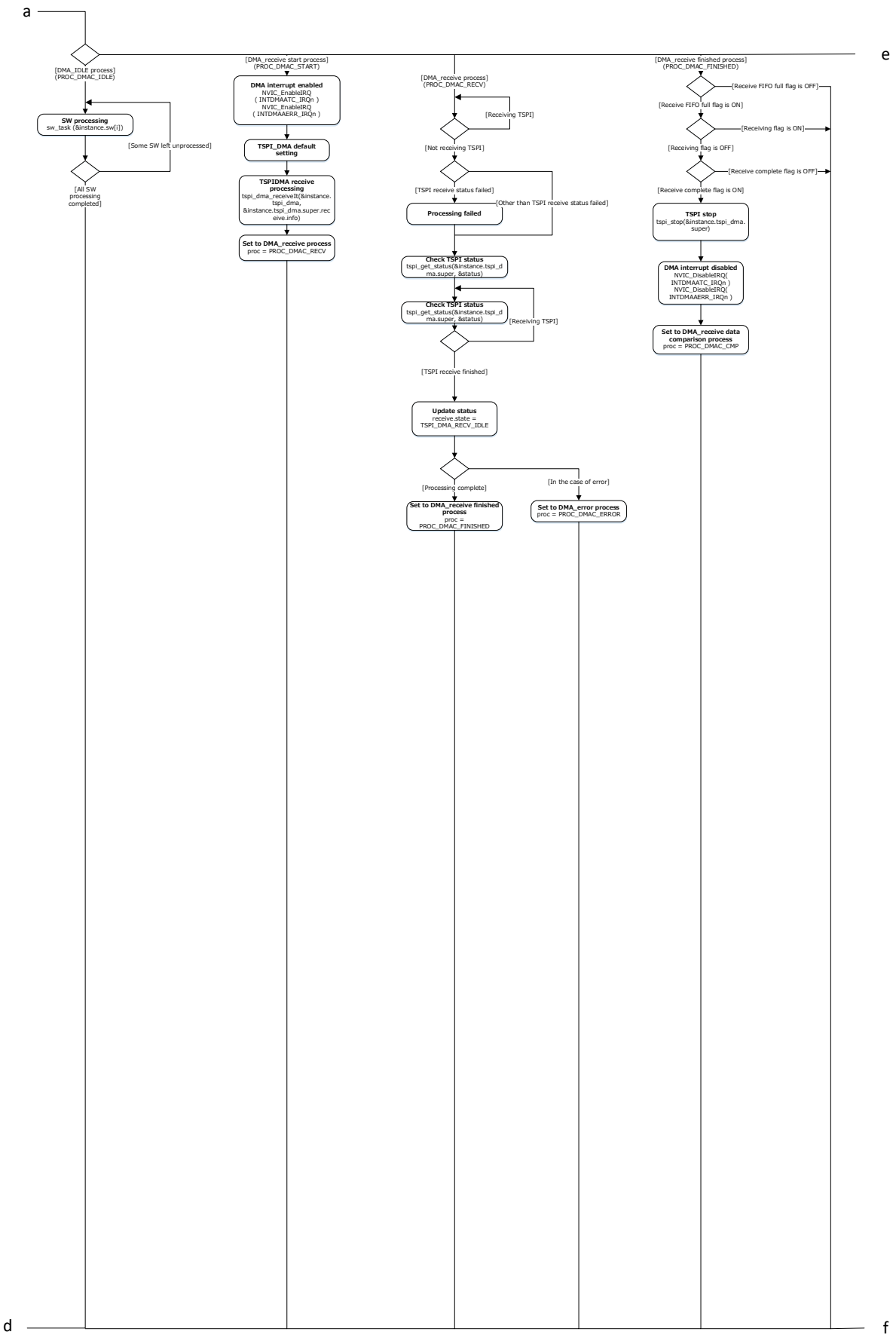
7.1. main



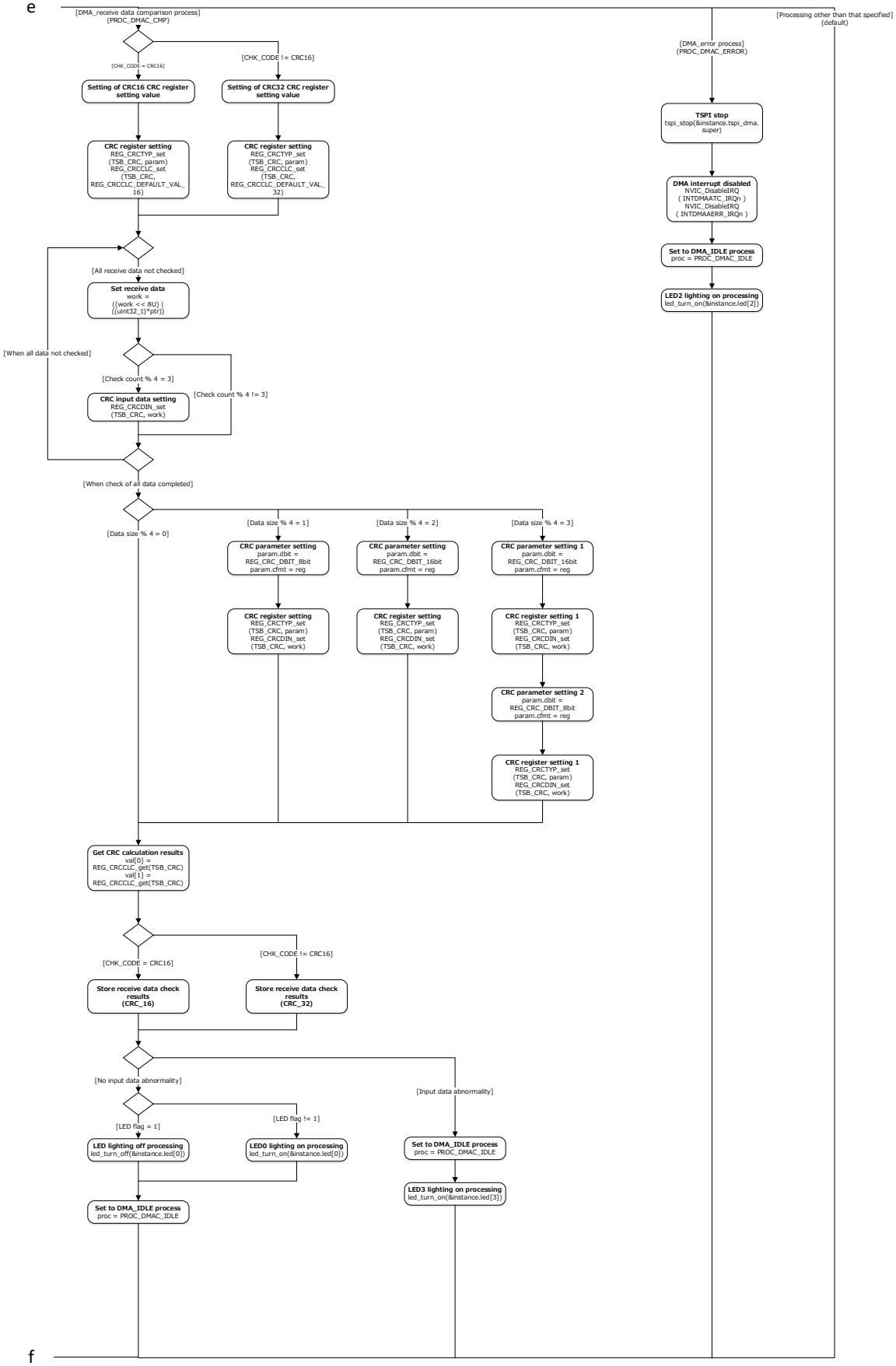


* In the case of M4KN/MN, n=0; in the case of M3H, n=1


b
[FIFO_error process]
(PROC_FIFO_ERROR)
[Processing other than that specified]
(default)
TSPI stop
tspi_stop(&instance.tspi_rx_obj)
TSPI interrupt disabled
NVIC_DisableIRQ
(bsp_get_tspi_rx_nvic
(BSP_TSPI_n※))
NVIC_DisableIRQ
(bsp_get_tspi_rx_err_nvic(BSP_TSPI_n※))
Set to FIFO_IDLE process
proc = PROC_FIFO_IDLE
LED2 lighting on processing
led_turn_on(&instance.led[2])
[CHK_CODE = CRC16]
[CHK_CODE != CRC16]
Setting of CRC16 CRC register setting value
Setting of CRC32 CRC register setting value
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCCLC_set
(TSB_CRC,
REG_CRCCLC_DEFAULT_VAL_16)
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCCLC_set
(TSB_CRC,
REG_CRCCLC_DEFAULT_VAL_32)
[All receive data not checked]
Set receive data
work =
((work << 8U) | ((uint32_t)*ptr))
[When all data not checked]
[Check count % 4 = 3]
[Check count % 4 != 3]
CRC input data setting
REG_CRCDIN_set
(TSB_CRC, work)
[When check of all data completed]
[Data size % 4 = 0]
[Data size % 4 = 1]
[Data size % 4 = 2]
[Data size % 4 = 3]
CRC parameter setting
param.dbit =
REG_CRC_DBIT_8bit
param.cfmt = reg
CRC parameter setting
param.dbit =
REG_CRC_DBIT_16bit
param.cfmt = reg
CRC parameter setting 1
param.dbit =
REG_CRC_DBIT_16bit
param.cfmt = reg
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC register setting 1
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC parameter setting 2
param.dbit =
REG_CRC_DBIT_8bit
param.cfmt = reg
CRC register setting 1
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
Get CRC calculation results
val[0] =
REG_CRCCLC_get(TSB_CRC)
val[1] =
REG_CRCCLC_get(TSB_CRC)
[CHK_CODE = CRC16]
[CHK_CODE != CRC16]
Store receive data check results
(CRC_16)
Store receive data check results
(CRC_32)
[No input data abnormality]
[Input data abnormality]
[LED flag = 1]
[LED flag != 1]
LED0 lighting off processing
led_turn_off(&instance.led[0])
LED0 lighting on processing
led_turn_on(&instance.led[0])
Set to FIFO_IDLE process
proc = PROC_FIFO_IDLE
LED3 lighting on processing
led_turn_on(&instance.led[3])
Set to FIFO_IDLE process
proc = PROC_FIFO_IDLE
c
d

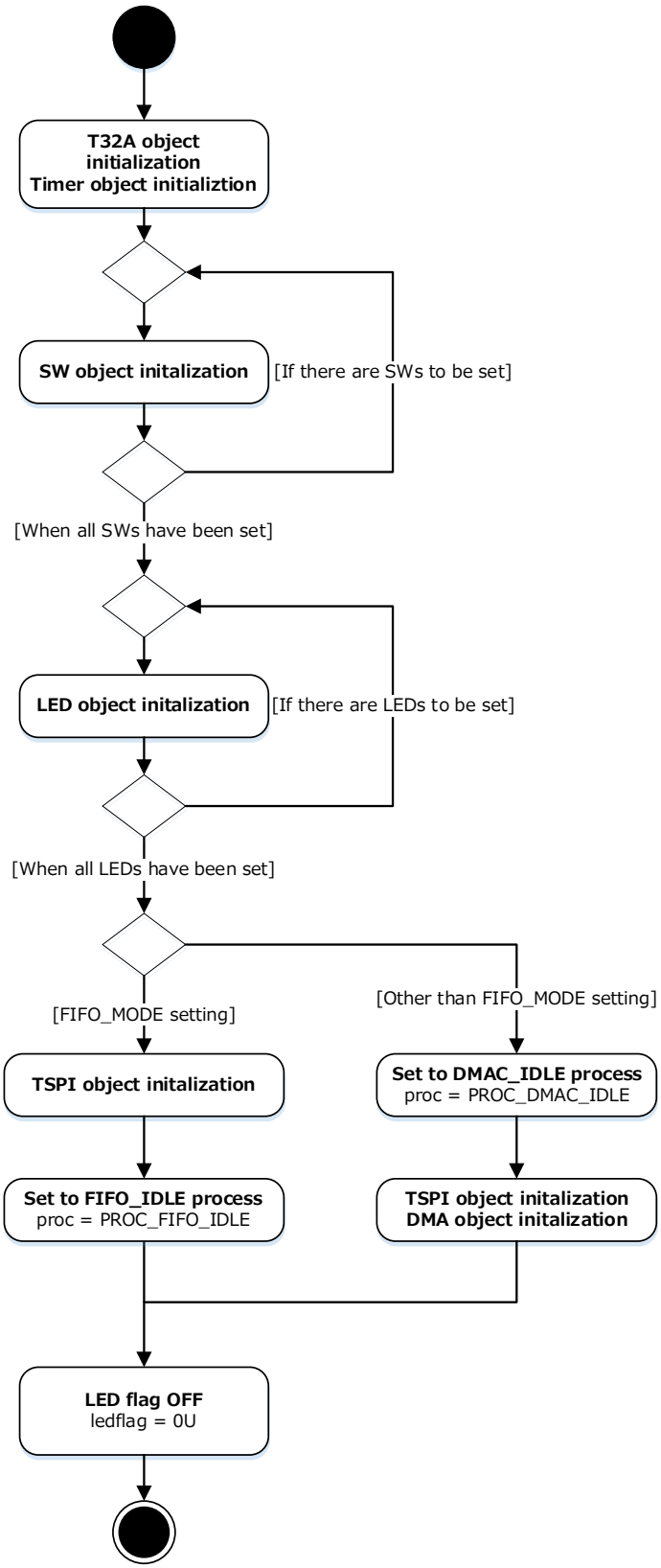

a
e
d
f
[DMA_IDLE process]
(PROC_DMAC_IDLE)
SW processing
sw_task (&instance.sw[i])
[Some SW left unprocessed]
[All SW processing completed]
[DMA_receive start process]
(PROC_DMAC_START)
DMA interrupt enabled
NVIC_EnableIRQ
(INTDMAATC_IRQn)
NVIC_EnableIRQ
(INTDMAAERR_IRQn)
TSPI_DMA default setting
TSPIDMA receive processing
tspi_dma_receiveIt(&instance.tspi_dma, &instance.tspi_dma.super.receive.info)
Set to DMA_receive process
proc = PROC_DMAC_RECV
[DMA_receive process]
(PROC_DMAC_RECV)
[Receiving TSPI]
[Not receiving TSPI]
[TSPI receive status failed]
[Other than TSPI receive status failed]
Processing failed
Check TSPI status
tspi_get_status(&instance.tspi_dma.super, &status)
Check TSPI status
tspi_get_status(&instance.tspi_dma.super, &status)
[Receiving TSPI]
[TSPI receive finished]
Update status
receive.state =
TSPI_DMA_RECV_IDLE
[Processing complete]
[In the case of error]
Set to DMA_receive finished process
proc =
PROC_DMAC_FINISHED
Set to DMA_error process
proc = PROC_DMAC_ERROR
[DMA_receive finished process]
(PROC_DMAC_FINISHED)
[Receive FIFO full flag is OFF]
[Receive FIFO full flag is ON]
[Receiving flag is ON]
[Receiving flag is OFF]
[Receive complete flag is OFF]
[Receive complete flag is ON]
TSPI stop
tspi_stop(&instance.tspi_dma.super)
DMA interrupt disabled
NVIC_DisableIRQ(
INTDMAATC_IRQn)
NVIC_DisableIRQ(
INTDMAAERR_IRQn)
Set to DMA_receive data comparison process
proc = PROC_DMAC_CMP

e

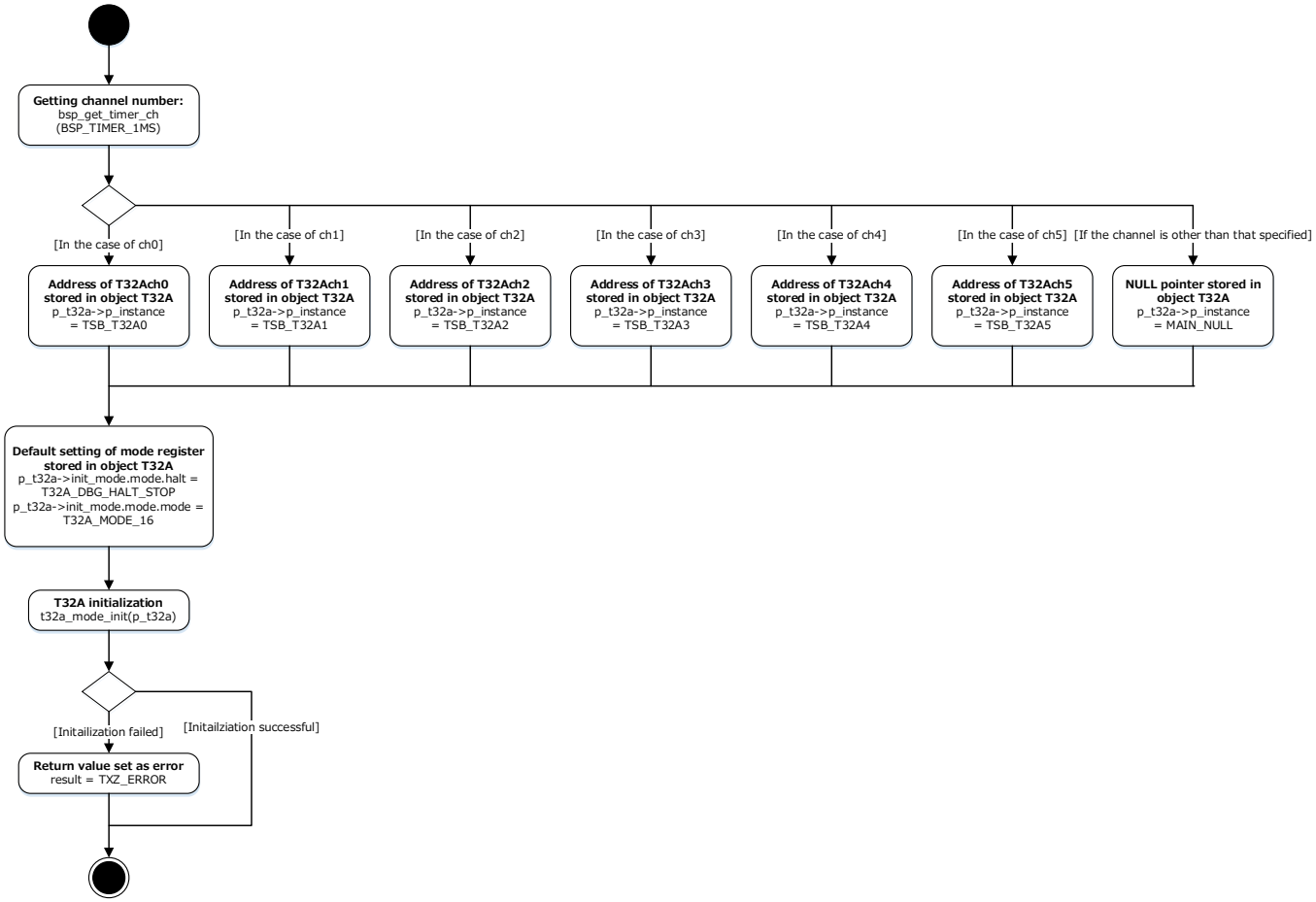

[DMA_receive data comparison process]
(PROC_DMAC_CMP)
[CHK_CODE = CRC16]
[CHK_CODE != CRC16]
Setting of CRC16 CRC register setting value
Setting of CRC32 CRC register setting value
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCCLC_set
(TSB_CRC,
REG_CRCCLC_DEFAULT_VAL_16)
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCCLC_set
(TSB_CRC,
REG_CRCCLC_DEFAULT_VAL_32)
[All receive data not checked]
Set receive data
work =
((work << 8U) |
((uint32_t)*ptr))
[When all data not checked]
[Check count % 4 = 3]
[Check count % 4 != 3]
CRC input data setting
REG_CRCDIN_set
(TSB_CRC, work)
[When check of all data completed]
[Data size % 4 = 0]
[Data size % 4 = 1]
[Data size % 4 = 2]
[Data size % 4 = 3]
CRC parameter setting
param.dbit =
REG_CRC_DBIT_8bit
param.cfmt = reg
CRC parameter setting
param.dbit =
REG_CRC_DBIT_16bit
param.cfmt = reg
CRC parameter setting 1
param.dbit =
REG_CRC_DBIT_16bit
param.cfmt = reg
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC register setting
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC register setting 1
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
CRC parameter setting 2
param.dbit =
REG_CRC_DBIT_8bit
param.cfmt = reg
CRC register setting 1
REG_CRCTYP_set
(TSB_CRC, param)
REG_CRCDIN_set
(TSB_CRC, work)
Get CRC calculation results
val[0] =
REG_CRCCLC_get(TSB_CRC)
val[1] =
REG_CRCCLC_get(TSB_CRC)
[CHK_CODE = CRC16]
[CHK_CODE != CRC16]
Store receive data check results
(CRC_16)
Store receive data check results
(CRC_32)
[No input data abnormality]
[Input data abnormality]
[LED flag = 1]
[LED flag != 1]
LED lighting off processing
led_turn_off(&instance.led[0])
LED0 lighting on processing
led_turn_on(&instance.led[0])
Set to DMA_IDLE process
proc = PROC_DMAC_IDLE
LED3 lighting on processing
led_turn_on(&instance.led[3])
Set to DMA_IDLE process
proc = PROC_DMAC_IDLE
[DMA_error process]
(PROC_DMAC_ERROR)
TSPI stop
tspi_stop(&instance.tspi_dma.super)
DMA interrupt disabled
NVIC_DisableIRQ
(INTDMAATC_IRQn)
NVIC_DisableIRQ
(INTDMAAERR_IRQn)
Set to DMA_IDLE process
proc = PROC_DMAC_IDLE
LED2 lighting on processing
led_turn_on(&instance.led[2])
[Processing other than that specified]
(default)

f

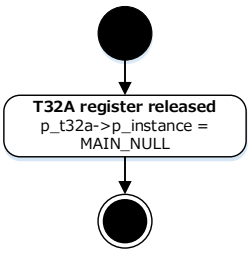
7.2. variable_initalize



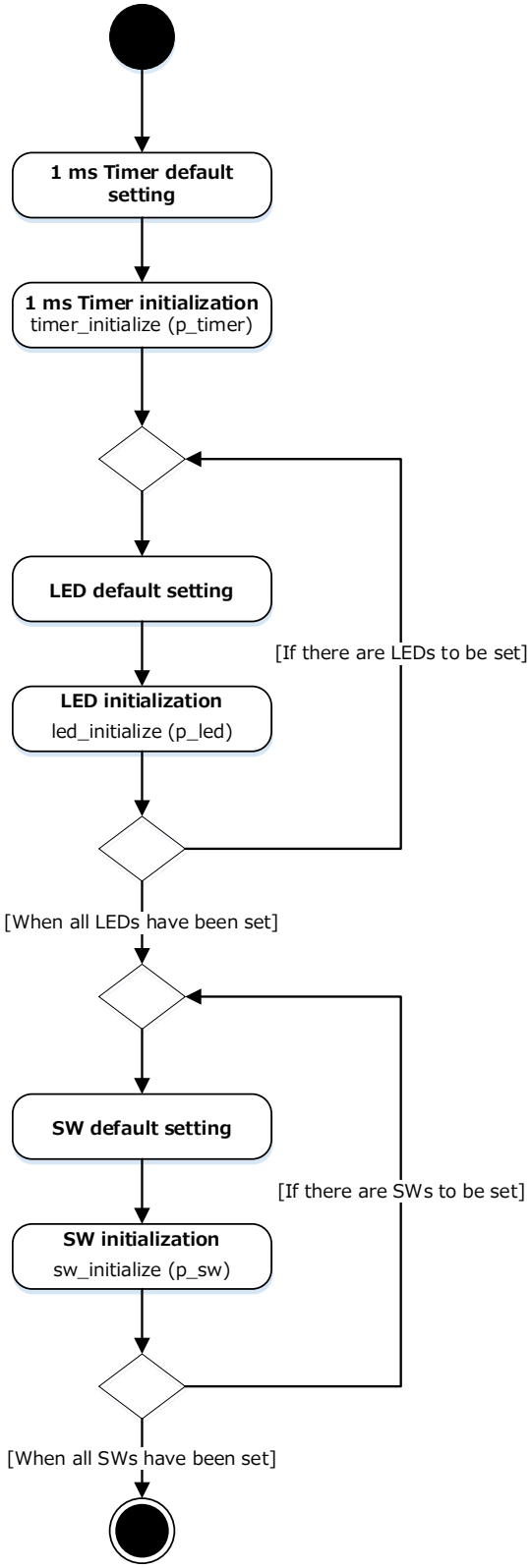
7.3. driver_initialize



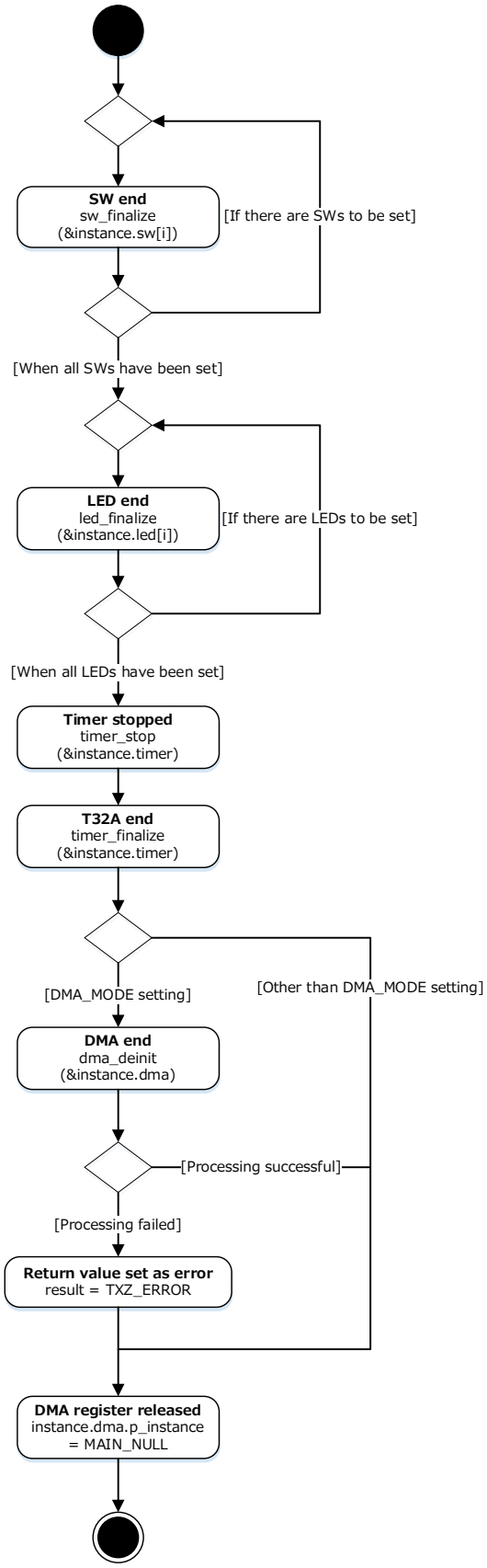
7.4. driver_finalize



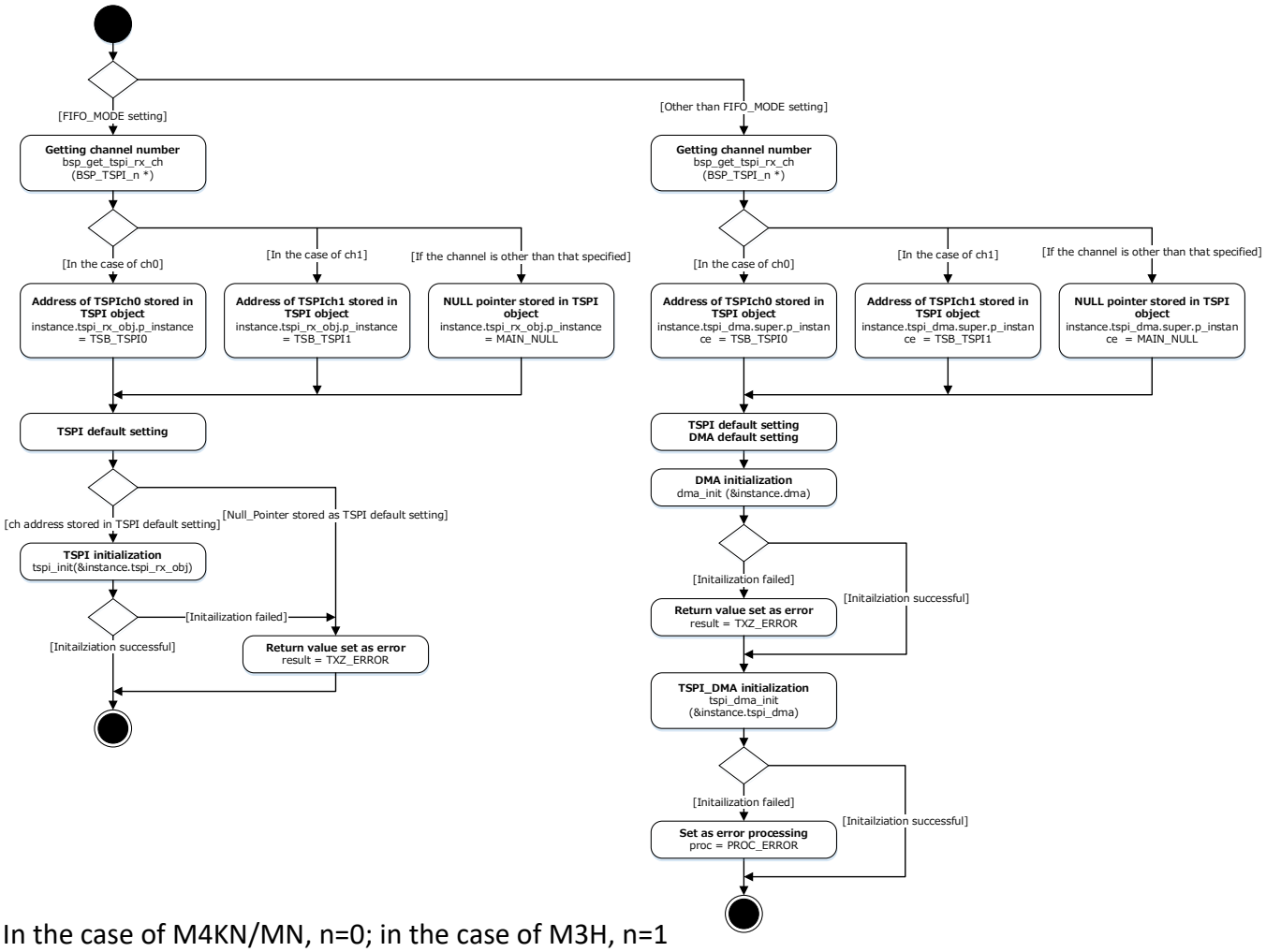
7.5. application_initialize



7.6. application_finalize

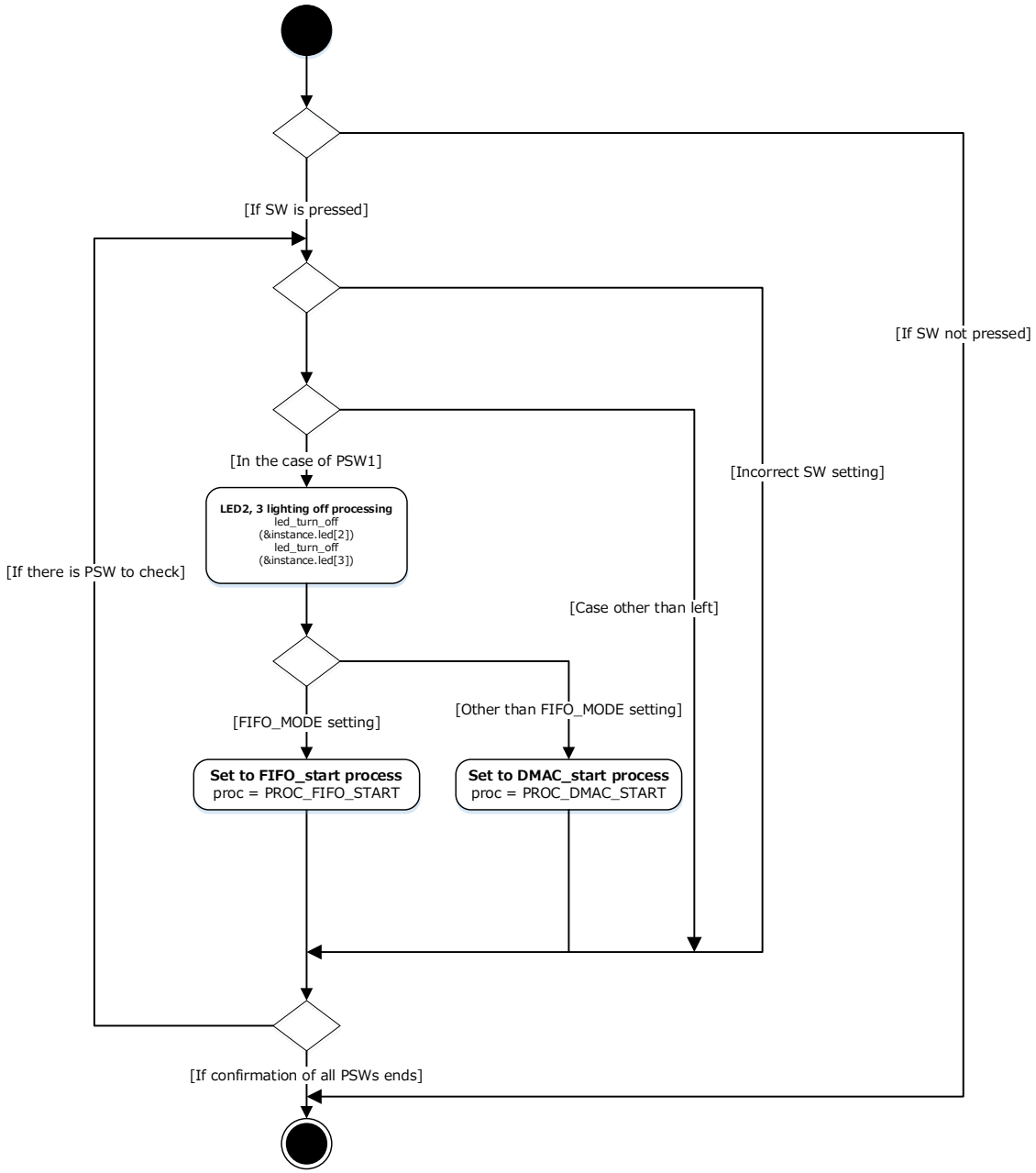


7.7. tspi_initialize

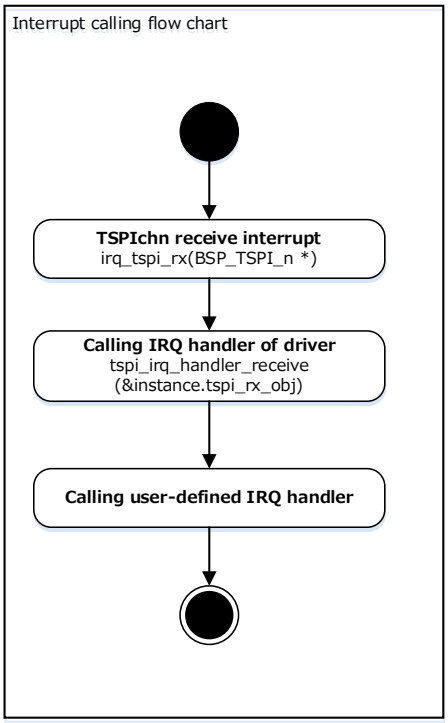


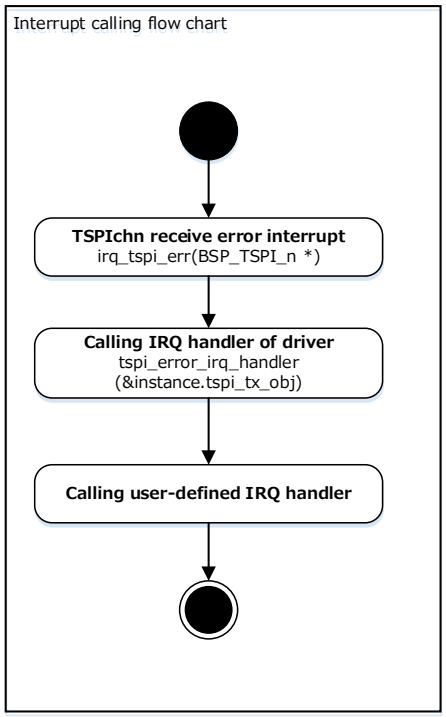
* In the case of M4KN/MN, n=0; in the case of M3H, n=1

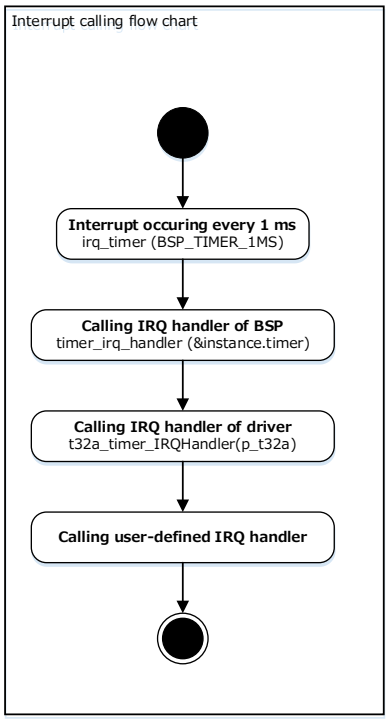
7.8. sw_state_change_handler

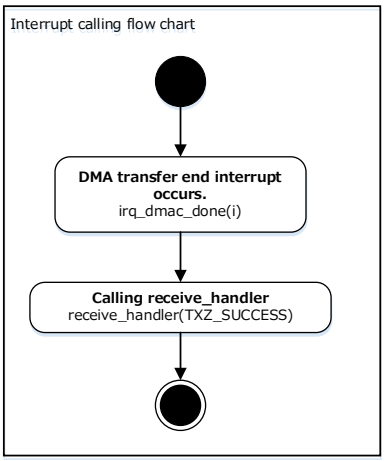


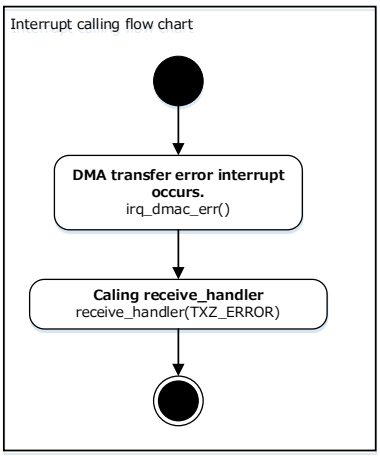
7.9. Interrupt

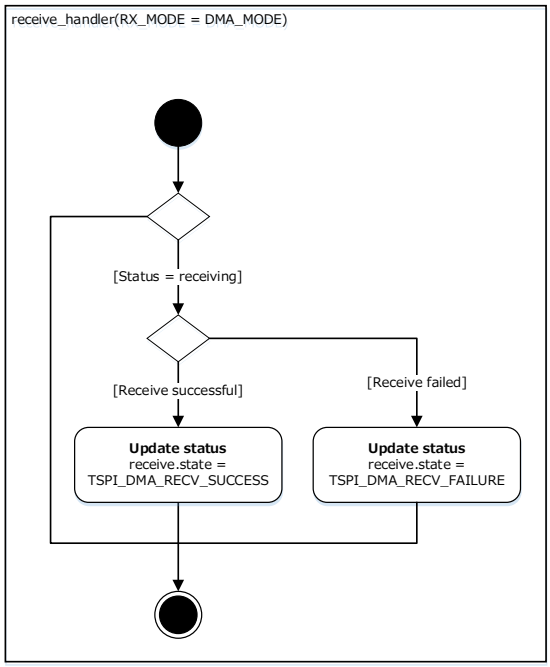




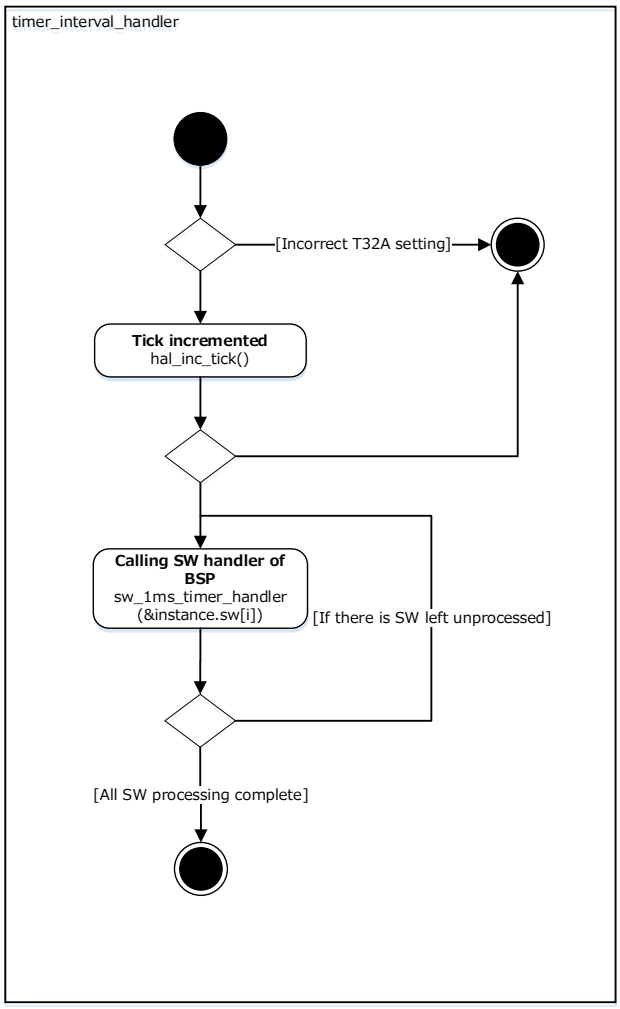


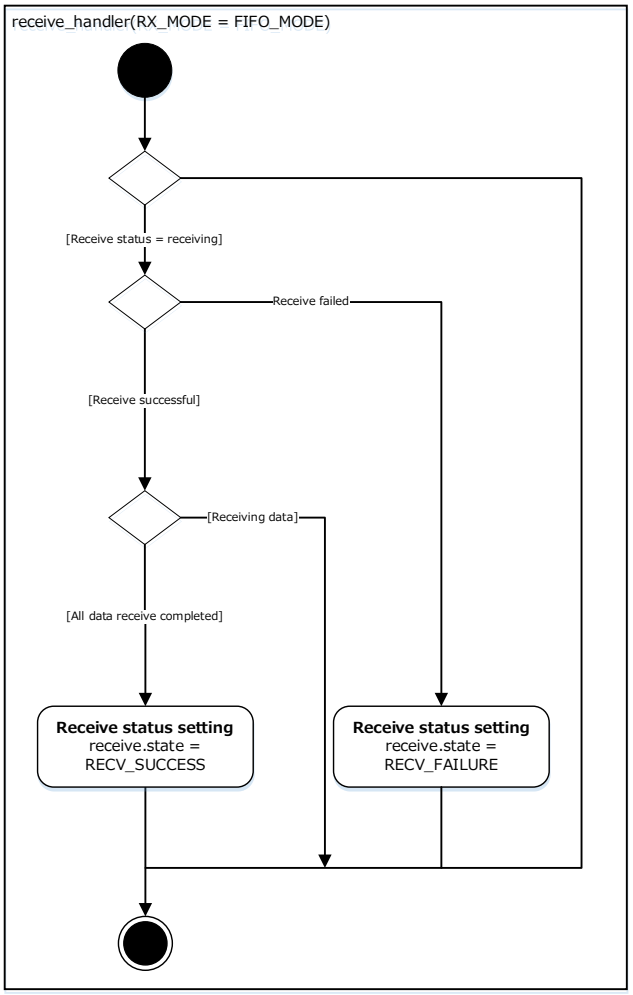


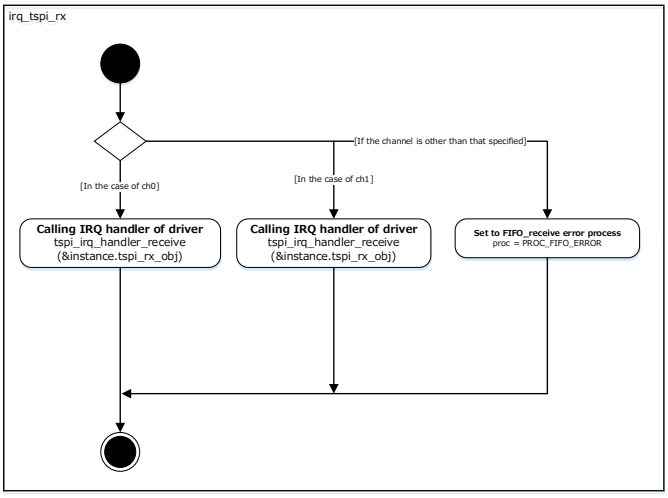


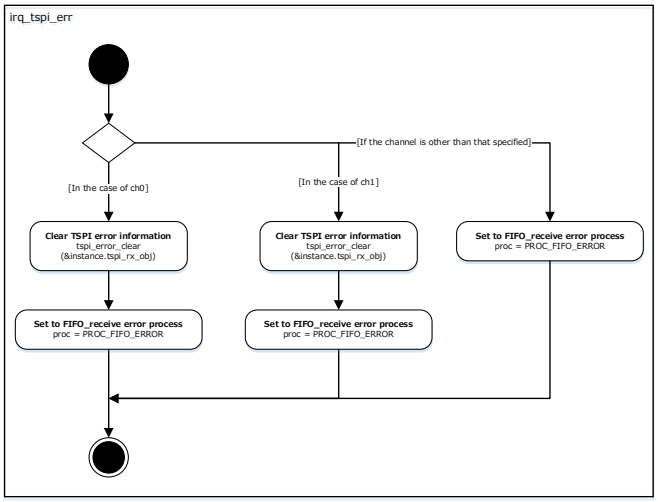


* In the case of M4KN/MN, n=0; in the case of M3H, n=1









8. Revision History

Revision	Date	Description
1.0	2023-10-16	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.**