M4G Group (1)
Application Note
High-Speed DMA Controller
(HDMAC-A)
Memory to Memory

Outlines

This application note is a reference material for developing products that the function of transferring data from the memory to the memory using the high speed DMA controller (HDMAC) of the M4G group (1). This document helps the user check operation of the product and develop its program.

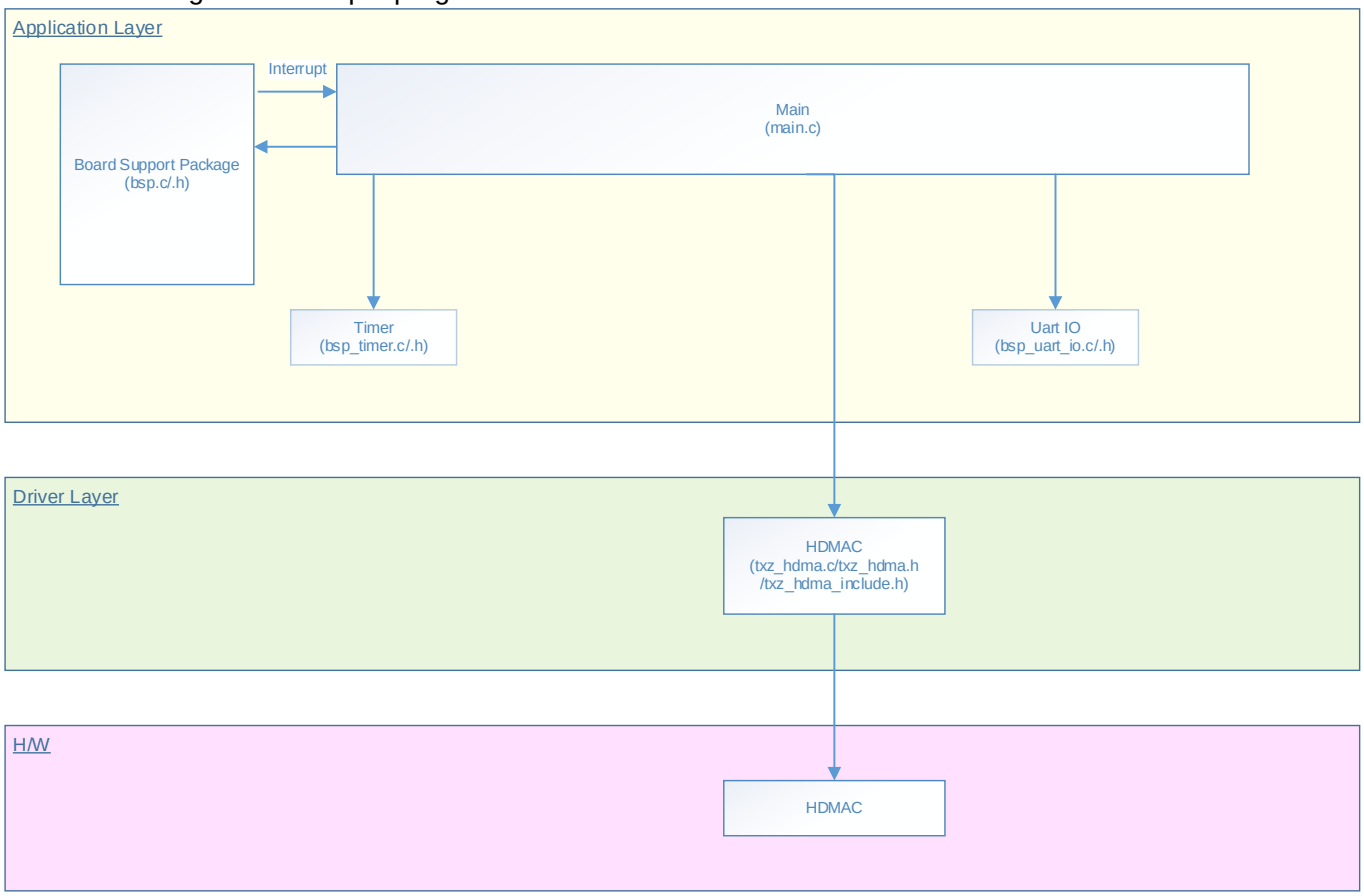
Target sample program: HDMA_MEM_MEM

Table of Contents

1. Preface

This sample program is used to check the operation of the HDMAC function which transfers data between memories.
The program transfers reception data directly between a memory and another memory. The CPU is not used for the data transfer.

Structure diagram of Sample program



2. Reference Document

- Datasheet
  TMPM4G Group (1) datasheet Rev1.0 (Japanese edition)
- Reference manual
  High Speed DMA Controller (HDMAC-A) Rev1.0 (Japanese edition)
  Asynchronous Serial Communication Circuit (UART-C) Rev3.0 (Japanese edition)
- Application Note
  M4G Group (1) Application Note Startup (CMSIS System & Clock Configuration) Rev1.0
- Other reference document
  TMPM4G(1) Group Peripheral Driver User Manual (Doxygen)

3. Function to Use

IP	Channel	Port	Function/Operation mode
High Speed DMA Controller	ch0 (transfer) ch0 (request)	-	Data transfer between memories Single transfer
Asynchronous Serial Communication Circuit	ch0	PE2 (UT0RXD) PE3 (UT0TXDA)	UART mode

4. Target Device

The target devices of this application note are as follows;

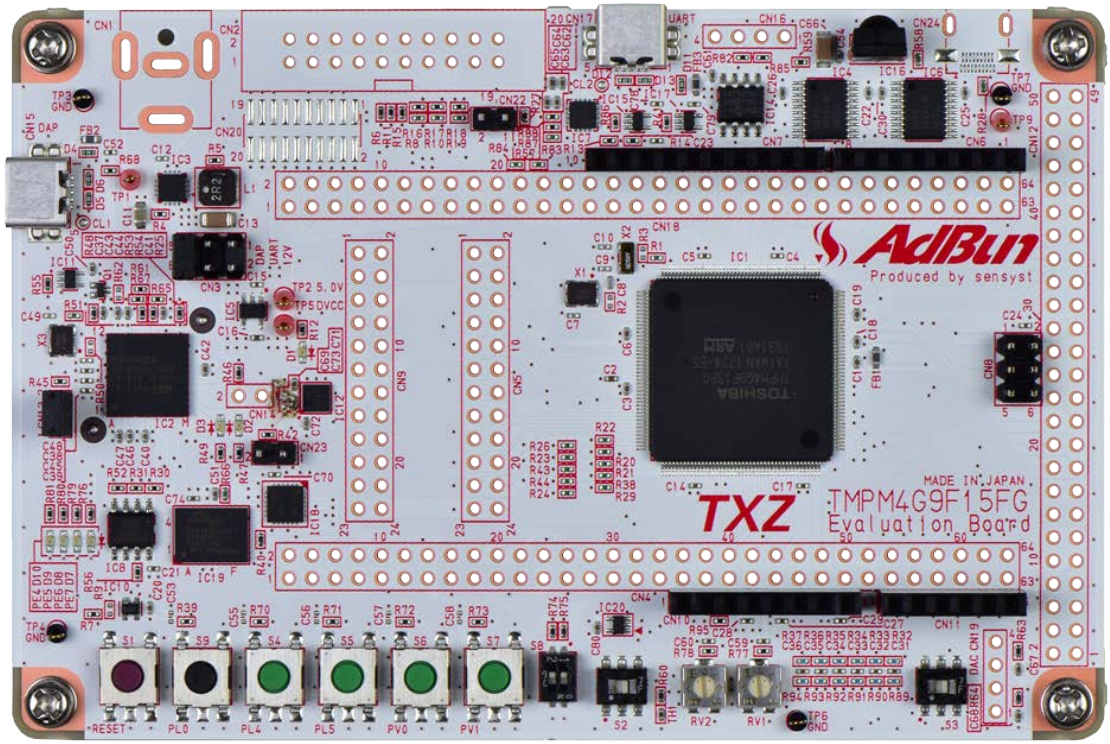
TMPM4G9F15FG	TMPM4G9F10FG	TMPM4G9FEFG	TMPM4G9FDFG
TMPM4G9F15XBG	TMPM4G9F10XBG	TMPM4G9FEXBG	TMPM4G9FDXBG
TMPM4G8F15FG	TMPM4G8F10FG	TMPM4G8FEFG	TMPM4G8FDFG
TMPM4G8F15XBG	TMPM4G8F10XBG	TMPM4G8FEXBG	TMPM4G8FDXBG
	TMPM4G7F10FG	TMPM4G7FEFG	TMPM4G7FDFG
	TMPM4G6F10FG	TMPM4G6FEFG	TMPM4G6FDFG

*This sample program operates on the evaluation board of TMPM4G9F15FG.
If other function than the TMPM4G9F15 one is checked, it is necessary that CMSIS Core related files (C startup file and I/O header file) should be changed properly.
The BSP related file is dedicated to the evaluation board (TMPM4G9F15). If other function than the TMPM4G9F15 one is checked, the BSP related file should be changed properly.

5. Operation Confirmation Condition

Used microcontroller	TMPM4G9F15FG
Used board	TMPM4G9F15FG Evaluation Board by Sensyst
Unified development environment	IAR Embedded Workbench for ARM 8.11.2.13606
Unified development environment	µVision MDK Version 5.24.2.0
Terminal software	Tera Term V4.96
Sample program	V1000

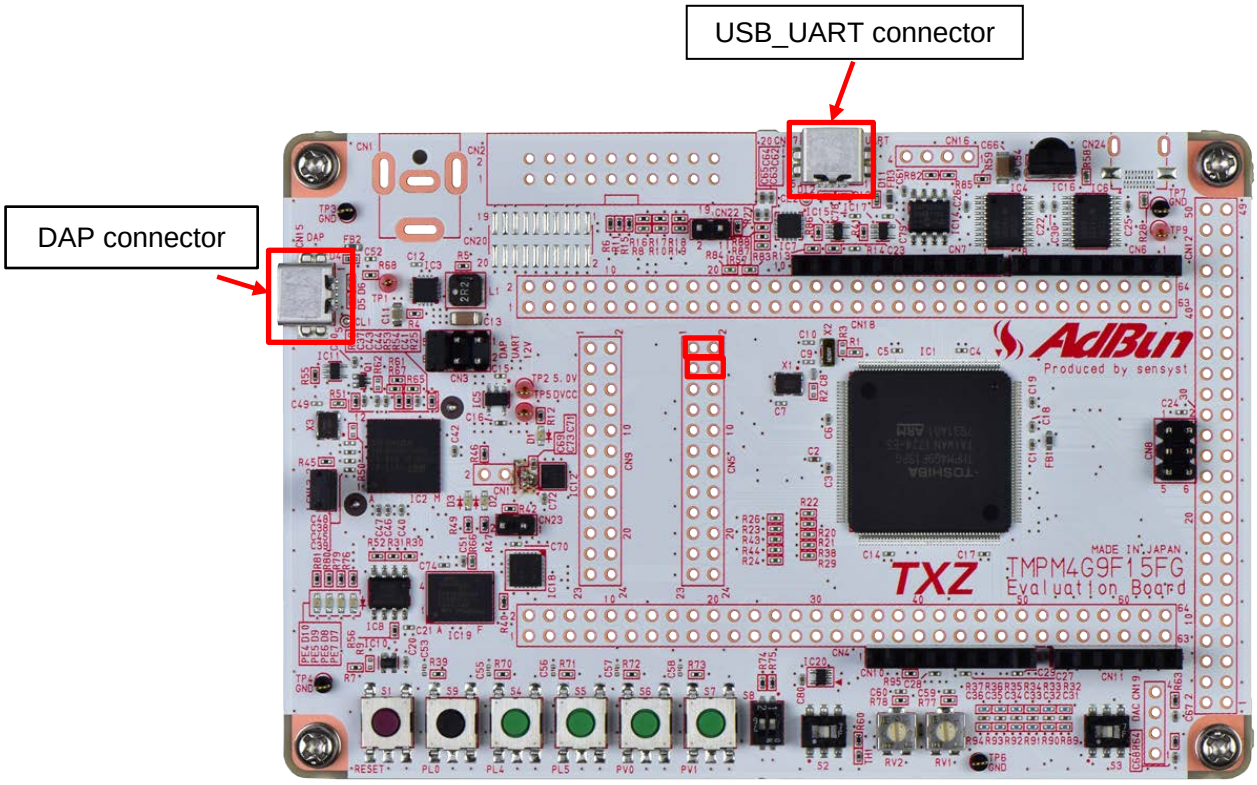
Evaluation board (TMPM4G9F15FG Evaluation Board) Top view



6. Evaluation Board Setting

The following pin connections should be done on the evaluation board.

CN5		
Board function	Through-hole No.	Through-hole No.
USB UART conversion	1: USB_UT_RX	2: PE2
USB UART conversion	3: USB_UT_TX	4: PE3



7. Operation of Evaluation Board

PC and the USB_UART are connected for communication with the terminal software.
An arbitrary character should be input to the terminal software.
After the input data is transferred from a memory to another memory, it is output to the terminal software through the UART interface.

8. Outline of HDMAC function

The functions of HDMAC per unit are shown in the following table.

Function category	Function	Description
Transfer request	Peripheral function	Transfer request from a peripheral device (Single transfer request/Burst transfer request)
	Software	Transfer request by software (Single transfer request/Burst transfer request)
	Request channel	16 channels (Transfer request count by peripheral devices)
Transfer mode	Single transfer	Data transfer is executed once.
	Burst transfer	Data transfer is executed once or multiple times Burst size: 1, 4, 8, 16, 32, 64, 128, and 256 beats Continuous data transfer of the specified burst count is possible without releasing the bus using the lock transfer setting.
	Chain transfer	Continuous data transfer is possible using discontinuous addresses according to Linked List Item (LLI).
Transfer type	Transfer source -> Transfer destination	Peripheral device (Register) -> Memory Peripheral device (Register) -> Peripheral device (Register) Memory -> Peripheral device (Register) Memory -> Memory (start up by only software)
Transfer control	Transfer channel	2 channels (ch 0 and ch 1)
	Transfer address	Addresses of the transfer source and destination are set. Selection of the increment of the address or fixed address for either the transfer source or destination.
	Transfer data size	8 bits, 16 bits, or 32 bits The sizes of the transfer source and destination can be set independently.
	Priority	ch 0 > ch 1 (Fixed in the unit.) Unit A > Unit B (For the integrated units, refer to “Product Information”.)
	FIFO	4 Words x 2 ch (1 Word = 32 bits)
Transfer count	Transfer count	4095 at maximum Infinite count transfers can be done using LLI.
Endian	Little endian	-
Interrupt	Transfer completion interrupt	Transfer completion interrupt (INTHDMACxTC) is generated when a transfer completes.
	Error interrupt	Error interrupt (INTHDMACxERR) is generated when a bus error or a memory protection error is detected during data transfer.

9. Sample Program

The data received from the terminal software is transferred from a memory to another memory using the HDMAC.
The transferred data is output to the terminal software through the UART interface.

9.1. Initialization

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

9.2. Sample Program Main Operation

BSP initialization is done.
Driver initialization is done.
Application initialization is done.
HDMA setting is done.
The HDMA transfer end interrupt and the HDMA transfer error interrupt are enabled.
The message which requests a user to input appropriate data to the terminal software is issued.
The sample program waits for data transmitted by the terminal software.

When a character is input to the terminal software, the input data is received through the UART interface.
The received data is stored in a memory and it is copied to another memory. Then the data is output to the terminal software through the UART interface.
If the input character exceeds 32 byte (including a line feed code), an input error is output to the terminal software.

9.3. HDMA_MEM_MEM Setting

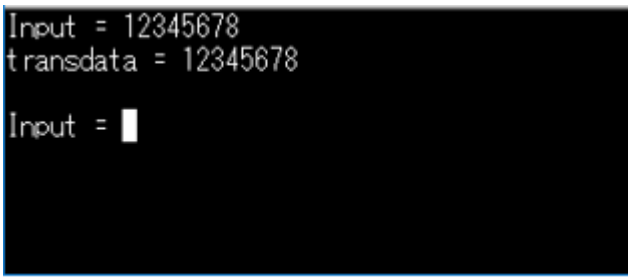
The transfer channel in the sample program can be changed using the following parameter;
CFG_DMA_CH in “main.c” should be modified.

The priority of the transfer control is ch 0 > ch 1.

When transferring a lot of data in "memory to memory", we recommend using a transfer channel with low priority. If you use a transfer channel with a higher priority, you can not start transferring a transfer channel with a lower priority until the transfer ends.

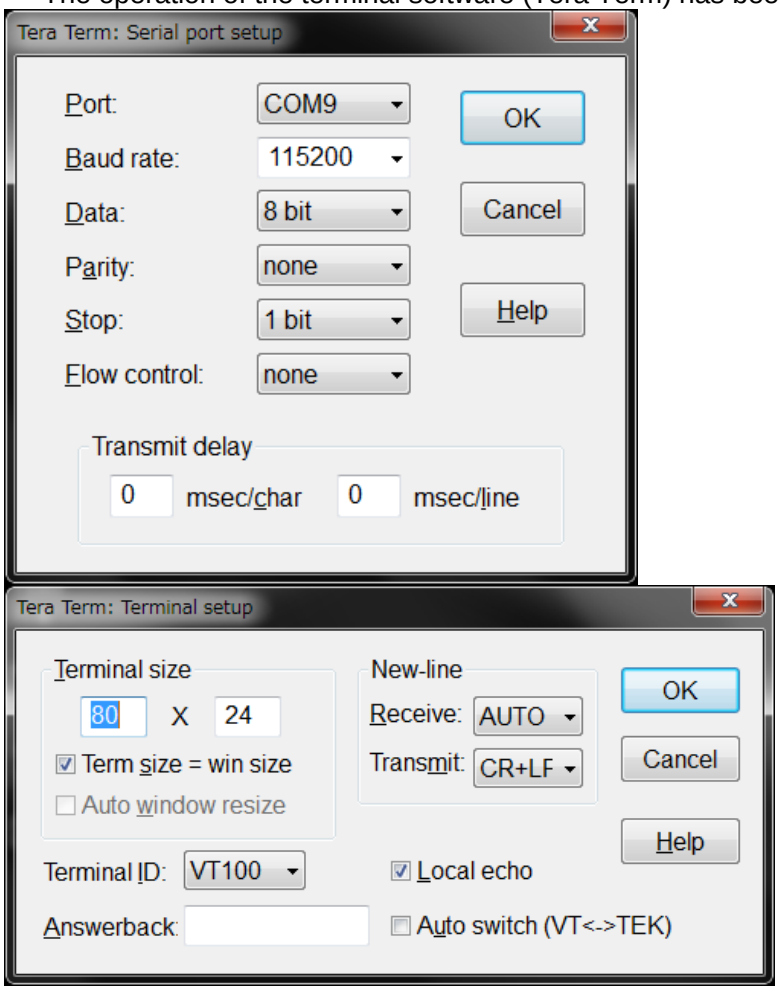
9.4. Output Example of Sample Program

When the sample program is executed, the transfer result done by the HDMAC is output shown as follows;



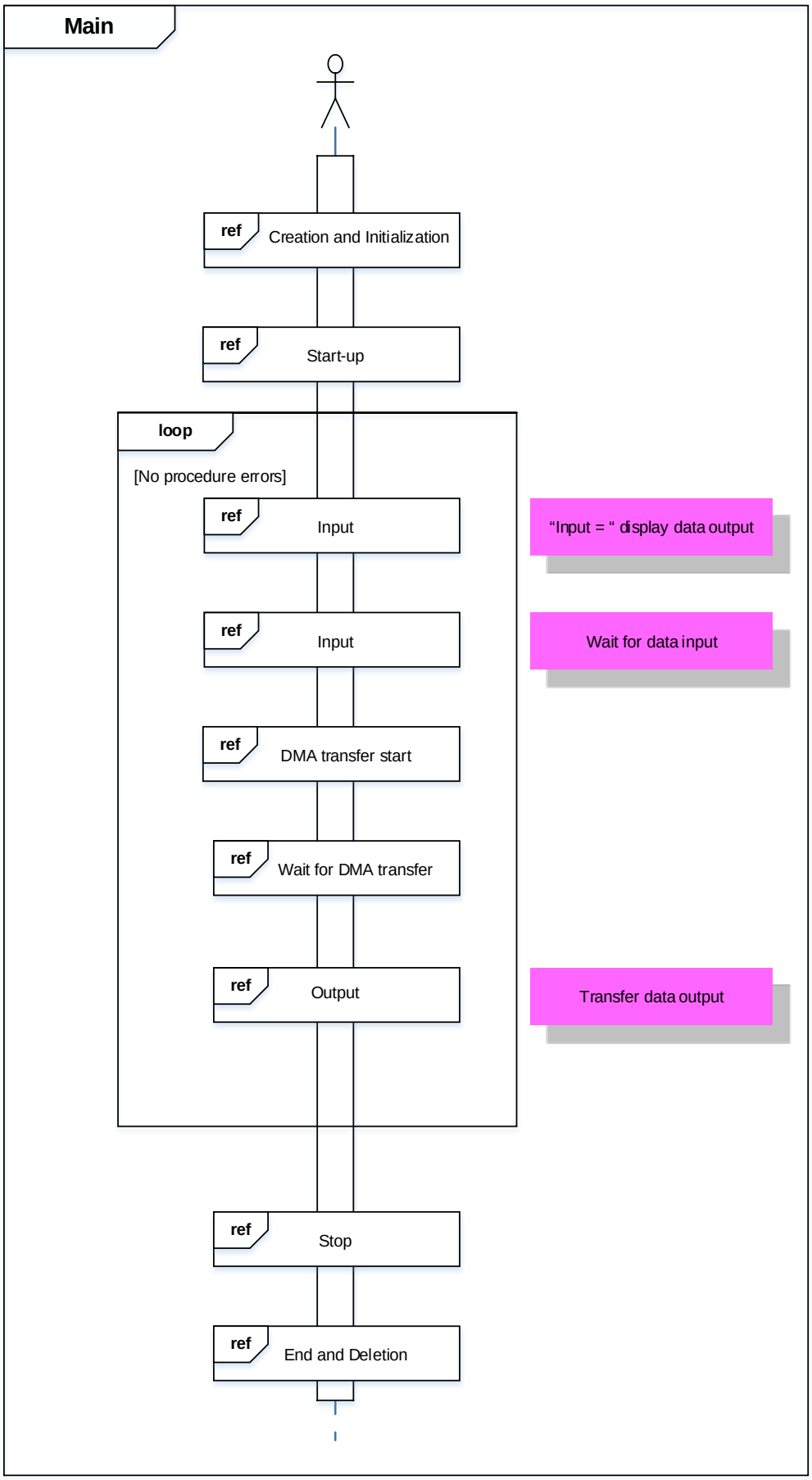
9.4.1. Setting Example of Terminal Software

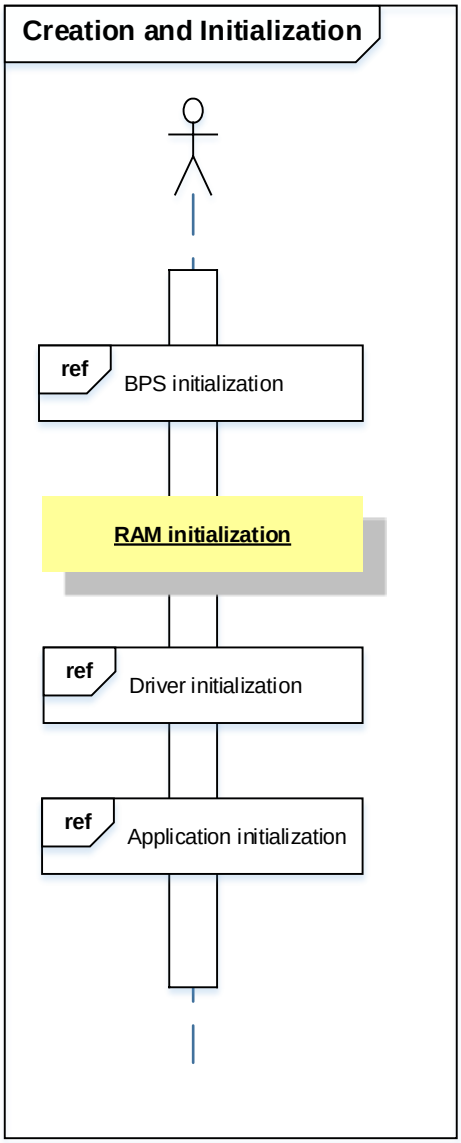
The operation of the terminal software (Tera Term) has been checked with the following settings.

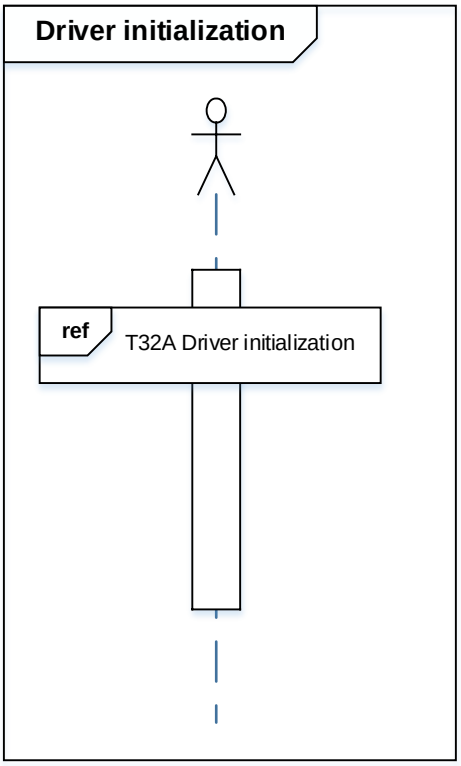


9.5. Operating Flow of Sample Program

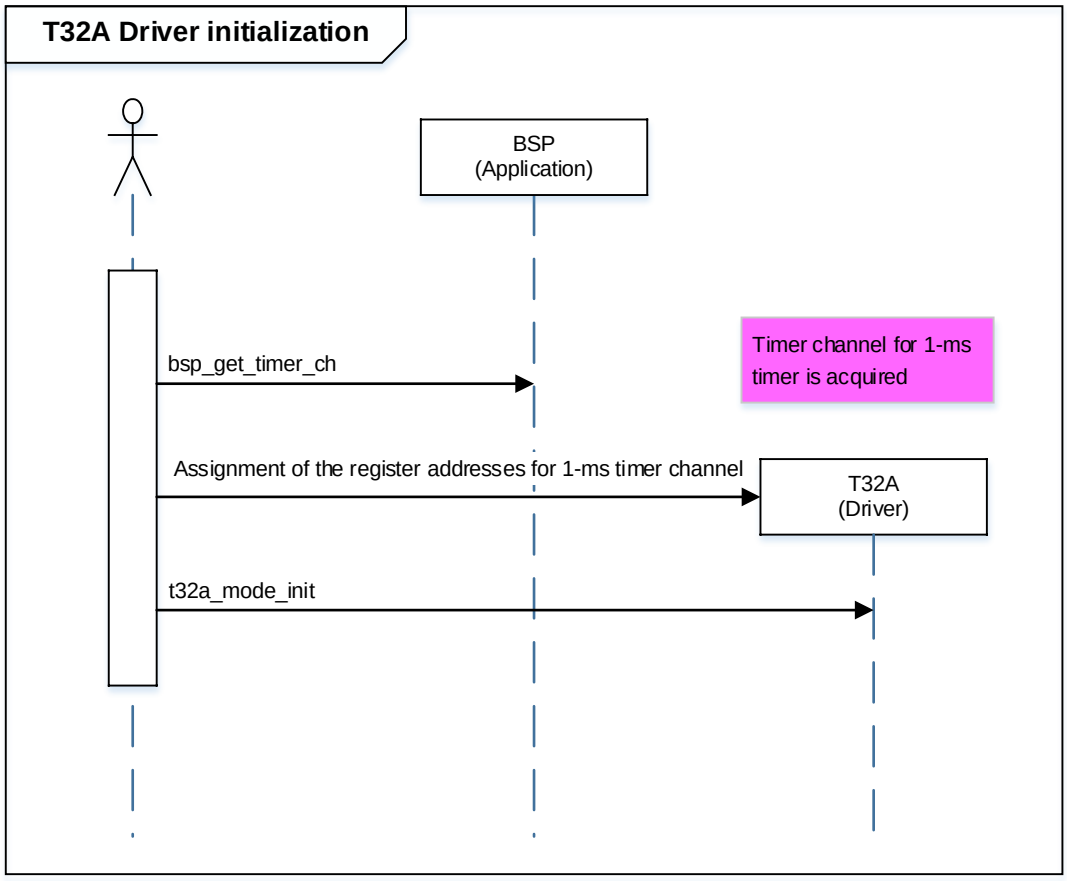
The basic operating flows of the sample program are shown in the following;

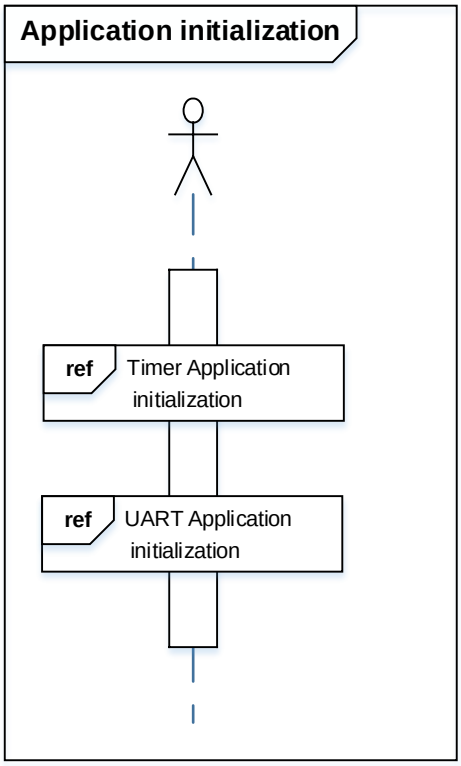


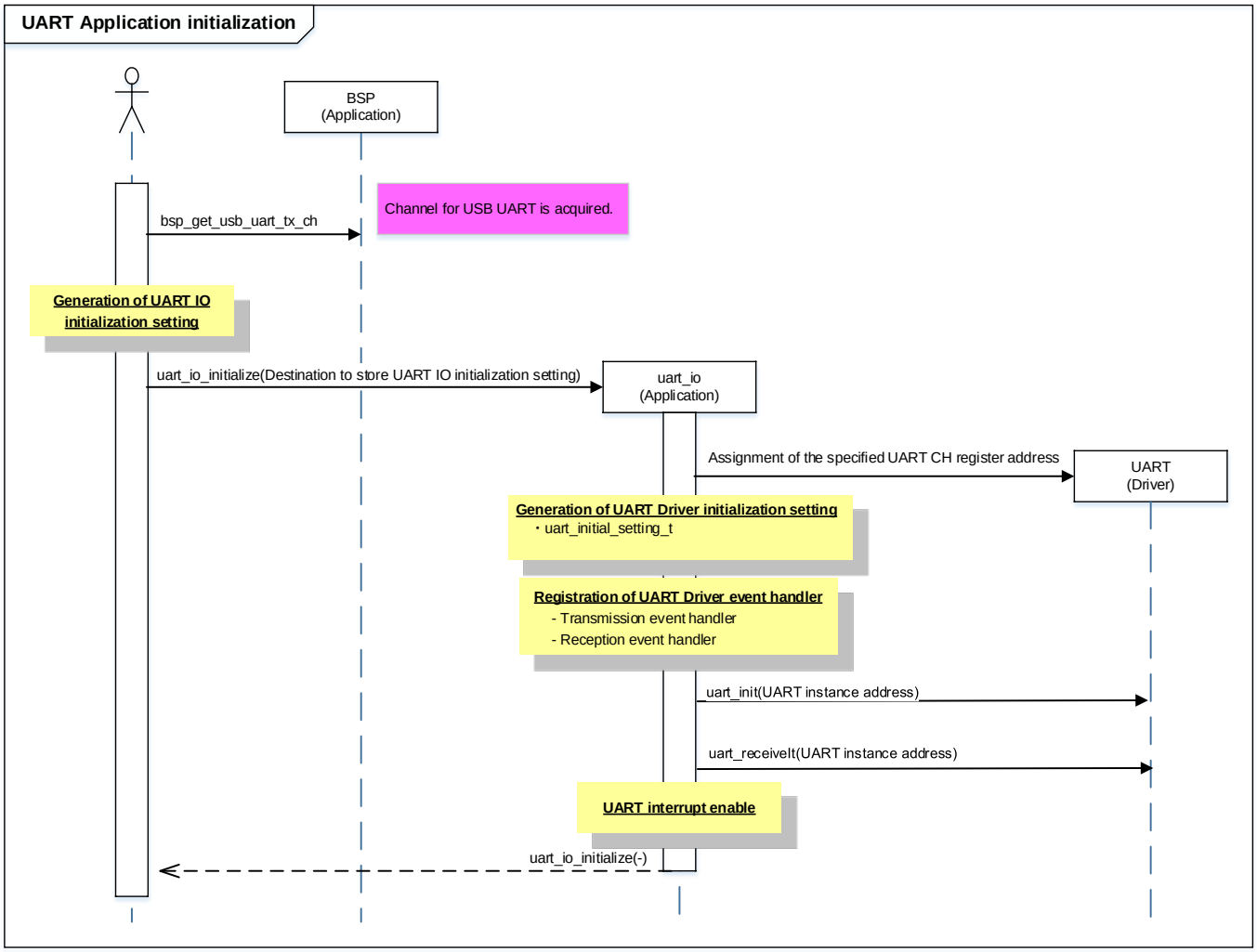

Creation and Initialization
ref
BPS initialization
RAM initialization
ref
Driver initialization
ref
Application initialization

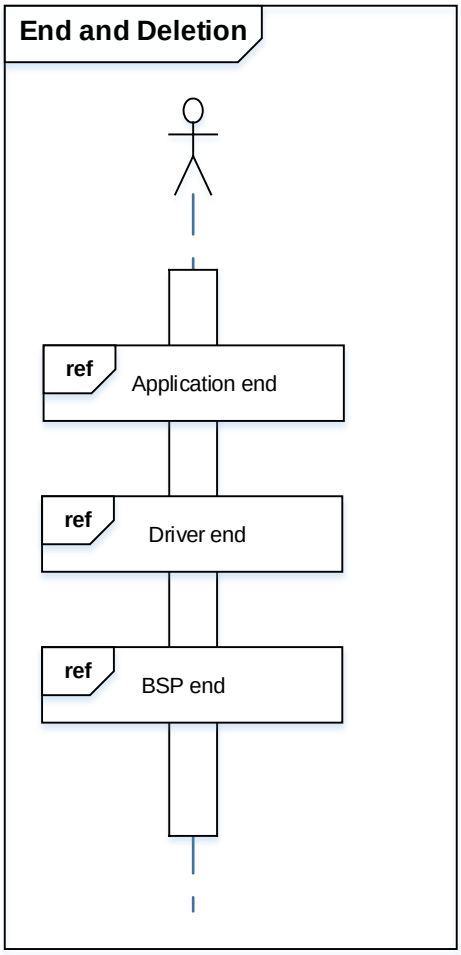


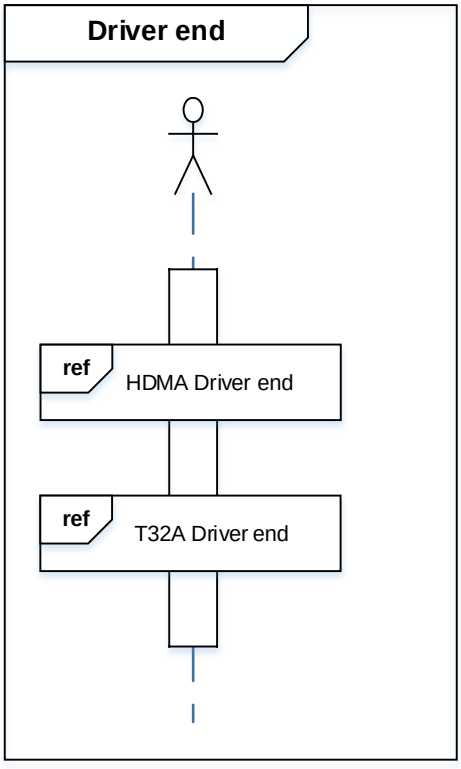
The 32-bit timer event counter of TMPM4G9 is running, but processing using timer count is not performed.

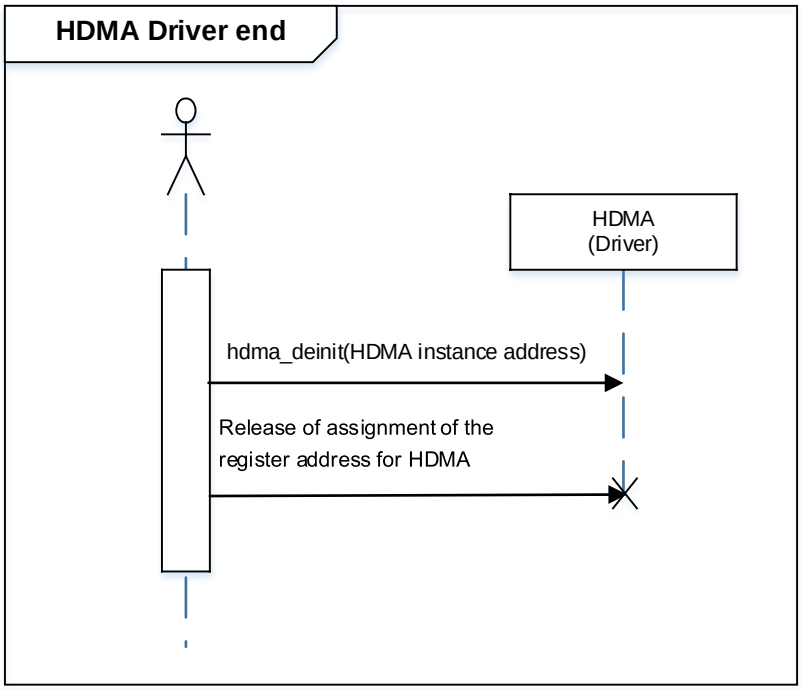


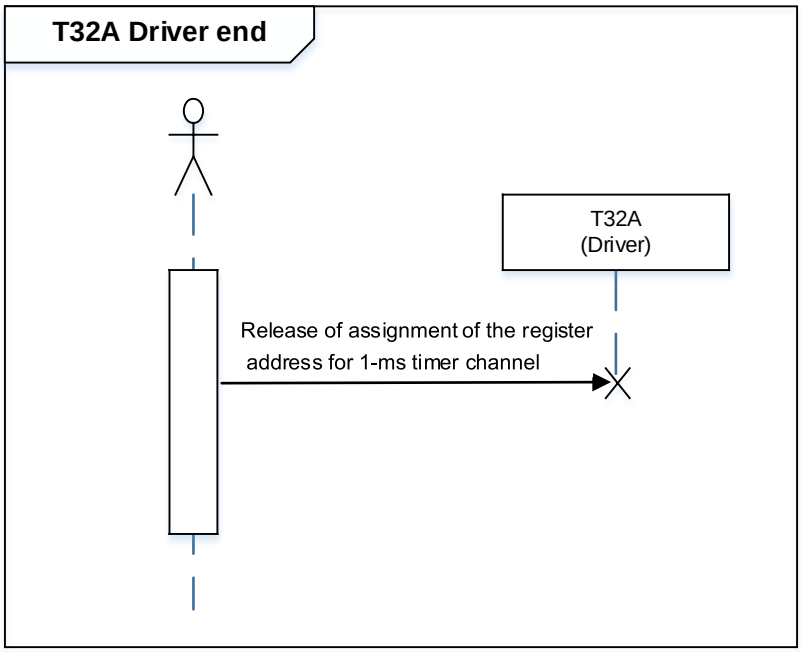

Application initialization
ref
Timer Application initialization
ref
UART Application initialization

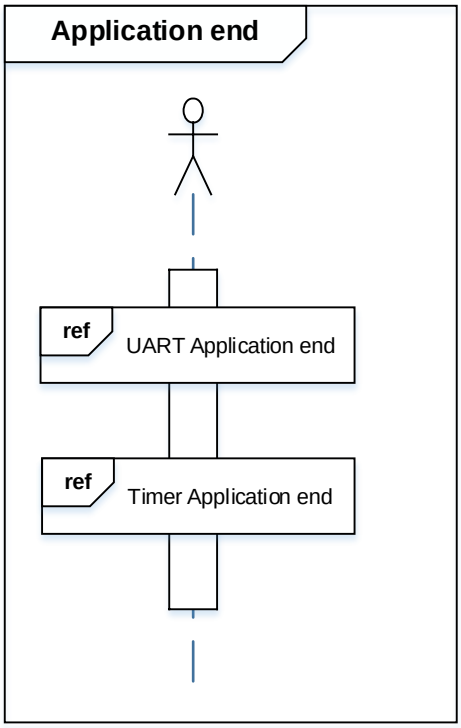

UART Application initialization
BSP (Application)
Channel for USB UART is acquired.
bsp_get_usb_uart_tx_ch
Generation of UART IO initialization setting
uart_io_initialize(Destination to store UART IO initialization setting)
uart_io (Application)
Assignment of the specified UART CH register address
UART (Driver)
Generation of UART Driver initialization setting
・uart_initial_setting_t
Registration of UART Driver event handler
- Transmission event handler
- Reception event handler
uart_init(UART instance address)
uart_receiveIt(UART instance address)
UART interrupt enable
uart_io_initialize(-)

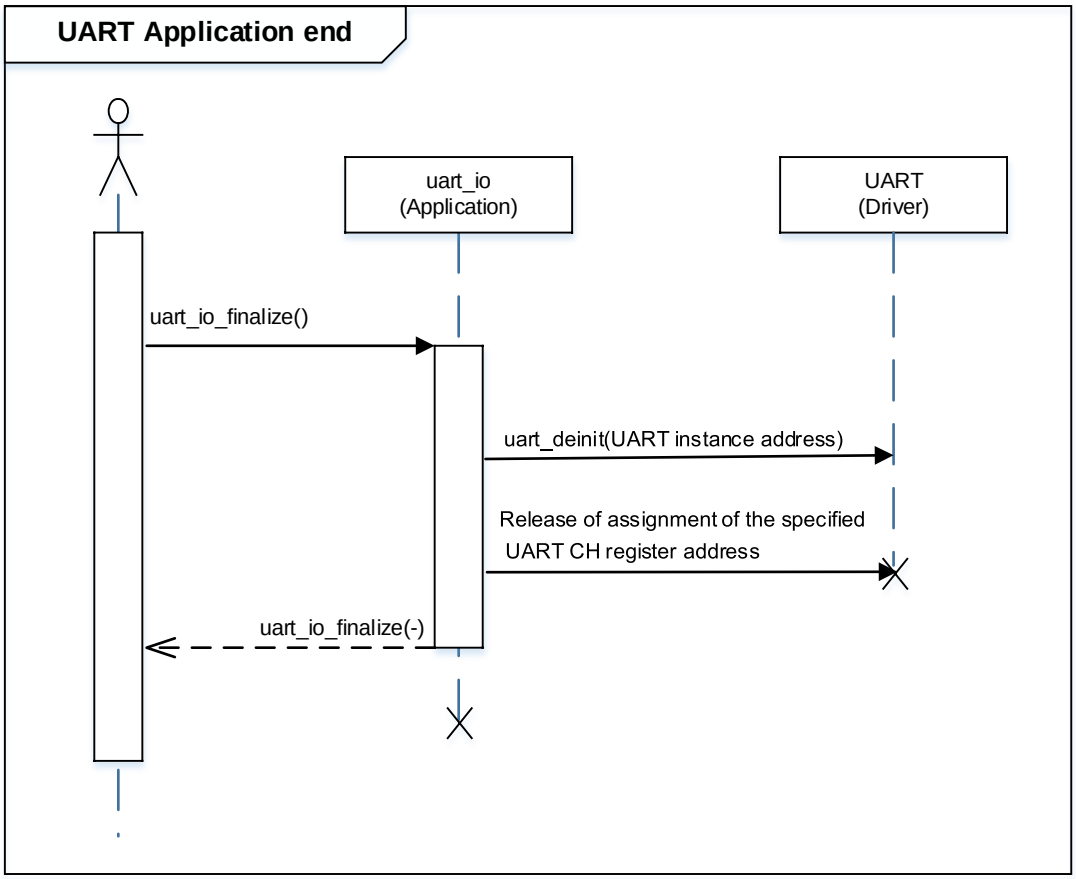

End and Deletion
ref
Application end
ref
Driver end
ref
BSP end

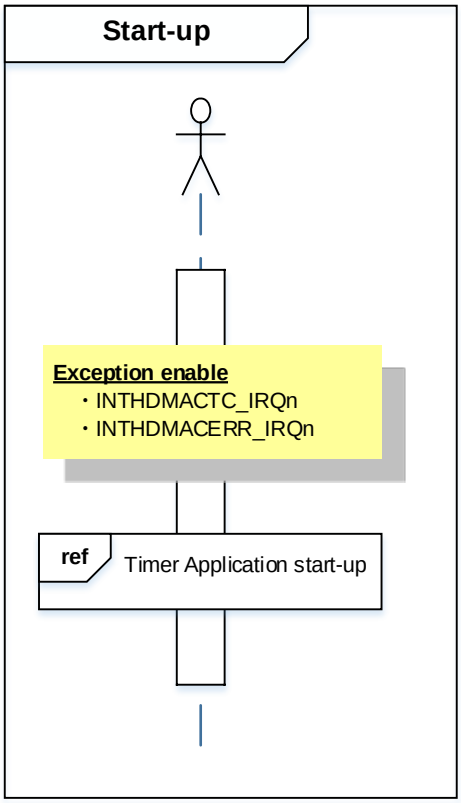

Driver end
ref
HDMA Driver end
ref
T32A Driver end

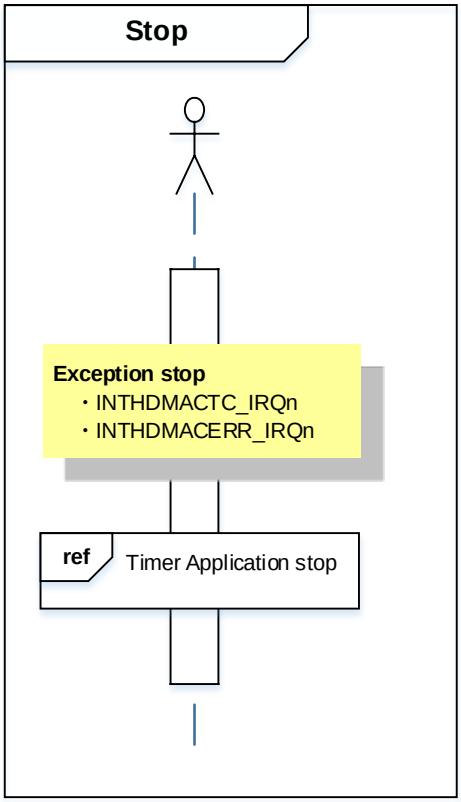

HDMA Driver end
HDMA
(Driver)
hdma_deinit(HDMA instance address)
Release of assignment of the
register address for HDMA

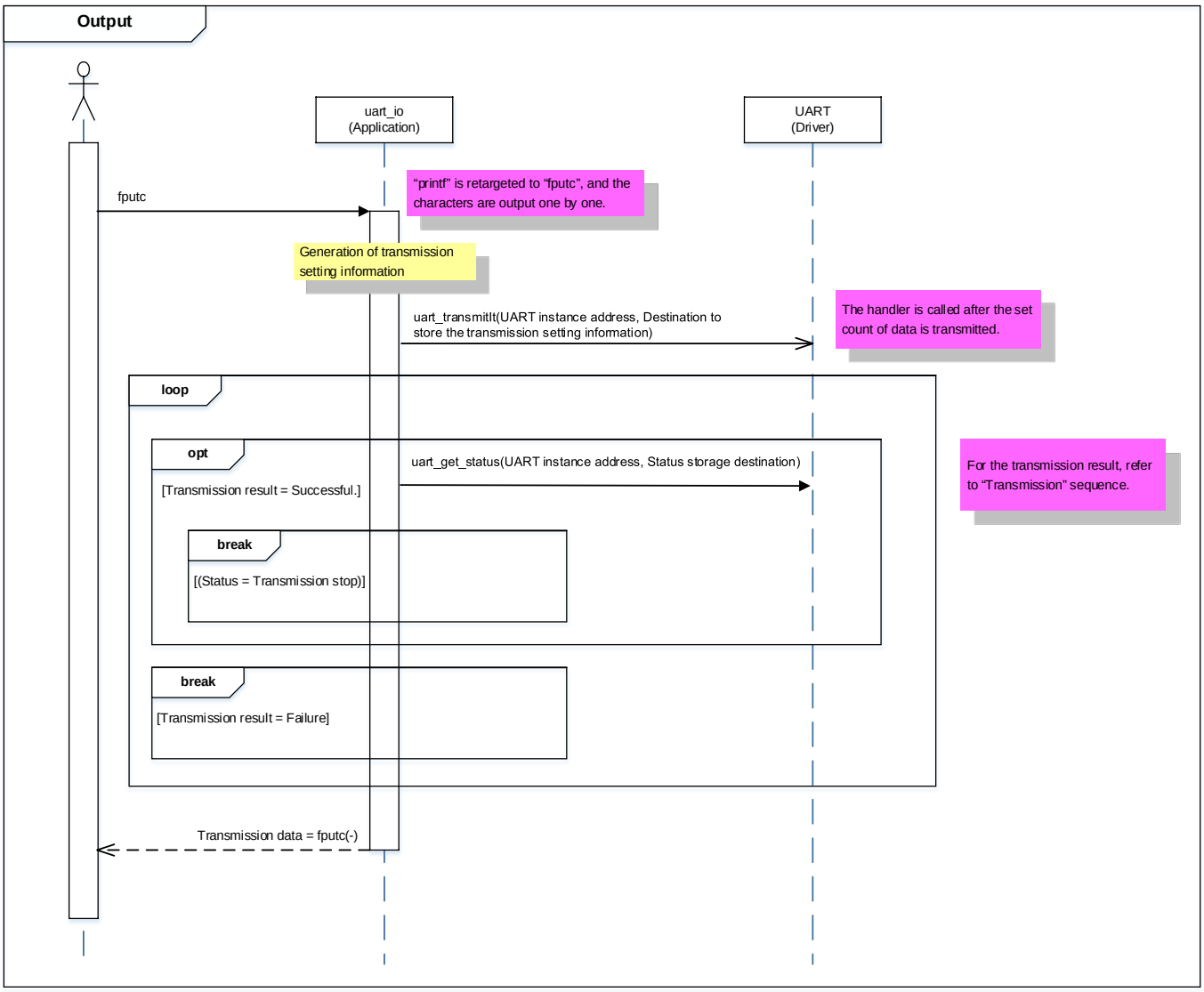

T32A Driver end
T32A
(Driver)
Release of assignment of the register
address for 1-ms timer channel

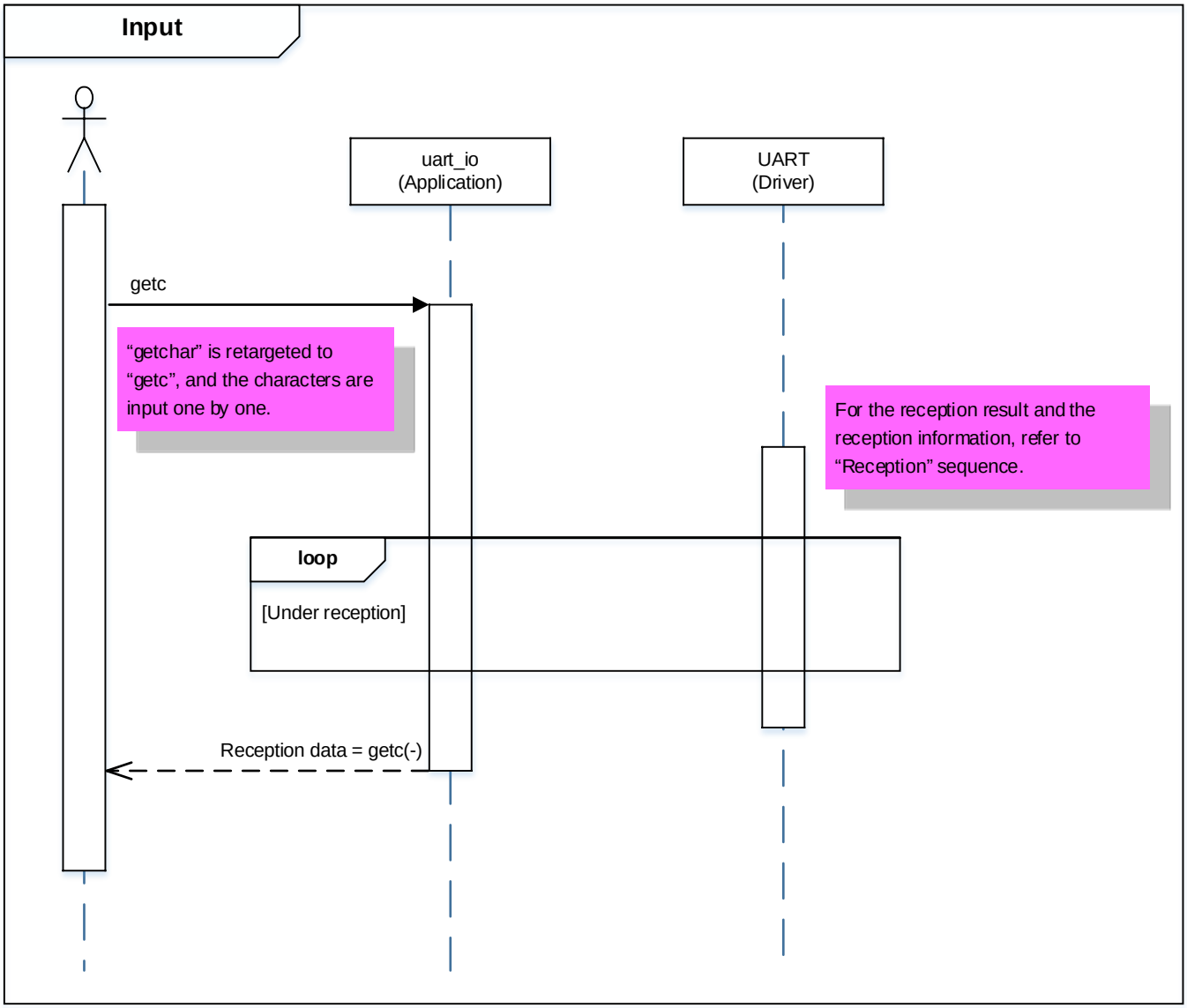

Application end
ref
UART Application end
ref
Timer Application end

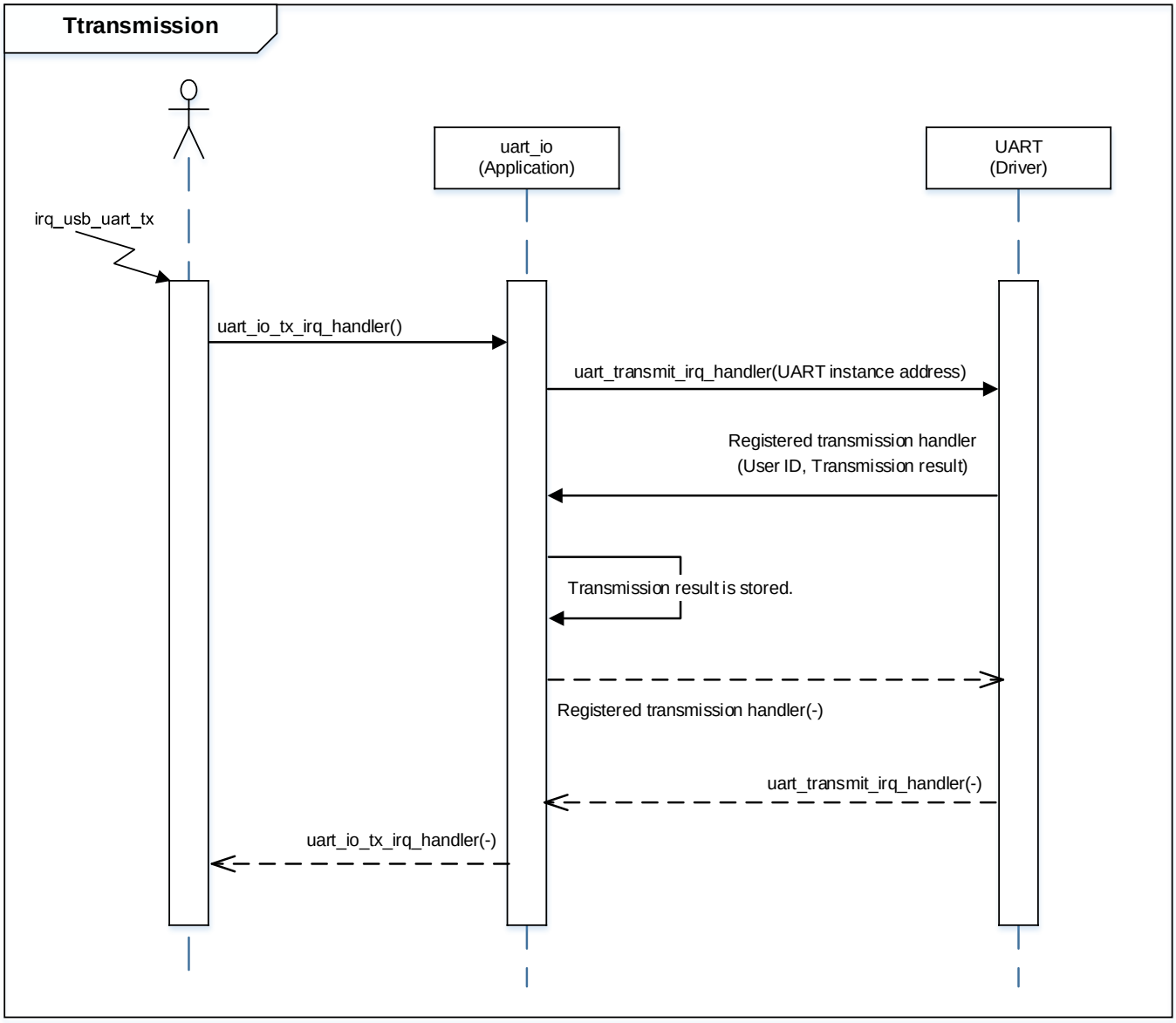

UART Application end
uart_io
(Application)
UART
(Driver)
uart_io_finalize()
uart_deinit(UART instance address)
Release of assignment of the specified
UART CH register address
uart_io_finalize(-)

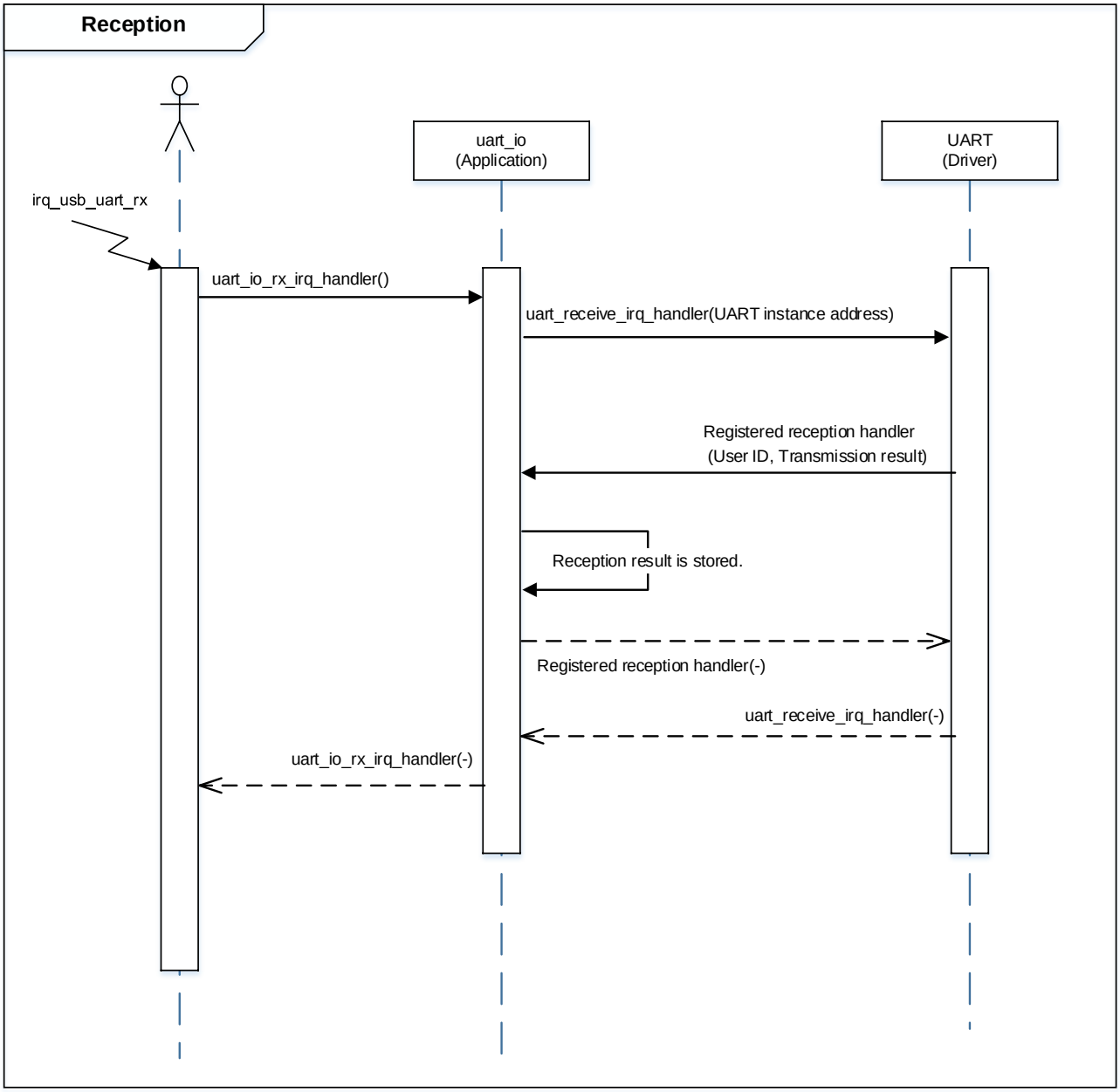

Start-up
Exception enable
・INTHDMACTC_IRQn
・INTHDMACERR_IRQn
ref
Timer Application start-up

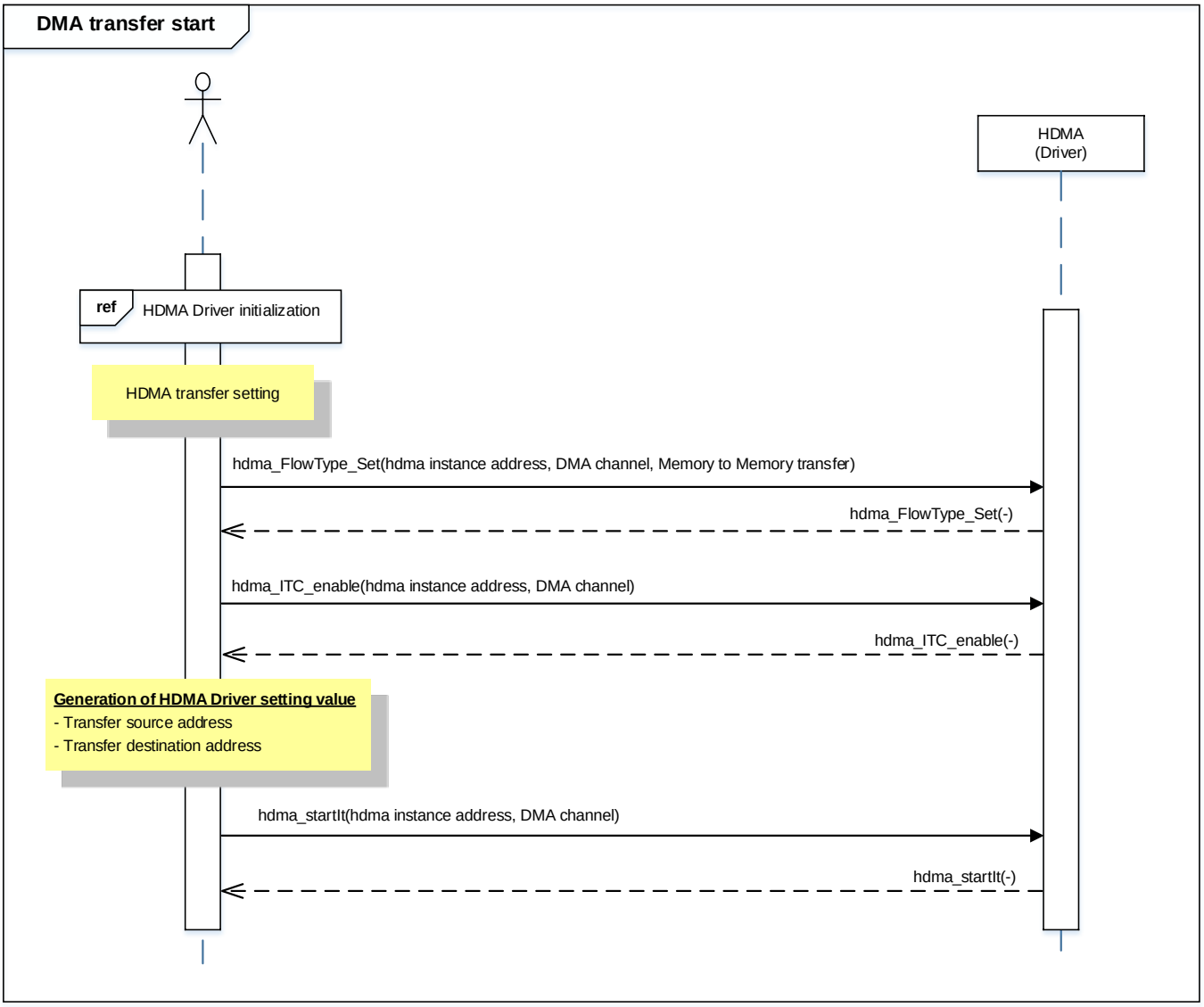

Stop
Exception stop
・INTHDMACTC_IRQn
・INTHDMACERR_IRQn
ref
Timer Application stop

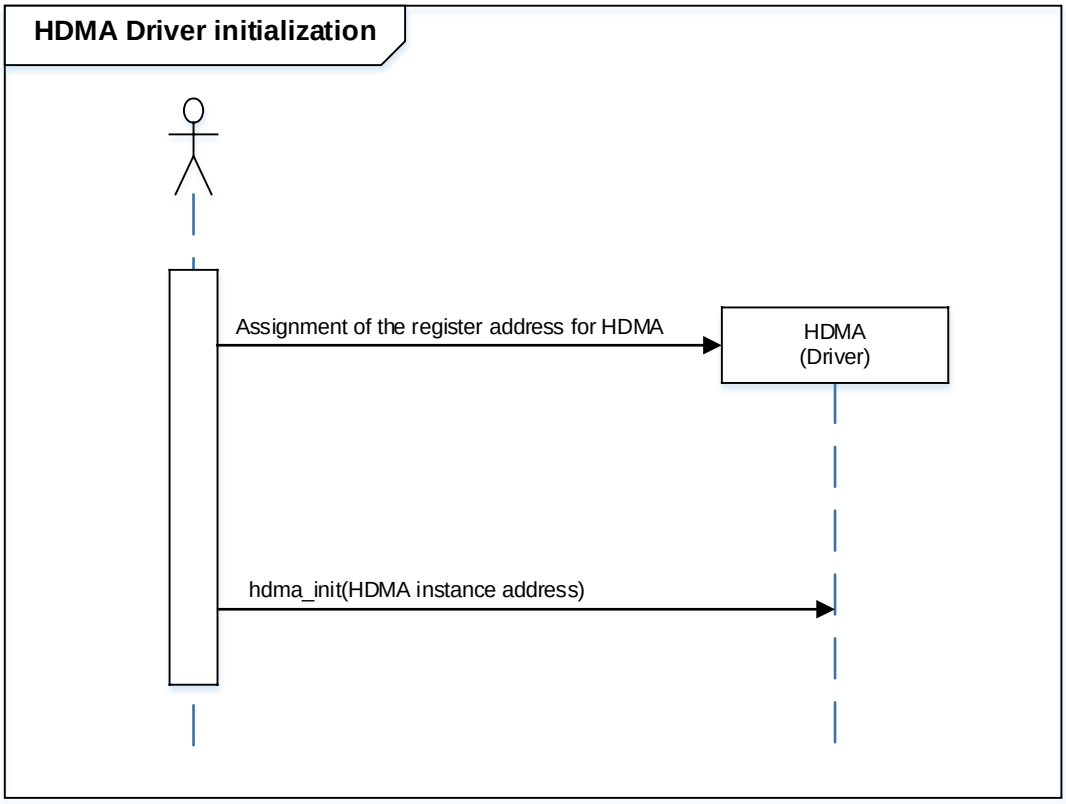

Output
uart_io
(Application)
UART
(Driver)
fputc
"printf" is retargeted to "fputc", and the characters are output one by one.
Generation of transmission setting information
uart_transmitIt(UART instance address, Destination to store the transmission setting information)
The handler is called after the set count of data is transmitted.
loop
opt
uart_get_status(UART instance address, Status storage destination)
[Transmission result = Successful.]
For the transmission result, refer to "Transmission" sequence.
break
[(Status = Transmission stop)]
break
[Transmission result = Failure]
Transmission data = fputc(-)

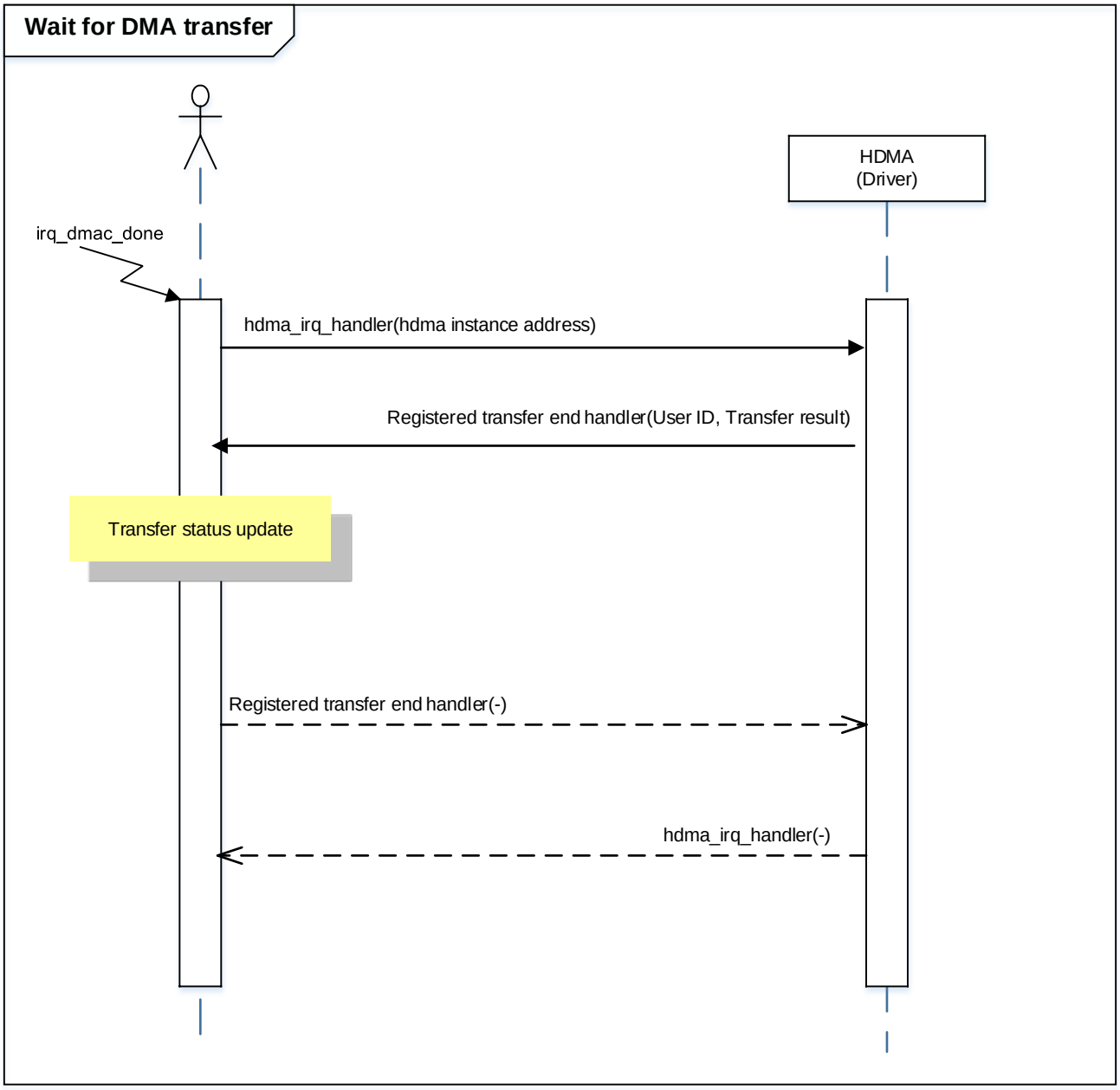

Input
uart_io
(Application)
UART
(Driver)
getc
“getchar” is retargeted to “getc”, and the characters are input one by one.
For the reception result and the reception information, refer to “Reception” sequence.
loop
[Under reception]
Reception data = getc(-)


Ttransmission
uart_io
(Application)
UART
(Driver)
irq_usb_uart_tx
uart_io_tx_irq_handler()
uart_transmit_irq_handler(UART instance address)
Registered transmission handler
(User ID, Transmission result)
Transmission result is stored.
Registered transmission handler(-)
uart_transmit_irq_handler(-)
uart_io_tx_irq_handler(-)


Reception
uart_io
(Application)
UART
(Driver)
irq_usb_uart_rx
uart_io_rx_irq_handler()
uart_receive_irq_handler(UART instance address)
Registered reception handler
(User ID, Transmission result)
Reception result is stored.
Registered reception handler(-)
uart_receive_irq_handler(-)
uart_io_rx_irq_handler(-)


DMA transfer start
HDMA
(Driver)
ref
HDMA Driver initialization
HDMA transfer setting
hdma_FlowType_Set(hdma instance address, DMA channel, Memory to Memory transfer)
hdma_FlowType_Set(-)
hdma_ITC_enable(hdma instance address, DMA channel)
hdma_ITC_enable(-)
Generation of HDMA Driver setting value
- Transfer source address
- Transfer destination address
hdma_startIt(hdma instance address, DMA channel)
hdma_startIt(-)


HDMA Driver initialization
Assignment of the register address for HDMA
HDMA
(Driver)
hdma_init(HDMA instance address)


Wait for DMA transfer
HDMA
(Driver)
irq_dmac_done
hdma_irq_handler(hdma instance address)
Registered transfer end handler(User ID, Transfer result)
Transfer status update
Registered transfer end handler(-)
hdma_irq_handler(-)

10. Precaution

When using the sample program with CPU other than TMPM4G9F15, please check operation sufficiently.

11. Revision History

Rev	Date	Description
1.0	2018-11-30	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.
- 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.**