Outlines

This application note is a reference material for developing products using the Master/Slave function in I2C interface (I2C) functions of M3H Group(2).

This document helps the user check operation of the product and develop its program.

Target sample program: I2C_MASTER_SLAVE
# Table of Contents

Outlines ................................................................................................................................. 1  
Table of Contents .................................................................................................................. 2  
1. Preface .................................................................................................................................. 3  
2. Reference Document .......................................................................................................... 4  
3. Function to Use .................................................................................................................. 4  
4. Target Device ..................................................................................................................... 4  
5. Conditions for Correct Operation ..................................................................................... 5  
6. Evaluation Board Setting .................................................................................................. 6  
   Evaluation Board B ............................................................................................................... 6  
7. Outline of I²C Interface function ....................................................................................... 8  
   7.1. Clock Supply ............................................................................................................... 8  
8. Sample Program ............................................................................................................... 9  
   8.1. Initialization ............................................................................................................... 9  
   8.2. Sample program main operation ............................................................................... 9  
   8.3. Output Example of Sample Program ......................................................................... 11  
      8.3.1. Setting Example of Terminal Software .............................................................. 13  
   8.4. Operating Flow of Sample Program ......................................................................... 14  
9. Precaution ......................................................................................................................... 20  
10. Revision History .............................................................................................................. 20  
REstrictions on product use ................................................................................................. 21
1. Preface

This sample program is used to check the operation of the I2C communication function. Data transmission and reception by I2C Master / Slave is performed by terminal software on the host PC via USB-UART.

Structure diagram of Sample program
2. Reference Document
- Datasheet
  TMPM3H group (2) datasheet Rev2.0 (Japanese edition)
- Reference manual
  I2C interface (I2C-B) Rev2.0 (Japanese edition)
- Other reference document
  TMPM3H(2) Group Peripheral Driver User Manual (Doxygen)

3. Function to Use

<table>
<thead>
<tr>
<th>IP</th>
<th>Channel</th>
<th>Port</th>
<th>Function / operation mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2C interface</td>
<td>ch1</td>
<td>PA4 (I2C1SCL)</td>
<td>I2C mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PA5 (I2C1SDA)</td>
<td></td>
</tr>
<tr>
<td>Asynchronous communication</td>
<td>ch0</td>
<td>PA1 (UT0TXDA)</td>
<td>UART mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PA2 (UT0RXD)</td>
<td></td>
</tr>
</tbody>
</table>

4. Target Device

The target devices of application note are as follows.

<table>
<thead>
<tr>
<th>TMPM3HQFDFG</th>
<th>TMPM3HQFZFG</th>
<th>TMPM3HQFYFG</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMPM3HPFDFG</td>
<td>TMPM3HPFZFG</td>
<td>TMPM3HPFYFG</td>
</tr>
<tr>
<td>TMPM3HNFDFG</td>
<td>TMPM3HNFZFG</td>
<td>TMPM3HNFYDFG</td>
</tr>
<tr>
<td>TMPM3HNFDDFG</td>
<td>TMPM3HNFZDFG</td>
<td>TMPM3HNFYDFG</td>
</tr>
<tr>
<td>TMPM3HMFDFG</td>
<td>TMPM3HMFZFG</td>
<td>TMPM3HMFYFG</td>
</tr>
</tbody>
</table>

* This sample program operates on the evaluation board of TMPM3HQFDFG. If other function than the TMPM3HQ 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 (TMPM3HQ). If other function than the TMPM3HQ one is checked, the BSP related file should be changed properly.
5. Conditions for Correct Operation

<table>
<thead>
<tr>
<th>Used microcontroller</th>
<th>TMPM3HQFDFG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used board</td>
<td>TMPM3HQFDFG Evaluation Board (Product of Sensyst)</td>
</tr>
<tr>
<td>Unified development environment</td>
<td>IAR Embedded Workbench for ARM 8.11.2.13606</td>
</tr>
<tr>
<td>Unified development environment</td>
<td>μVision MDK Version 5.24.2.0</td>
</tr>
<tr>
<td>Terminal software</td>
<td>Tera Term V4.96</td>
</tr>
<tr>
<td>Sample program</td>
<td>V1100</td>
</tr>
</tbody>
</table>

Evaluation board (TMPM3HQFDFG Evaluation Board) (Top view)

For purchasing the board, refer to the following homepage. ([http://www.chip1stop.com/](http://www.chip1stop.com/))
6. Evaluation Board Setting

Two evaluation boards should be prepared.

Prepare board
Evaluation Board A: TMPM3HQFDFG Evaluation Board
Evaluation Board B: TMPM3HQFDFG Evaluation Board

The following connections should be done.
Note 1: Evaluation Board A
Write the program created with the Default setting of the project for TMPM3HQ.
Connection method details

<table>
<thead>
<tr>
<th>Board function</th>
<th>Through hole No.</th>
<th>Through hole No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB UART conversion</td>
<td>53 : PA1</td>
<td>54 : USB_TXD</td>
</tr>
<tr>
<td>USB UART conversion</td>
<td>55 : PA2</td>
<td>56 : USB_RXD</td>
</tr>
</tbody>
</table>

Connection between CN in board A
Connect “CN5 No.45” and “CN12 No.57”
Connect “CN5 No.47” and “CN12 No.59”

Note 2: Evaluation Board B
Write the program created with the Default setting of the project for TMPM3HQ.
Connection method details

<table>
<thead>
<tr>
<th>Board function</th>
<th>Through hole No.</th>
<th>Through hole No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB UART conversion</td>
<td>53 : PA1</td>
<td>54 : USB_TXD</td>
</tr>
<tr>
<td>USB UART conversion</td>
<td>55 : PA2</td>
<td>56 : USB_RXD</td>
</tr>
</tbody>
</table>

Connections between the boards
Connect “Board A CN12 No57” and “Board B CN12 No57”
Connect “Board A CN12 No59” and “Board B CN12 No59”
7. Outline of I²C Interface function

The I²C can operate as a transceiver circuit of 1ch (SCL, SDA) in 1 unit circuit.

7.1. Clock Supply

When using I²C, please set a clock enabling bit corresponding with the fsys supply on/off register A ([CGFSYSENA]) or B ([CGFSYSENB]) and fc supply on/off register ([CGFCEN]) as “1” (clock supply). Please refer to “Clock Control and Operation Mode” of the reference manual for the details.
8. Sample Program

According to the command input in the terminal software, Master and Slave are operated.

8.1. Initialization

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

8.2. Sample program main operation

After the initialization operation, shift to the main function and perform the following initialization.
1 Initialization of BSP (Board Support Package)
2 Initialize the application
3 Configuration processing
4 Clear slave receive data buffer
5 Clear slave transmit data buffer
6 Clear master receive data buffer
7 Clear master transmit data buffer
8 Initialization of I2C setting value

After the above processing, "command>" on PC terminal software (Tera Term) is displayed.
By entering characters according to the command format, the microcontroller can operate I2C Master or I2C Slave.
From Master to Slave can be switched by command, command can be input when I2C is Master mode.

Command format:
"Command [ parameter]"
Parameters vary depending on command

Command list

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Parameter (hex)</th>
<th>Input example</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>write</td>
<td>Data transmission</td>
<td>XX .... XX ....</td>
<td>&quot;write&quot; &quot;write B0&quot; &quot;write B011223344&quot;</td>
<td>The transmission size is the total number of bytes of data_form1 + data_form2.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[slave_address] [master_tx_data]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>read</td>
<td>Data transmission + data reception</td>
<td>XX .... XX ....</td>
<td>&quot;read&quot; &quot;read B0&quot; &quot;read B05566&quot;</td>
<td>Send and receive size Transmission: Number of bytes of data_form1 Received: Number of bytes of data_form 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[slave_address] [master_tx_data]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>slave</td>
<td>Switch to Slave Mode</td>
<td>XX</td>
<td>-</td>
<td>&quot;slave&quot; &quot;slave B0&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[slave_address]</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

Note 1) "XX" is a hexadecimal number. In the case of 0x12, enter "12".
Note 2) Accept parameters of [] without specification.
8.3. Output Example of Sample Program

When the sample program operates, the command results are shown as follows;

[Log basic information]
"I2C1" indicates the channel used by I2C.
"sa B0" indicates the Slave Address waiting for reception.
"tx[]" indicates transmission data, and "rx[]" indicates reception data.

MASTER output log example

<MASTER output log example 1-1>

```
I2C TEST - I2C1
------------
| I2C master mode |
------------
command >
```

<MASTER output log example 1-2>

```
command > write
master
sa    B0
tx[0] 00
tx[1] 01
tx[2] 02
tx[3] 03
command >
```

<MASTER output log example 1-3>

```
command > read
master
sa    B0
tx[0] 00
tx[1] 01
rx[0] 80
rx[1] 81
command >
```
SLAVE output log example

<SLAVE output log example 1-1>

command > slave
-------------
| I2C slave mode |
-------------
slave
sa    B0

<SLAVE output log example 1-2>

slave
sa    B0
rx[0] 00
rx[1] 01
rx[2] 02
rx[3] 03
slave
sa    B0

<SLAVE output log example 1-3>

slave
sa    B0
rx[0] 00
rx[1] 01
tx[0] 80
tx[1] 81
slave
sa    B0
8.3.1. Setting Example of Terminal Software

The operation of the terminal software (Tera Term) has been checked with the following settings.
8.4. Operating Flow of Sample Program

The operating flows of the sample program are shown in the following:
Creation and Initialization

EEPROM_i2c_init()

Creation

i2c_init(Instance address, sda, scl)

Port:
I2C1SCL and I2C1SDA settings

Register assignment

I2C_init(Instance address)

i2c_reset()

i2c_frequency(100KHz)

I2C initial setting

result = i2c_init() Successful

I2C_get_clock_setting(Instance address)

frequency = I2C_get_clock_setting()

result = I2C_init() Successful
"write" command procedure

In the case of "write":

result = strncmp()

```
i2c_api
  main
```

```
to_i2c_master_non_blocking_write()

ref i2c_transfer_asynch
  Instance handle, Transmission buffer address, Transmission Byte count,
  Reception buffer address, Reception Byte count, Slave address, stop = 1
```

```
ref i2c_stop
```

```
Terminal display
Write data is displayed.
```

```
ref i2c_stop
[stop=1]
```

```
Main
```

```
Terminal display
Write data is displayed.
```
"slave" command procedure

- `ts_i2c_set_slave_mode()`
- `i2c_slave_address()` (Instance handle, Slave address)
- `i2c_slave_mode()` (Instance handle, Slave mode)
- `ts_i2c_slave_non_blocking_receive()`

In the case of "slave";

- `result strncmp()`
- `i2c_stop()` [stop=1]
- `i2c_slave_transfer_asynch()` (Instance handle, Transmission buffer address, Transmission Byte count, Reception buffer address, Reception Byte count, Slave address, stop = 1)
- `ref` `i2c_slave_init()` (Instance handle)

- `opt` [stop=1]
- `ref` `i2c_stop`

Terminal display
Read data is displayed.
i2c_api

i2c_transfer_asynch
(Instance handle, Transmission buffer address, Transmission Byte count, Reception buffer address, Reception Byte count, Slave address, stop = 1)

ACK result = i2c_transfer_asynch(); I2C_ACK

I2C_interrupt_request is present.

I2C_start_condition
(Instance handle, Slave address)

I2C_start_condition(-):

I2C_clear_int_status
(Instance handle)

I2C_clear_int_status(-):

I2C_int_status
(Instance handle)

I2C_int_status(-):

ACK result = I2C_get_ack() ACK

I2C_get_ack
(Instance handle)
9. Precaution

When using the sample program with CPU other than TMPM3HQ, please check operation sufficiently.

10. Revision History

<table>
<thead>
<tr>
<th>Rev</th>
<th>Date</th>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2018-05-17</td>
<td>-</td>
<td>First release</td>
</tr>
</tbody>
</table>
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, 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, devices related to electric power, and equipment used in finance-related fields. IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT. For details, please contact your TOSHIBA sales representative.

- 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.

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION