

ZMID Communication Board (ZMID-COMBOARD)

Serial Communication and Commands

**Contents**

<b>1. Introduction</b> .....	<b>3</b>
1.1 Requirements for User's Computer .....	3
1.2 Driver Installation .....	3
1.3 Communication Basics .....	4
1.3.1. Examples.....	4
<b>2. Commands</b> .....	<b>5</b>
2.1 General Commands .....	5
2.2 Power and Trigger Commands.....	6
2.3 Communication Commands .....	7
2.4 Commands for Reading the Output.....	8
2.5 Pin State Commands.....	9
<b>3. Examples</b> .....	<b>10</b>
<b>4. Glossary</b> .....	<b>14</b>
<b>5. Revision History</b> .....	<b>14</b>

**Figures**

Figure 1. Location of the Driver Installer.....	3
Figure 2. The ZMID-COMBOARD Appears as a Serial Device .....	3
Figure 3. Trigger Command Timing Diagram .....	6

**Tables**

Table 1. Commands List.....	5
Table 2. Version Command: V.....	5
Table 3. Hardware Revision Command: V_HW .....	5
Table 4. Supported Interfaces Command: V_FW.....	5
Table 5. Module Select Command: MS .....	6
Table 6. Power and On-Delay Trigger Setup Command: T .....	6
Table 7. Off-Delay Trigger Setup Command: T_.....	7
Table 8. OWI WRITE with Trigger Command: OWT .....	7
Table 9. OWI WRITE Command: OW_.....	7
Table 10. OWI READ Command: OR_ .....	7
Table 11. OWI READ Continuous Command: ORS.....	8
Table 12. OWI READ Special STOP Command: ORSX.....	8
Table 13. Set Output Interpretation Command: TSO .....	8
Table 14. Read Output Command: MRO .....	8

Table 15. Read Last SENT Frame Command: MRS .....	9
Table 16. Pin State Command: PS_.....	9
Table 17. Connecting and Reading EEPROM and SWR Memory (Device 1) .....	10
Table 18. Writing to the First 7 Registers in EEPROM (Device 1) .....	11
Table 19. Bulk Writing to the First 7 Registers in EEPROM (Device 1) .....	11
Table 20. Reading 3 Analog Output Samples from Device 1 .....	11
Table 21. Reading SENT Frames from Device 1 .....	12
Table 22. Enter Command Mode on Device 2 .....	12
Table 23. Reading PWM Output from Device 1 and Device 2 .....	13

# 1. Introduction

This document describes how to setup and use the serial communication capabilities of Renesas' USB Communication and Programming Board (ZMID-COMBOARD) for ZMID Application Modules in order to provide an interface between the user's computer and the Renesas ZMID4200 that is the device-under-test (DUT) on the module.

## 1.1 Requirements for User's Computer

- Windows® XP, Vista SP1 or later, 7 (including SP1), 8, 8.1, or 10
- Available USB port

## 1.2 Driver Installation

The driver required for serial communication is automatically installed on operating systems newer than Windows® 8. For older operating systems, the driver must be manually installed.

To manually install the driver, follow these steps:

1. Connect the ZMID-COMBOARD to an available USB port on the user's computer via the micro-USB cable.
2. The board will appear as two removable storage devices named EVKIT-1 and EVKIT-2. Open either one of them.
3. Locate the *LPC-VCOM.INF* file and open the menu by clicking with the right mouse button over it.

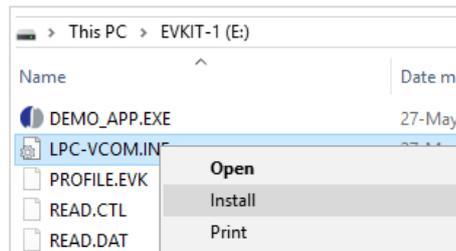


Figure 1. Location of the Driver Installer

4. Select the "Install" option and complete the install setup.
5. After successful driver installation, the device appears in the Device Manager under "Ports" as a "USB Serial Device." See Figure 2.

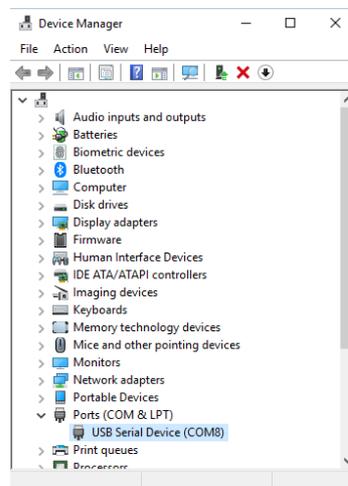


Figure 2. The ZMID-COMBOARD Appears as a Serial Device

## 1.3 Communication Basics

The computer communicates with the ZMID-COMBOARD through a virtual COM port (VCOM). The commands and responses can be interpreted as ASCII characters. The computer is the “master” in the communication – it sends a command and the ZMID-COMBOARD always returns a response.

Default COM port settings:

- Port Number: Check the Device Manager; the port number is assigned by the operating system and can vary
- Baud Rate: 19200
- Data Bits: 8
- Stop Bits: 1
- Parity Bits: No
- Flow Control: No

Format of the commands and responses:

Both commands and responses end with a carriage return and a line feed character: “\r\n” which corresponds to the ASCII bytes 0D<sub>HEX</sub> and 0A<sub>HEX</sub>.

The first byte of the response is a status byte which can be either an Acknowledge (06<sub>HEX</sub> in ASCII) or Not Acknowledge (15<sub>HEX</sub> in ASCII). These responses are represented as <ACK> and <NACK> in this document. Depending on the command, a response can have only a status byte or it can be followed by a number of data bytes.

The commands and response are case-insensitive.

Errors are returned as responses that start with a Not Acknowledge byte and can have optional error code bytes.

### 1.3.1. Examples

**Command:** OR\_E2

**Description:** OWI Read with command address E2<sub>HEX</sub>

<b>Characters</b>	O	R	_	E	2	\r (Carriage Return)	\n (Line Feed)
<b>Bytes</b>	72 <sub>HEX</sub>	52 <sub>HEX</sub>	5F <sub>HEX</sub>	45 <sub>HEX</sub>	32 <sub>HEX</sub>	0D <sub>HEX</sub>	0A <sub>HEX</sub>

**Reply:** <ACK>1C3F

**Description:** Acknowledge byte and data: 1C3F<sub>HEX</sub>

<b>Characters</b>	<ACK>	1	C	3	F	\r (Carriage Return)	\n (Line Feed)
<b>Bytes</b>	06 <sub>HEX</sub>	31 <sub>HEX</sub>	43 <sub>HEX</sub>	33 <sub>HEX</sub>	46 <sub>HEX</sub>	0D <sub>HEX</sub>	0A <sub>HEX</sub>

**Command:** T00000

**Description:** Turn off the power for the DUT

<b>Characters</b>	T	0	0	0	0	0	\r (Carriage Return)	\n (Line Feed)
<b>Bytes</b>	54 <sub>HEX</sub>	30 <sub>HEX</sub>	0D <sub>HEX</sub>	0A <sub>HEX</sub>				

**Reply:** <ACK>

**Description:** Acknowledge without extra data bytes

<b>Characters</b>	<ACK>	\r (Carriage Return)	\n (Line Feed)
<b>Bytes</b>	06 <sub>HEX</sub>	0D <sub>HEX</sub>	0A <sub>HEX</sub>

## 2. Commands

**Table 1. Commands List**

Command	Action
<b>General Commands</b>	
V	Returns the firmware version information (see Table 2 for details)
V_HW	Returns the hardware revision information (see Table 3 for details)
V_FW	Returns the supported interfaces information (see Table 4 for details)
MS	Selects the active module (device) for communication and output reading (see Table 5 for details)
<b>Power and Trigger Commands</b>	
T	Device under test (DUT) power control and power-on delay trigger setup (see Table 6 for details)
T_	Power-off delay trigger setup (see Table 7 for details)
<b>Communication Commands</b>	
OWT	OWI WRITE with trigger (see Table 8 for details)
OW_	OWI WRITE (see Table 9 for details)
OR_	OWI READ (see Table 10 for details)
ORS	OWI READ continuous (see Table 11 for details)
ORSX	OWI stop continuous READ (see Table 12 for details)
<b>Commands for Reading the Output</b>	
TSO	Output interpretation setup (see Table 13 for details)
MRO	Read last measured output (see Table 14 for details)
MRS	Read last SENT frame (see Table 15 for details)
<b>Pin State Commands</b>	
PS_	Set pin state (see Table 16 for details)

### 2.1 General Commands

**Table 2. Version Command: V**

<b>Command</b>	V	
<b>Description</b>	Returns a string with the firmware version of the ZMID-COMBOARD.	
<b>Syntax</b>	V	
<b>Example</b>	Send	V
	Response	<ACK>ZMID COM BOARD FW_00.05.1309

**Table 3. Hardware Revision Command: V\_HW**

<b>Command</b>	V_HW	
<b>Description</b>	Returns a string with the recognized main hardware revision of the ZMID-COMBOARD.	
<b>Syntax</b>	V_HW	
<b>Example</b>	Send	V_HW
	Response	<ACK>R5.1

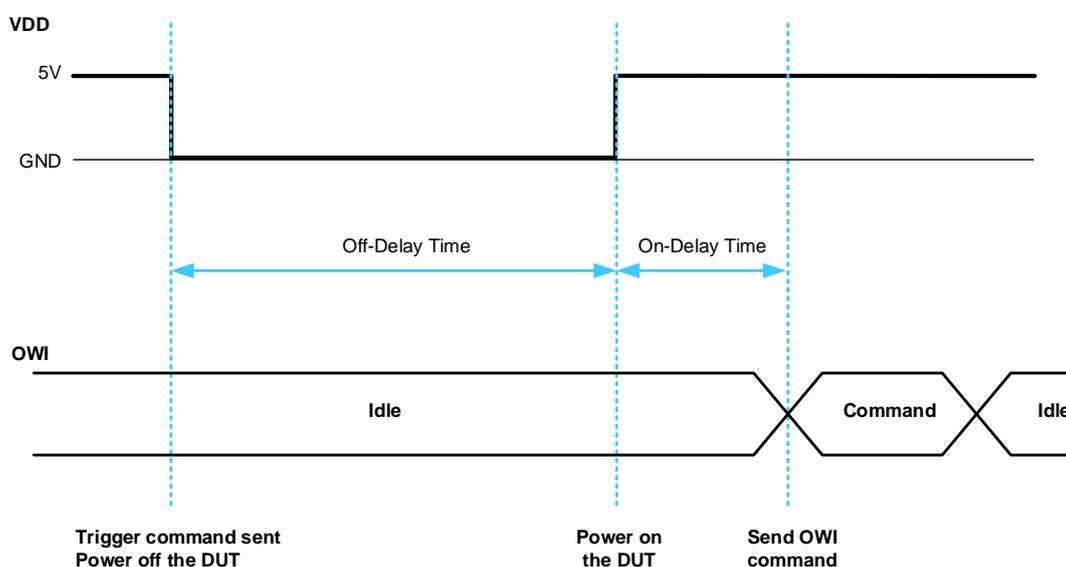
**Table 4. Supported Interfaces Command: V\_FW**

<b>Command</b>	V_FW	
<b>Description</b>	Returns a string with the supported interfaces of the ZMID-COMBOARD.	
<b>Syntax</b>	V_FW	
<b>Example</b>	Send	V_FW
	Response	<ACK>FW Interfaces: ANALOG, OWI, SENT, PWM

**Table 5. Module Select Command: MS**

<b>Command</b>	MS	
<b>Description</b>	Selects the active module (device) between 1 and 2. Further OWI communication or output reading will be performed with the module selected.	
<b>Syntax</b>	MSx x – module/device: x = 0 = Module 1 (Device 1) x = 1 = Module 2 (Device 2)	
<b>Examples</b>	Send	MS0 – Select Device 1 as active
	Response	<ACK>
	Send	MS1 – Select Device 2 as active
	Response	<ACK>

## 2.2 Power and Trigger Commands



**Figure 3. Trigger Command Timing Diagram**

**Table 6. Power and On-Delay Trigger Setup Command: T**

<b>Command</b>	T	
<b>Description</b>	Changes the DUT’s VDD state and sets the on-delay used when executing a trigger command.	
<b>Syntax</b>	Txxttt xx = ZMID VDD state (binary) ▪ xx = 00 <sub>BIN</sub> = Off ▪ xx = 11 <sub>BIN</sub> = On ▪ xx = 01 <sub>BIN</sub> / 10 <sub>BIN</sub> = Forbidden ttt = On-delay time in milliseconds (decimal) from 000 to 999	
<b>Examples</b>	Send	T00000 = Turn off the VDD for the DUT
	Response	<ACK>
	Send	T11001 = Turn on the ZMID VDD and set the on-delay trigger time to 1 millisecond
	Response	<ACK>
	Send	T11020 = Turn on the ZMID VDD and set the on-delay trigger time to 20 milliseconds
Response	<ACK>	

**Table 7. Off-Delay Trigger Setup Command: T\_**

<b>Command</b>	T_	
<b>Description</b>	Defines the off-delay time for the ZMID VDD when executing a trigger command.	
<b>Syntax</b>	T_ttt ttt = Off-delay time in milliseconds (decimal) from 000 to 999	
<b>Example</b>	Send	T_100 = Set the off-delay trigger time to 100 milliseconds
	Response	<ACK>

## 2.3 Communication Commands

**Table 8. OWI WRITE with Trigger Command: OWT**

<b>Command</b>	OWT	
<b>Description</b>	Performs a triggered OWI WRITE command with a command byte and optional data bytes.	
<b>Syntax</b>	OWTccddd cc = command byte (hex string) ddd = data bytes (hex string) - optional	
<b>Example</b>	Send	OWT81FFFF = Trigger command, write 81 <sub>HEX</sub> as command byte and FFFF <sub>HEX</sub> as data bytes
	Response	<ACK>

**Table 9. OWI WRITE Command: OW\_**

<b>Command</b>	OW_	
<b>Description</b>	Performs an OWI WRITE command with a command byte and optional data bytes. If the data bytes are more than 2, a bulk WRITE is performed where the command byte is incremented before writing the next two data bytes. Writing xxx instead of a hex value in the bulk WRITE operation causes the current command byte to be skipped.	
<b>Syntax</b>	OW_ccddd OW_ccddddddd... = Bulk WRITE cc = command byte (hex string) ddd = data bytes (hex string)	
<b>Examples</b>	Send	OW_A0FFFF = WRITE command byte A0 <sub>HEX</sub> and 2 byte data FFFF <sub>HEX</sub>
	Response	<ACK>
	Send	OW_A1BEEFCAFExxxxFFFF = bulk WRITE – start command byte is A1 <sub>HEX</sub> , will skip command byte A3 <sub>HEX</sub>
	Response	<ACK>

**Table 10. OWI READ Command: OR\_**

<b>Command</b>	OR_	
<b>Description</b>	Performs an OWI READ command with a command byte. If a number is specified after the command byte a bulk READ is performed where the command byte is incremented for each READ operation.	
<b>Syntax</b>	OR_cc OR_ccnnn cc = command byte (hex string) nnn = optional: number of bulk READs to perform (decimal) from 000 to 015	
<b>Examples</b>	Send	OR_05 = command byte is 05 <sub>HEX</sub> ; reads one register (2 bytes)
	Response	<ACK>0004 = 2 byte reply from the READ operation
	Send	OR_E2004 = bulk READ; command byte is E2 <sub>HEX</sub> ; reads 4 registers (8 bytes) by incrementing the command byte; Equivalent to sending OR_E2; OR_E3; OR_E4; OR_E5
	Response	<ACK>BEEFCAFE3333FFFF = BEEF <sub>HEX</sub> , CAFE <sub>HEX</sub> , 3333 <sub>HEX</sub> , FFFF <sub>HEX</sub> returned from the bulk READ.

**Table 11. OWI READ Continuous Command: ORS**

<b>Command</b>	ORS	
<b>Description</b>	Starts a continuous reading of a specified command byte. Does 5000 reads if not stopped. The reading cycle includes sending specific commands for stopping and starting the position processing of the ZMID4200. The command returns one normal reply with <ACK> and then continuously sends 2 byte readings. During the reading, the computer (master) must continuously poll its receive buffer for new data from the DUT.	
<b>Syntax</b>	ORScc cc = command byte (hex string)	
<b>Example</b>	Send	ORSD8 = starts a continuous READ of the D8 command byte; reads the spatial angle ( <i>Spa</i> ) register in the SWR memory of the DUT
	Response	<ACK>\r\n = acknowledge reply for the command 13F2\r\n = 2 byte reading 15B3\r\n = 2 byte reading 188C\r\n = 2 byte reading ... 188C\r\n = 2 byte reading runs until 5000 readings are performed or until a STOP command is sent by the computer

**Table 12. OWI READ Special STOP Command: ORSX**

<b>Command</b>	ORSX	
<b>Description</b>	Stops the continuous reading started by the ORS command. Recommendation: The computer (master) should wait approximately 500 milliseconds and then clear its receive buffer before sending another command. There is a small delay between receiving the command in the firmware and stopping the continuous reading	
<b>Syntax</b>	ORSX	
<b>Example</b>	Send	ORSX
	Response	<ACK>

## 2.4 Commands for Reading the Output

The ZMID-COMBOARD supports the reading and interpretation of the analog, PWM, or SENT output depending on the DUT product version. Before reading the output, the ZMID-COMBOARD must be instructed on how to interpret the output from the DUT.

**Table 13. Set Output Interpretation Command: TSO**

<b>Command</b>	TSO	
<b>Description</b>	Sets the output interpretation of the DUT's signal to analog, PWM, or SENT.	
<b>Syntax</b>	TSOxxxx xxxx = 5201 = interpret output as analog xxxx = 5202 = interpret output as PWM xxxx = 5203 = interpret output as SENT Any other combination is forbidden.	
<b>Example</b>	Send	TSO5202 = instructs the firmware to interpret the output of the DUT as a PWM signal
	Response	<ACK>

**Table 14. Read Output Command: MRO**

<b>Command</b>	MRO	
<b>Description</b>	Reads a sample from the interpreted DUT's output; returns a 4-byte reply from which the 12 LSBs are the output data. For analog: 0 <sub>DEC</sub> = 0% VDD; 4095 <sub>DEC</sub> = 100% VDD For PWM: 0 <sub>DEC</sub> = 0% duty cycle; 4095 <sub>DEC</sub> = 100% duty Cycle	

	For SENT: the FC1 (Fast Channel 1) data is directly mapped to the 12 LSBs of the output data	
<b>Syntax</b>	MRO	
<b>Example</b>	Send	MRO
	Response	<ACK>0FFF <sub>HEX</sub> = extracting the 12 LSBs results in an output reading of FFF <sub>HEX</sub> = 4095 <sub>DEC</sub> .

**Table 15. Read Last SENT Frame Command: MRS**

<b>Command</b>	MRS	
<b>Description</b>	<p>Reads the decoded contents of the last received SENT frame. Returns a 4-byte reply with the following encoding:                      SCAAABBB (hex string)                      S – 4-bit status data                      C – 4-bit CRC data                      AAA – 12-bit FC1 data                      BBB – 12-bit FC2 data</p>	
<b>Syntax</b>	MRS	
<b>Example</b>	Send	MRS
	Response	<ACK>06D8DC62 Status: 0 <sub>HEX</sub> CRC: 6 <sub>HEX</sub> FC1: D8D <sub>HEX</sub> FC2: C62 <sub>HEX</sub>

## 2.5 Pin State Commands

**Table 16. Pin State Command: PS\_**

<b>Command</b>	PS_	
<b>Description</b>	<p>Sets the state of a controllable pin of the header on the ZMID-COMBOARD to operate external components such as output signal multiplexors or additional pull-up resistors. The pins can be set to a HIGH, LOW, or high impedance state.</p> <p>Newer versions of the ZMID-COMBOARD (R5_1 and above) have no pin header; instead two signal multiplexors and an additional pull-up resistor are mounted on the board and connected to the following pins:</p> <p>Pin 3 – Stronger pull-up resistors for Device 1 and 2 – used for OWI or PWM                      LOW or high impedance = pull-up inactive                      HIGH = pull-up active</p> <p>Pin 4 – Multiplexor for the output of Device 2                      LOW = used to read analog output                      HIGH = used for OWI, PWM, and SENT                      High Impedance = not defined</p> <p>Pin 5 – Multiplexor for the output of Device 1                      LOW = used to read analog output                      HIGH = used for OWI, PWM, and SENT                      High Impedance = not defined</p> <p><b>Important:</b> Do not change the state of pins 1, 6, or 8.</p>	
<b>Syntax</b>	PS_ppx pp = pin number (decimal) from 01 to 08 x = pin state x = 0 = LOW x = 1 = HIGH x = 2 = Tri-state (high impedance)	
<b>Examples</b>	Send	PS_031 = Enable the pull-up for PWM and OWI communication
	Response	<ACK>
	Send	PS_041 = Set the output multiplexor for Device 2 for digital interfaces (OWI/SENT/PWM)

Response	<ACK>
Send	PS_050 = Set the output multiplexor for Device 1 for analog interface
Response	<ACK>

### 3. Examples

**Table 17. Connecting and Reading EEPROM and SWR Memory (Device 1)**

Command	Comment
MS0	Select Device 1 as active for communication and output reading
<ACK>	
T_100	Power-off delay trigger setup = 100ms
<ACK>	
T11001	Power-on delay trigger setup = 1ms; power on the device
<ACK>	
PS_051	Device 1 output multiplexor set for digital communication
<ACK>	
PS_031	Enable additional pull-up for OWI communication
<ACK>	
OWT0283AE	OWI WRITE with trigger – enter Command Mode
<ACK>	
OR_05	OWI READ – read the status register of the device
<ACK>0004	Status register reply – device is in Command Mode
<b>Memory Read: EEPROM</b>	
<ACK>	
OR_E0015	OWI bulk READ – read 15 registers starting from command byte E0 <sub>HEX</sub>
<ACK>23C8048D00000600120A9D87888E008054BF01085803B107083B0255BFFF	OWI bulk READ reply – 15 registers (30 bytes)
OR_EF003	OWI bulk READ – read 3 registers starting from command byte EF <sub>HEX</sub>
<ACK>0000000000C2	OWI bulk READ reply – 3 registers (6 bytes)
<b>Memory Read: SWR</b>	
OW_04	OWI WRITE – HOLD_DPU command to stop the position calculation while reading data
<ACK>	
OR_C0015	OWI bulk READ – 15 registers starting from command byte C0 <sub>HEX</sub>
<ACK>23C8048D00000600120A9D87888E008054BF01085803B107083B0255BFFF	OWI bulk READ reply – 15 registers (30 bytes)
OR_D1	OWI READ – command byte D1 <sub>HEX</sub>
<ACK>00C2	OWI READ reply – 1 register (2 bytes)
OR_D3009	OWI bulk READ – 9 registers starting from command byte D3 <sub>HEX</sub>
<ACK>03B901E600017FF30321400640E042270001	OWI bulk READ reply – 9 registers (18 bytes)
OW_03	OWI WRITE – RUN_DPU command to start the position calculation
<ACK>	
T00000	Power off the device

Command	Comment
<ACK>	

**Table 18. Writing to the First 7 Registers in EEPROM (Device 1)**

Command	Comment
OW_A023C8	Write to EEPROM register 00 <sub>HEX</sub> ( <i>Offset</i> ); command byte = A0 <sub>HEX</sub>
<ACK>	
OW_A1048D	Write to EEPROM register 01 <sub>HEX</sub> ( <i>Slope</i> ); command byte = A1 <sub>HEX</sub>
<ACK>	
OW_A20000	Write to EEPROM register 02 <sub>HEX</sub> (clamping limits); command byte = A2 <sub>HEX</sub>
<ACK>	
OW_A30600	Write to EEPROM register 03 <sub>HEX</sub> (linear interpolation points 0 and 1); command byte = A3 <sub>HEX</sub>
<ACK>	
OW_A4120A	Write to EEPROM register 04 <sub>HEX</sub> (linear interpolation points 2 and 3); command byte = A4 <sub>HEX</sub>
<ACK>	
OW_A59D87	Write to EEPROM register 05 <sub>HEX</sub> (linear interpolation points 4 and 5); command byte = A5 <sub>HEX</sub>
<ACK>	
OW_A6888E	Write to EEPROM register 06 <sub>HEX</sub> (linear interpolation points 6 and 7); command byte = A6 <sub>HEX</sub>
<ACK>	
OW_A70080	Write to EEPROM register 07 <sub>HEX</sub> (linear interpolation point 8); command byte = A7 <sub>HEX</sub>
<ACK>	

**Table 19. Bulk Writing to the First 7 Registers in EEPROM (Device 1)**

Command	Comment
OW_A023C8048D00000600412A9D87888E0080	Bulk WRITE to EEPROM registers 00 <sub>HEX</sub> to 07 <sub>HEX</sub> (command byte A0 <sub>HEX</sub> to A7 <sub>HEX</sub> )
<ACK>	

**Table 20. Reading 3 Analog Output Samples from Device 1**

Command	Comment
T_100	Power-off delay trigger setup = 100ms
<ACK>	
TSO5201	Set the output recognition to analog
<ACK>	
PS_050	Device 1 output multiplexor set for analog signal
<ACK>	
PS_032	Disable the additional pull-up
<ACK>	
T11001	Power-on delay trigger setup = 1ms; power on the device
<ACK>	
MRO	Read an output sample
<ACK>00000424	Read reply = 424 <sub>HEX</sub> (1060 <sub>DEC</sub> ); 1060 / 4095 * 100 = 25.89% VDD

Command	Comment
MRO	Read an output sample
<ACK>00000424	Read reply
MRO	Read an output sample
<ACK>00000424	Read reply
T00000	Power off the DUT
<ACK>	

**Table 21. Reading SENT Frames from Device 1**

Command	Comment
T_100	Power-off delay trigger setup = 100ms
<ACK>	
TSO5203	Set the output recognition to SENT
<ACK>	
PS_051	Device 1 output multiplexor set for digital signal
<ACK>	
PS_032	Disable the additional pull-up
<ACK>	
T11001	Power-on delay trigger setup = 1ms; power on the device
<ACK>	
MRS	Read last SENT frame
<ACK>05C81B43	Read reply = status: 0 <sub>HEX</sub> ; CRC: 5 <sub>HEX</sub> ; FC1: C81 <sub>HEX</sub> ; FC2: B43 <sub>HEX</sub>
MRS	
<ACK>08C81733	Read reply = status: 0 <sub>HEX</sub> ; CRC: 8 <sub>HEX</sub> ; FC1: C81 <sub>HEX</sub> ; FC2: 733 <sub>HEX</sub>
MRS	
<ACK>0BC812F3	Read reply = status: 0 <sub>HEX</sub> ; CRC: B <sub>HEX</sub> ; FC1: C81 <sub>HEX</sub> ; FC2: 2F3 <sub>HEX</sub>
T00000	Power off the DUT
<ACK>	

**Table 22. Enter Command Mode on Device 2**

Command	Comment
MS1	Select Device 2 as active for communication and output reading
<ACK>	
T_100	Power-off delay trigger setup = 100ms
<ACK>	
T11001	Power-on delay trigger setup = 1ms; power on the device
<ACK>	
PS_041	Device 2 output multiplexor set for digital communication
<ACK>	
PS_031	Enable additional pull-up for OWI communication
<ACK>	
OWT0283AE	OWI WRITE with trigger: Enter Command Mode
<ACK>	
OR_05	OWI READ: Read the status register of the device

Command	Comment
<ACK>0004	Status register reply: Device is in Command Mode

**Table 23. Reading PWM Output from Device 1 and Device 2**

Command	Comment
T_100	Power-off delay trigger setup = 100ms
<ACK>	
TSO5202	Set the output recognition to PWM
<ACK>	
PS_041	Device 2 output multiplexor set for digital signal
<ACK>	
PS_051	Device 1 output multiplexor set for digital signal
<ACK>	
PS_031	Enable the additional pull-up
<ACK>	
T11001	Power-on delay trigger setup = 1ms; power on the device
<ACK>	
MS0	Select Device 1 as active for communication and output reading
<ACK>	
MRO	Read last output sample
<ACK>0000FD0	
MS1	Select Device 2 as active for communication and output reading
<ACK>	
MRO	Read last output sample
<ACK>0000224	
MS0	Select Device 1 as active for communication and output reading
<ACK>	
MRO	Read last output sample
<ACK>00007BC	
MS1	Select Device 2 as active for communication and output reading
<ACK>	
MRO	Read last output sample
<ACK>0000C84	

## 4. Glossary

Abbreviation	Meaning
DUT	Device Under Test
VCOM Port	Virtual Communication Port
ASCII	American Standard Code for Information Interchange – character encoding standard
PWM	Pulse Width Modulation
SENT	Single Edge Nibble Transmission
EEPROM	Electrically Erasable Programmable Read-Only Memory
SWR	Shadow Registers – Working memory of the ZMID4200
DPU	Digital Processing Unit
CRC	Cyclic Redundancy Check
LSB	Least Significant Bit
FC1	Fast Channel 1 of the SENT transmission data
FC2	Fast Channel 2 of the SENT transmission data

## 5. Revision History

Revision	Date	Description
1.0	Jul 21, 2021	Initial release.

## Notice

1. 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
  - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
  - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems 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. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall 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. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Disclaimer Rev.5.0-1)

## Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,  
Koto-ku, Tokyo 135-0061, Japan  
[www.renesas.com](http://www.renesas.com)

## Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

## Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:  
[www.renesas.com/contact/](http://www.renesas.com/contact/)