tinyMicon MatisseCORE™

MtChecker User's Guide

Debugger Configuration Checker for tinyMicon MatisseCORE™ MtChecker User's Guide

No. 67UG027E Rev.002
June 2022

Revision History

Date	Version	Description
2021/10/15	Rev.001	Describe the contents of MtChecker V1.01.00
2022/06/06	Rev.002	Describe the two command line options --show-log and --single-api

Table of contents

1 Technical Terms

Table 1. Technical Terms

Terminology	Description.
tinyMicon MatisseCORE™.	Ultra-compact 8-bit microcontroller developed by ROHM exclusively for LSI embedded systems.
Matisse Debug Interface	A circuit that provides Matisse debugging functions (reset, break, run, register read/write, memory read/write, hardware configuration information acquisition, etc.) and an interface to control it. See the Matisse hardware manual for more information.
Debug adapter	Hardware for operating the Matisse Debug Interface via USB communication, such as matiseye™-adapter and matiseye™-adapter Pro. See the matiseye™-adapter manual for more information.
Debug server	Software for Windows to manipulate the debug adapter via TCP communication. e.g. MtProxy.
Debugger	Software for Windows that communicates with a debug server to debug programs for Matisse. e.g. mtsim and mtcsim.
matiseye™-studio	C language IDE for Matisse, which has a function to launch MtChecker. See the matiseye™-studio User's Guide for more information.

2 Overview

MtChecker is a debugging configuration checker for tinyMicon MatisseCORE™. MtChecker uses the functionality provided by the Matisse Debug Interface to check Matisse configuration and to debug Matisse programs. It is a software for Windows.

3 Features

MtChecker has the following features

1 Displays Matisse hardware configuration information (on-board memory size, initial stack pointer value, presence of multiplier, presence of No-Break memory access).
2 Provides Matisse debugging functions (reset, break, run, breakpoint setting, register read/write, memory read/write, etc.).
3 Other useful functions (program load, program verify, memory dump).

MtChecker was developed as a tool to check the operation of Matisse, so that you can easily check the operation of the Matisse core, the target board, and the program running on Matisse.

4 Operating environment

The operating environment of MtChecker is described below.

4.1 System Requirements

Table 2. System Requirements

OS	Windows 7(32-bit/64-bit) Windows 10(32-bit/64-bit)
CPU	Intel Core series or equivalent performance CPU.
memory	4GByte or more installed.
HDD/SSD	100MByte or more free space.

4.2 Installation

When you run the installer of C compiler for Matisse (MatisseCCompiler-XX.XX.XX.exe), MtChecker is installed at the same time.
In the default configuration, it is installed in "C:\Program Files\ROHM\Matisse\MtChecker".
Moreover, the debug server (MtProxy) that is necessary for MtChecker is installed at the same time as the C compiler for Matisse.

4.3 Configuring the tool

The configuration of the MtChecker is described below.
MtProxy and matiseye™-adapter are required for MtChecker to work.
MtProxy is installed at the same time as MtChecker. matiseye™-adapter is included with the development board/evaluation board.

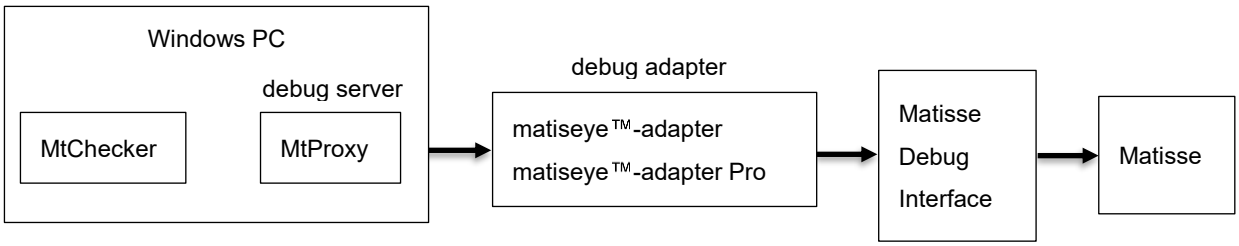

Figure 1. MtChecker System Block Diagram

5 How to use

5.1 Launching MtChecker

You can launch MtChecker by running "MtChecker" from the list of installed apps in Windows.
MtChecker can be also launched from matiseye™-studio.
Run the command "Start MtChecker" from the command palette in matiseye™-studio. The debug server will start automatically and then MtChecker will start.

5.2 GUI Functions

The functions of MtChecker are described below.

5.3 Functions of the Info tab

The information in the Info tab will not be updated automatically. If you want to update the information in the Info tab, restart the application.

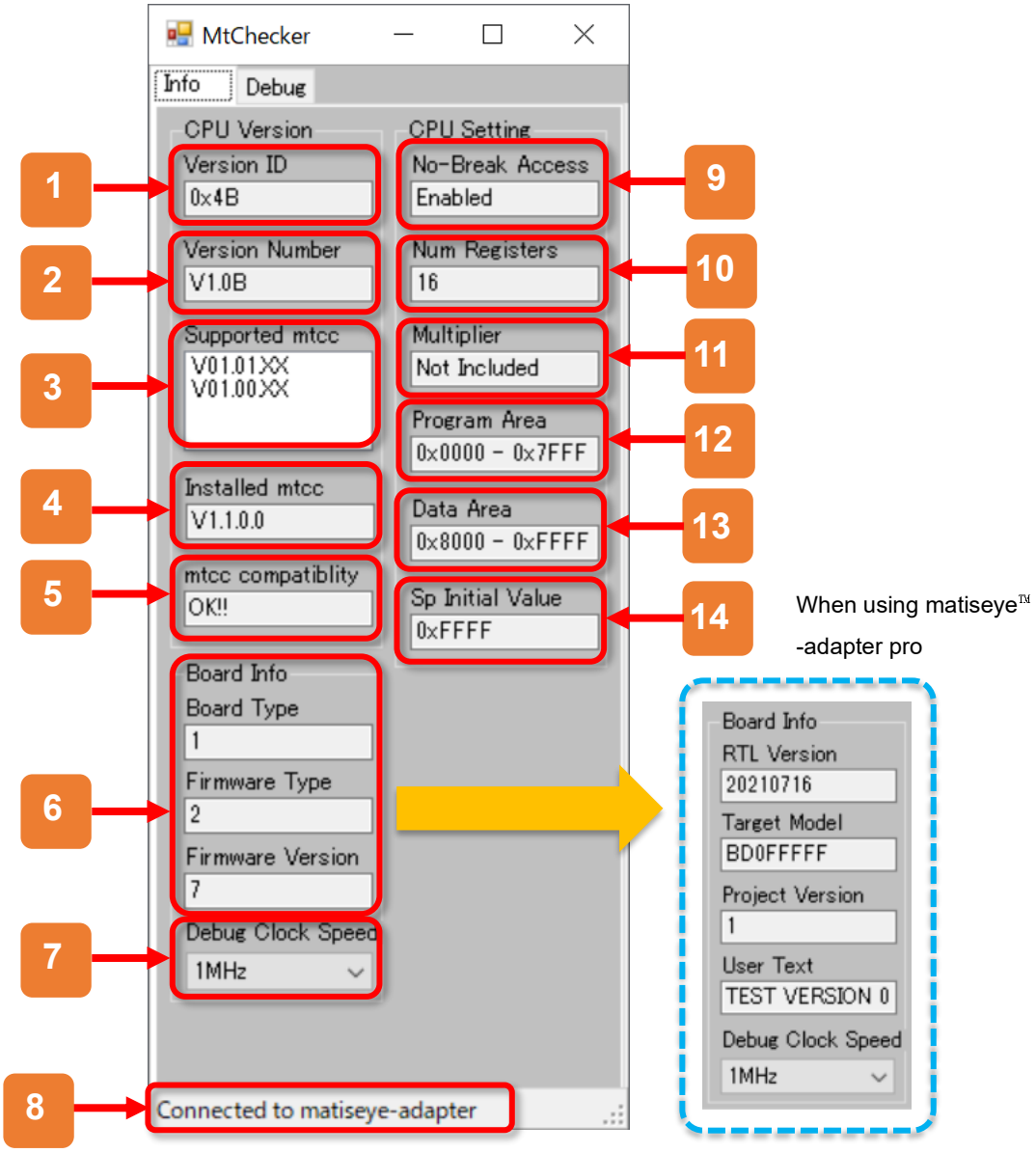

Figure 2. Info Tab Window

Table 3. Names and descriptions of each function in the Info tab

Number	Name	Description
1	Matisse Version ID	The version ID of the Matisse CPU core on the target board is displayed.
2	Matisse Version Number	The version number of the Matisse CPU core installed on the target board is displayed.
3	Supporting Compiler Version	The version series of the compiler that supports Matisse on the target board is displayed.
4	Installed Compiler Version	The version of the compiler installed in the PC is displayed.
5	Compiler Compatibility	Whether the compiler installed in PC supports Matisse on the target board or not is displayed. If it is supported, "OK!!" is displayed. When it is not supported, "Not Compatible!!" is displayed.
6	Board Information	The information of the Debug Adapter board is displayed. For matiseye™-adapter, the board type, firmware type, and firmware version are displayed, and for matiseye™-adapter Pro, the RTL version, target model, and project version are displayed.
7	Debug Clock Speed	Set the clock speed of Matisse Debug Interface. For matiseye™-adapter, from 10 kHz to 1.5 MHz are configurable, and for the matiseye™-adapter Pro and from 10 kHz to 25 MHz are configurable.
8	Debug Adapter Type	The type of debug adapter installed on the target board. matiseye™-adapter or matiseye™-adapter Pro is displayed.
9	No-Break Memory Access	Whether the No-Break memory access feature is enabled or not is displayed. If the feature of the Matisse on the target board is enabled, "Enabled" is displayed, otherwise "Disabled" is displayed.
10	Number of Registers	The number of registers of Matisse on the target board is displayed.
11	Multiplier Presence	Whether the multiplier is installed in Matisse on the target board or not is displayed.
12	Program Area Range	The range of the program area of Matisse on the target board is displayed.
13	Data Area Range	The range of the data area of Matisse on the target board is displayed.
14	SP Initial Value	The initial value of SP (Stack Pointer) of Matisse on the target board is displayed.

5.4 Functions of the Debug tab

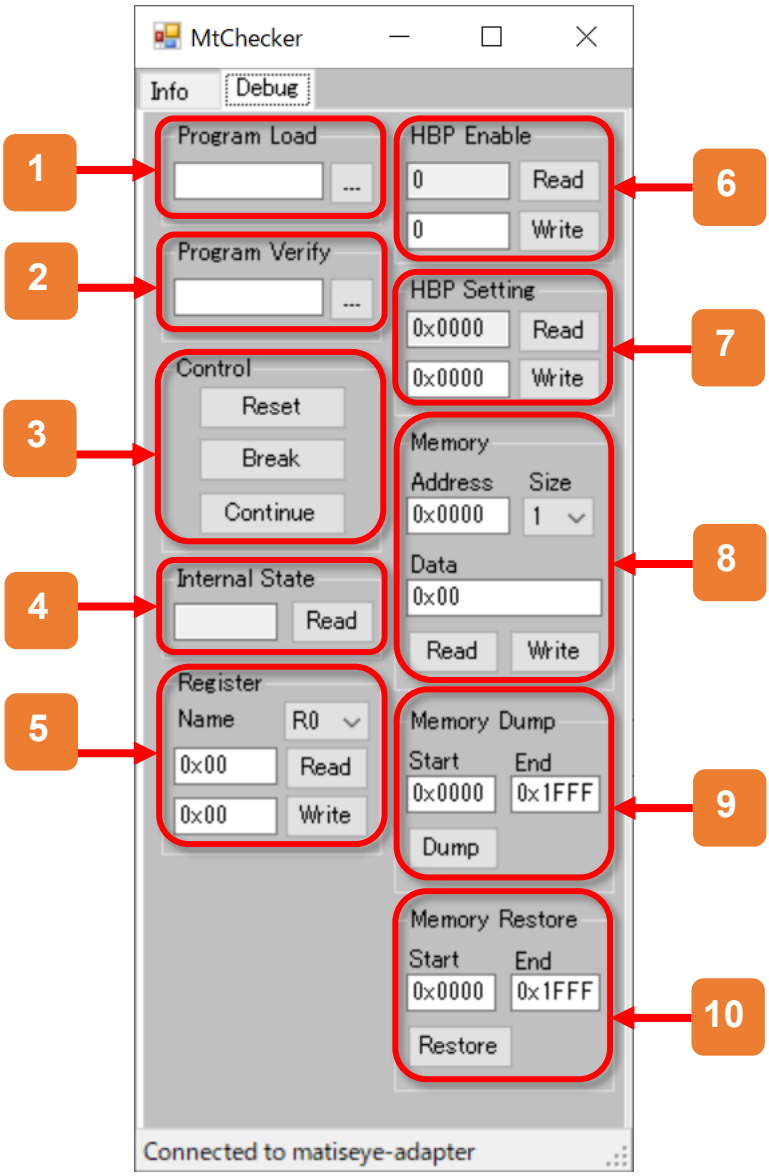

Figure 3. Debug Tab Window

Table 4. Names and descriptions of each function in the Debug tab

Number	Name	Description
1	Program Load	The specified binary file is loaded into the program area of Matisse on the target board. When the loading is completed, "Load finished" is displayed on the screen.
2	Program Verify	The content of the specified binary file is compared with the program area of Matisse on the target board from 0x0000. If all of them match, the message "Verify OK" is displayed. If the comparison fails, the error message "Verify Failed at address 0xXXXXX" and the address where the comparison failed are displayed.
3	CPU Control	If the Reset button is clicked, the CPU is reset. If the Break button is clicked, the CPU go into the break state and stops program execution. If the Continue button is clicked, the break state is released and the program execution is restarted.
4	CPU Internal Status Reading	If the Read button is clicked, the internal state of Matisse on the target board is acquired and displayed. There are 7 types of internal state as follows: IDLE, PRE_FETCH, FETCH, WAIT_EXBUS, WAIT_KEEP, HALT, BREAK.
5	Register Read/Write	It reads and writes the value of the register of Matisse on the target board. You can select the register of the target from the pull-down list, and the value is read by the Read button, and the value is written by the Write button.
6	Hardware Breakpoint Enable/Disable	Enable/Disable the hardware breakpoint of Matisse on the target board. If the Read button is clicked and 0 is displayed, the hardware breakpoint is disabled, and when 1 is displayed, the hardware breakpoint is enabled. Entering 0 and clicking the Write button will disable the hardware breakpoint, and entering 1 and clicking the Write button will enable the hardware breakpoint.
7	Hardware Breakpoint Setting	Set the address of the hardware breakpoint of Matisse on the target board. Read the current hardware breakpoint address with Read button, and change the hardware breakpoint address with Write button.
8	Memory Read/Write	Reads and writes to the memory at the specified address and size. Click the Read button to read the memory, and click the Write button to write the memory. Byte order is little endian.
9	Memory Dump	The value of the specified range of memory is dumped to the file as binary data. When dumping is completed, the message "Dump Finished" is displayed on the screen.
10	Memory Restore	Restore the data of the file saved in the memory dump function to the specified area of memory. When the restoration is complete, "Restore Finished" is displayed on the screen.

5.5 CUI Functions

The following is a description of the functions when MtChecker is used as a CUI tool.

The following command line options are available when using MtChecker from the command prompt.

5.5.1 --show-log

- The format of the command line option

```
MtChecker.exe --show-log
```

- Example of use

```
> MtChecker.exe --show-log
```

- Description

Running MtChecker from the command prompt with the "--show-log" option will output the MtChecker communication log to the command prompt.

With this option, the MtChecker GUI is displayed on the screen, and a log of commands sent and responses to those commands are output in real time as you interact with the GUI.

The command logs can be used with the "--single-api" to run MtChecker functions from the command line.

The description of the log is as follows.

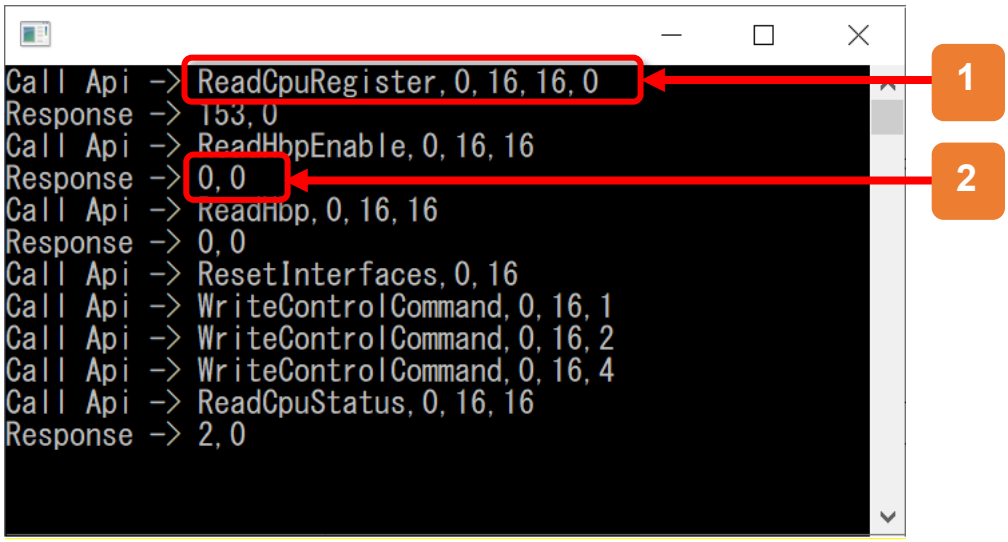

Figure 4. –show-log Optional execution screen

Table 5. –show-log Optional log names and descriptions

Number	Name	Description
1	Log of command sent	"Call Api ->" string followed by the contents of the command sent. The format of the contents of command sent is "API name, parameters, parameters,,,,". This content string can be used as a parameter for the parameter of --single-api command line option.
2	Log of response received	"Response -> " string followed by the response to the command sent. Some commands have no response.

- Precautions

This feature cannot be used at the same time as the "--single-api" option.

5.5.2 --single-api

- The format of the command line option

```
MtChecker.exe --single-api [API Name],[API Parameter],[API Parameter],,,,
```

The number of the API parameters is variable.

- Example of use

```
> MtChecker.exe --single-api ReadCpuRegister,0,16,16,0
```

- Description

When MtChecker is run from the command prompt with the "--show-log" option and command parameters, you can run MtChecker as a CUI tool without GUI window.

The available commands can be found in the log output displayed by the "--show-log".

This feature allows you to automate the execution of MtChecker from a batch file or script.

The steps for MtChecker automation are as follows.

1. Start MtChecker with the "--show-log" option.
2. Perform the function you want to automate.
3. Copy the API name and the parameters from log output.
4. Write a batch file or script that runs MtChecker with the "--single-api" option and copied parameters.
5. Run the batch file or script you created.

- Example of scripts

The following is an example script that reads the value of the R0 register and writes 1 to the R1 register.

```
"C:\Program Files\ROHM\Matisse\MtChecker\MtChecker.exe" --single-api ReadCpuRegister,0,16,16,0
"C:\Program Files\ROHM\Matisse\MtChecker\MtChecker.exe" --single-api WriteCpuRegister,0,16,1,1
```

- Precaution

This feature cannot be used at the same time as the "--show-log" option.

6 Open-source software licenses

This software includes open-source software (hereinafter referred to as "open-source software program") provided under the following license conditions, in addition to software for which ROHM owns or is licensed.
Open-source software programs are subject to their respective license terms, so in the event of a conflict between the license terms of an open-source software program and this material, the license terms of the open-source software program shall prevail.

Included open-source software and their license terms

・Mono.Options(The MIT License)

7 Trademark notices

"Windows" and "VS Code" are trademarks of Microsoft Group companies.
"Intel" is a trademark of Intel Corporation or its subsidiaries.
"Core™" is a trademark or registered trademark of Intel Corporation or its subsidiaries.
“tinyMicon MatisseCORE™” and “matiseye™” are a trademark or registered trademark of ROHM Corporation.

Caution

1. The information written in these materials regarding the software and system (hereinafter collectively “Software”) and the contents of the materials are current as of the date of the material’s issuance, and may be changed by ROHM, at any time and for any reason, without prior notice.

2. If you plan to use the Software in connection with any equipment or device (such as the medical equipment, transportation equipment, traffic equipment, aerospace equipment, nuclear power control equipment, vehicle equipment including the fuel control system and/or car accessories, and/or various kinds of safety devices etc.) which require extremely high reliability, and whose breakdown or malfunction relate to the risk of personal injury or death, or any other serious damage (such usage is hereinafter called “Special Usage”), you must first consult with the ROHM’s sales representative. ROHM is not responsible for any loss, injury, or damage etc. incurred by you or any other third party caused by any Special Usage without ROHM’s prior written approval.

3. Semiconductor products may break or malfunction due to various factors. You are responsible for designing, testing, and implementing safety measures in connection with your use of any ROHM products using the Software (such ROHM products are hereinafter called “Product”) Such safety measures include, but are not limited to, derating, reductant design, fire spread prevention, backup, and/or fail safe etc. in order to prevent the accident resulting in injury or death and/or fire damage etc.. ROHM is not responsible and hereby disclaims liability for any damage in relation to your use beyond the rated value, or the non-compliance with any precaution for use.

4. ROHM is not responsible for any direct and/or indirect damage to you, or any third parties, (including the damage caused by loss of intangible asset such as information, data, or program etc., loss and/or interruption of profit) which is caused by the use or impossibility to use of the Software.

5. Since the Software, these materials, and/or the Product contain confidential information of ROHM’, including technical information, and/or trade secrets, you are prohibited from engaging in any of the following acts in whole or part, without ROHM’s prior written approval:
   (i) disclosing any ROHM confidential information to a third party;
   (ii) disassembling, reverse engineering, and/or any other analysis;
   (iii) reprinting, copy, and/or reproduction; or
   (iv) removing the copyright notice included in the Software.

6. When exporting the Software, or the technology and/or confidential information written in these materials, you are required to follow the applicable export control laws and regulations such as “Foreign Exchange and Foreign Tarade Act” and/or “Export Administration Regulations (EAR).”.

7. ROHM disclaims all warranties, statutory or otherwise, and ROHM hereby disclaims any warranty for non-infringement for the Software and/or the information written in these materials. Accordingly, ROHM is not liable to you for any direct or third-party claims of infringement of rights.

8. No license, whether expressly or implied, is granted hereby under any intellectual property rights or other rights of ROHM or any third parties with respect to the Software or Products or the information contained in these materials.

9. You agree to indemnify, defend and hold harmless ROHM and ROHM’s officers and/or employees from responsibility, and hold them harmless, and defend them from any damage, loss, penalty, or cost caused by any claim of liability (including but not limited to the attorney fees) resulting from, or incurred relating to the following acts:
   (1) any alleged infringement of a third party’s rights or the violation of laws caused by reading, download, encryption, summarization, copy, or transfer etc.; or
   (2) violation of these materials.

10. ROHM does not guarantee that these materials or the Software is error free. ROHM shall not be in any way responsible or liable for any damages, expenses, or losses incurred by you or third parties resulting from errors contained in these materials.

Thank you for using ROHM products.
For inquiries about our products, please contact us.

ROHM Customer Support System

https://www.rohm.co.jp/contactus