**Freescale Semiconductor, Inc.**

Freescale Semiconductor, Inc.

# MCUez
# MMEVS0508 TARGET
# USER'S MANUAL

**Important Notice to Users**

While every effort has been made to ensure the accuracy of all information in this document, Motorola assumes no liability to any party for any loss or damage caused by errors or omissions or by statements resulting form negligence, accident, or any other cause. Motorola further assumes no liability arising out of the application or use of any information, product, or system described herein; nor any liability for incidental or consequential damages arising from the use of this document. Motorola disclaims alll warranties regarding the information contained herein, whether expressed, implied, or statutory, including implied warranties of merchantability or fitness for a particular purpose. Motorola makes no representation that the interconnection of products in the manner described herein will not infringe on existing or future patent rights, nor do the descriptions contained herein imply the granting or license to make, use, or sell equipment constructed in accordance with this description.

**Information contained in this document applies to REVision (1) MCUez.**

The computer program contains material copyrighted by Motorola Inc., first published 1997, and may be used only under a license such as the License For Computer Programs (Article 14) contained in Motorola's Terms and Conditions of Sale, Rev. 1/79.

**Trademarks**

This document includes these trademarks:

MCUez is a trademark of Motorola Inc.
EXORciser is a trademark of Motorola Inc.

The MCUez development, emulation, and debugging application is based on Hi-Wave, a software technology developed by HIWARE. Hi-Wave is a registed trademark of HIWARE AG.

AIX, IBM, and PowerPC are trademarks of International Business Machines Corporation.
SPARC is a trademark of SPARC International, Inc.
Sun and SunOS are trademarks of Sun Microsystems, Inc.
UNIX is a trademark of Novell, Inc., in the United States and other countries, licensed exclusively through X/Open Company, Ltd.
X Window System is a trademark of Massachusetts Institute of Technology.

# CONTENTS

## CHAPTER 1   IINTRODUCTION

## CHAPTER 2   MMEVS0508 TARGET COMPONENT

**CHAPTER 3    MMEVS COMMANDS**

**FIGURES**

# CHAPTER 1

# IINTRODUCTION

## 1.1    INTRODUCTION

MCUez features target system interfaces that give you the ability to load different target components of an embedded system. This document introduces the interfaces related to the MMEVS0508 Modular Evaluation System for the MC68HC05 and MC68HC08 MCU Families.

The MMEVS0508 is an emulator system designed by Motorola for MCUs that use a CPU05 or 08 to provide emulation memory. MMEVS interface allows an external target system based on a Motorola MCU to download an executable program from the MCUez environment, execute it, and relay the results of the real target system behavior to MCUez.

MCUez fully supervises and monitors the target system's MCU. It also controls CPU functions such as read and write in internal/external memory (even when the CPU is running) and single-step/run/stop.

**NOTE**

When an external MCU is executing code, the memory statistics are not available to the MMEVS target component. As a result, Profiling and Coverage Analysi will not perform properly with the MMEVS target component.

## 1.2    DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

| | |
|---|---|
| **DLL** | Dynamic Link Library: Windows library file used in dynamic linking. |
| **Dynamic Linking** | Windows process used to link a function call in one module to the actual function in the library module at run time. |
| **MMEVS** | Motorola Modular Evaluation System. |
| **MMEVS0508 Server** | The MMEVS hardware access library responsible for interfacing across the RS-232 port to the MMEVS0508 station. |
| **Modal Dialog** | Dialog that requires a response before continuing. |
| **MCU** | Micro Controller Unit |
| **EM** | Emulation Module |

## CHAPTER 2

## MMEVS0508 TARGET COMPONENT

### 2.1 INTRODUCTION

This chapter describes how set up communication between MCUez and the MMEVS0508 and explains the windows and dialog boxes associated with the process.

### 2.2 TARGET/SYSTEM INTERFACE

The MMEVS0508 box is connected to the RS-232 serial port of your system.

#### 2.2.1 Hardware Connection

Use the serial link cables provided with the MMEVS0508 to connect the host computer to the MMEVS0508 box (see the following diagram). The host computer is configured as a data terminal, i.e. it sends data on the TxD lead and receives data on the RxD lead (refer to the MMEVS0508 hardware manual).
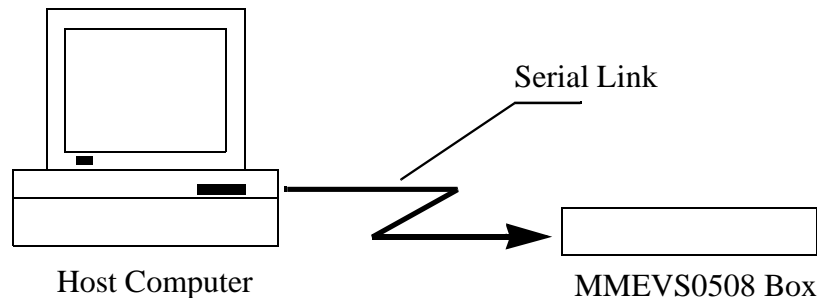


Serial Link

Host Computer

MMEVS0508 Box

**Figure 2-1.  Hardware Connection**

#### 2.2.2 Loading The MMEVS0508 Target

The target component is usually determined by the PROJECT.INI file. To make the MMEVS the target, change the "Target=" line in the PROJECT.INI file to "Target=Motosil". The MotoSIL driver automatically attempts to locate the MMEVS. If the driver can not locate the target, an error message appears and informs you that the target is not connected to the expected port. The Communication Configuration dialog also appears. Using the Communication Configuration dialog, you can change the baud rate and communication port parameters. See the Communication Configuration section in this chapter for more details.

**MOTOROLA**

You can also load the MMEVS0508 target by selecting *Set Target...* from the *Component* Menu (shown below) and choosing MotoSIL from the list of proposed targets.



**Figure 2-2. Component Menu**
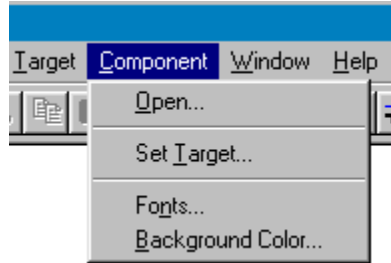
After the target is successfully loaded, the *Target* Menu is replaced by the *MMEVS0508* Menu.
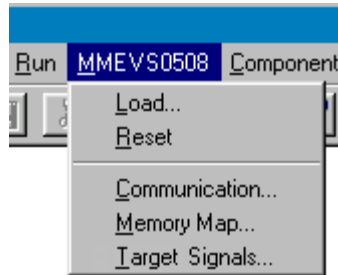


**Figure 2-3. Replacing The Target Menu**

If the MotoSIL driver does not detect a target, the *MotoSIL* Menu replaces the *Target* Menu:
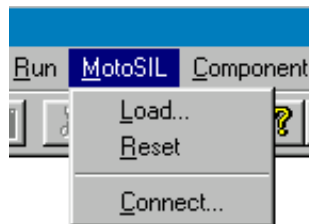


**Figure 2-4. MotoSIL Menu**

The *MMEVS0508* Menu will replace the *MotoSIL* Menu if the MMEVS target is successfully located during later attempts.

## 2.3    CONFIGURATION DIALOG

Ensure the parameters on the host computer are correctly configured and that the serial communication device is set correctly. If the configurations and settings are not correct, communication between MCUez and the target is not possible.

### 2.3.1    Communication Configuration

In a general way, the communication between MCUez and the MMEVS is set automatically. If MCUez does not detect a target, the Communication Device Specification dialog will open. This dialog can also be opened selecting the *MotoSIL* menu and choosing *Connect*....

### 2.3.2    Communication Device Specification Dialog

If the host and target are not connected or the communication device is not properly indicated, a dialog box, Communications Device Specification, appears:
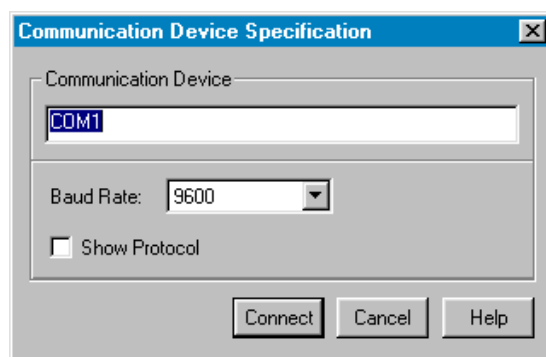


**Figure 2-5.  Communication Device Dialog**

The default device is COM1. To choose another available communication device, enter the device name in the *Communication Device* edit box, scroll for the corresponding *Baud Rate*, select it, and click *Connect*.

After the connection with the chosen communication device is established, the settings for that device are saved for later debug sessions. If the connection fails, a message box will be displayed and a new communication device can be defined. To close the dialog box without attempting a new connection, select *Cancel*.

**NOTE**

The communication device and baud rate, which are saved through this dialog, override the environment variables `BAUDRATE` and `COMDEV` in the `project.ini` file.

### 2.3.3 Data Format

MCUez uses the data format: 8 data bits, 1 stop bit, no parity, and a variable baud rate. The default baud rate is 9600 baud unless it has been redefined through the Communications Device Specification dialog.

## 2.4 MCUEZ STATUS BAR FOR THE MMEVS

When the MMEVS Target Component has been successfully loaded, the MCUez status bar displays specific information:
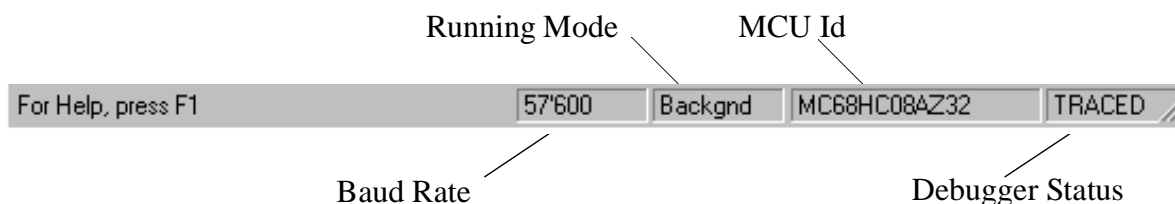


**Figure 2-6. MCUez Status Bar**

Displayed from left to right are: the baud rate of serial communication, the MCUez running mode, the name of the connected MCU (depending on the MCU-Id), and the debugger status.

## 2.5 MMEVS0508 MENU

When the MMEVS0508 is successfully set as the target, the *MMEVS0508* menu appears (see Loading the MMEVS0508 Target in this chapter). The *MMEVS0508* menu contains functions specific to the MMEVS0508.

### 2.5.1 MMEVS0508 Menu Entries

The functions in the *MMEVS0508* menu are specific to the MMEVS0508 and are only available with this emulator component. To use an item in the *MMEVS0508* menu, select the menu and choose the desired function:
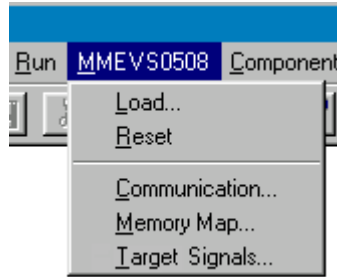
**Figure 2-7.  MMEVS0508 Menu**

The *Communication...*, *Memory Map...*, and *Target Signals...* functions are part of the MMEVS0508 Motorola Modular Evaluation System and are supported by libraries that consist of dialog boxes and routines that interface with the debugger and hardware.

### 2.5.1.1 Communication

The system operates most efficiently when the baud rate of the host computer is at its maximum. To display the Communication Device Specification dialog, select *Communication...* from the *MMEVS0508* menu. Select the maximum baud rate your host supports or 115200, the maximum baud rate currently supported. If communication fails, the previous baud rate is used.
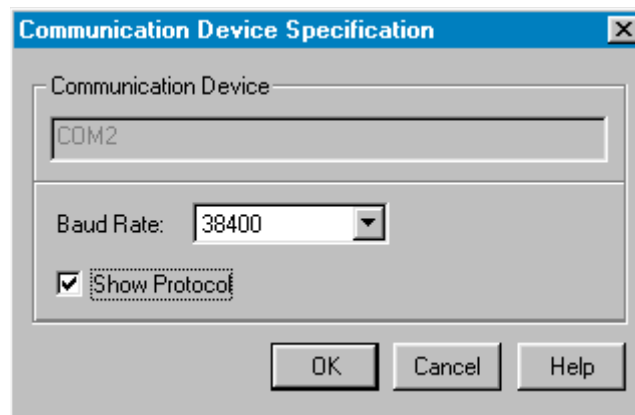


**Figure 2-8.  Communication Device Specification Dialog**

### 2.5.1.1.1   Baud Rate

The maximum baud rate depends on the speed and the interrupt load of the host computer. The default value is 9600.

#### 2.5.1.1.2 Show Protocol

Check the *Show Protocol* box to display the communication protocol in the Command line Component. Display from the communication protocol results in superfluous output in the in the Command Line component. This feature should only be used for advanced debugging issues.

### 2.5.1.2 Memory Map

Choose *Memory Map...* from the *MMEVS0508* menu to open the Memory Configuration dialog:



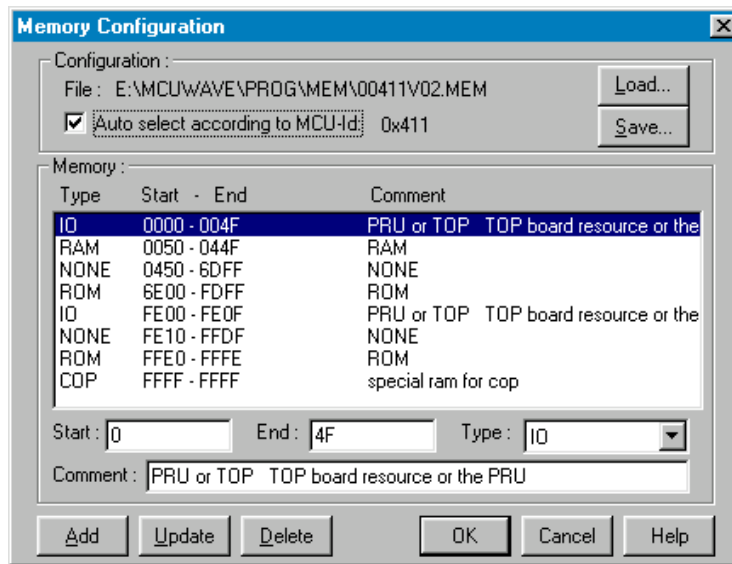**Figure 2-9. Memory Configuration Dialog**

*Configuration* contains the target's memory setup. The target memory setup is automatically loaded if the *Auto select according to MCU-Id box* is checked. MCUez identifies and sets the memory map through the MCU-Id. To open another configuration, select the *Load...* button. To save modifications to the current configuration, select the *Save...* button.

### 2.5.1.3 Target Signals

Choose *Target Signals...* from the *MMEVS0508* menu to open the Target Signals dialog shown below:
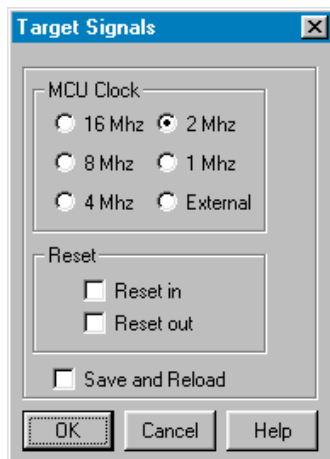


**Figure 2-10.  Target Signals Dialog**

The MMEVS0508 emulator signals can be specified through this dialog. When the Target Signals dialog is accessed, the settings are read from the MMEVS0508. If you alter the displayed values and select *Ok*, the values are written to the MMEVS0508. If you check *Save and Reload*, the configuration is reloaded when the MCUez debugger is started.

Target Signals dialog can also be used to alter the MCU clock and reset signal connection.

#### 2.5.1.3.1   MCU Clock

*MCU Clock* allows you to change the MCU clock if the EM is correctly configured. Please refer to the hardware manual that came with your emulator system.

#### 2.5.1.3.2   Reset

*Reset* allows you to control the reset signal connection.

## 2.6    ABOUT PERSONALITY (.MEM) FILES

To work properly, the Motosil target has to load the personality file (.MEM file) matching with the connected Emulation Module (EM).

 The .MEM file filename is built as follows:

0nnnnVxx.MEM

where 'nnnn' is the four hex digit 'MCU-Id' number of the MCU connected, and 'xx' is a two digit version number.

If this file is not found or is not valid, the 'Configuration Error' dialog box shown below is displayed:
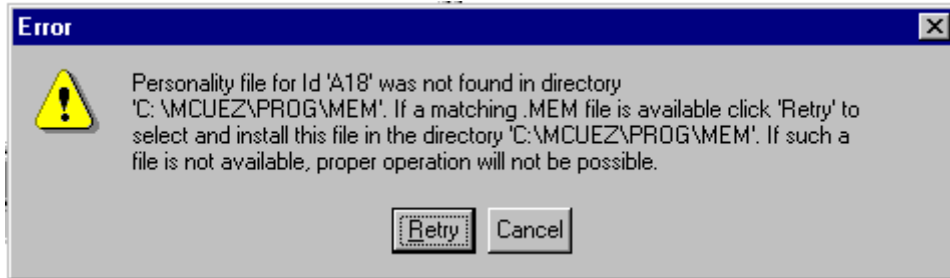


**Figure 2-11.  Configuration Error Dialog**

When choosing 'Cancel', the connection is not established and the 'Communication Device Specification' dialog is opened.

When choosing 'Retry', the 'Open Personality File' shown below is displayed
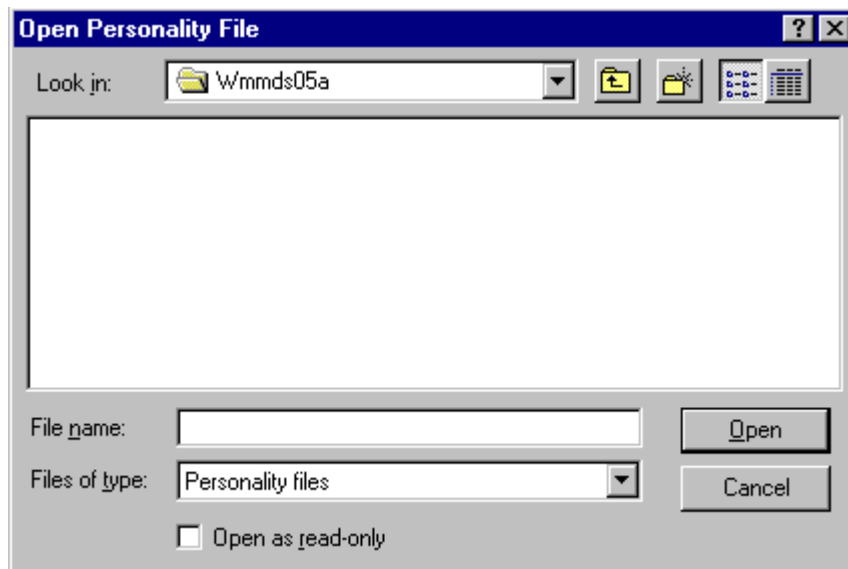


**Figure 2-12.  Open Personality File Dialog**

You can browse to open the required .MEM file.

If the .MEM file that you selected is not valid, an error message box pops up, and then the 'Configuration Error' dialog box is opened again.

If the selected .MEM file is valid, it is loaded and copied into the \PROG\MEM directory with 'V00' as version number (e.g. 00A18V00.MEM).

**NOTE**

The 'Memory Configuration' dialog displays the current memory map.

When starting MCUez, if the 'Auto Select according MCU-ID' check box is checked, the default personality file matching with the MCU-ID is automatically loaded. If this check box is not checked, the previous opened or saved memory map file is automatically loaded.

The memory configuration can be modified within this 'Memory Configuration' dialog and saved into a memory configuration file when clicking the 'SAVE' button .

You can load a different memory configuration file (e.g. that you defined yourself and saved previously) when clicking the 'LOAD' button of the 'Memory Configuration' dialog.

Also Personality files or any memory configuration files can be loaded with the LOADMAP command line command.

## 2.7  DEFAULT TARGET SETUP

The MMEVS target component can be loaded from the *Target* menu or set as a default target in the `project.ini` file, which should be in the working directory .

Example `project.ini` file:

```
[DEFAULTS]
Window0=Source     0    0   50   40
Window1=Assembly  50    0   50   40
Window2=Register  50   40   50   30
Window3=Memory    50   70   50   30
Window4=Data       0   40   50   25
Window5=Command    0   65   50   20
Window6=Module     0   85   50   15
Target=Motosil
[Motorola ESL]
COMDEV=COM2
BAUDRATE=57600
SHOWPROT=0
```

### 2.7.1  Motorola ESL Parameters

Normally, the *Motorola ESL* parameters are interactively set during installation. The settings are stored in the `project.ini` file and used in the future debugging sessions.

### 2.7.1.1 COMDEV

The *COMDEV* parameter specifies the communication device to be used between the host and MMEVS0508. COM1 is the default communication device for PCs. The default for UNIX systems is /dev/ttya.

For a PC: Any valid communication device (COM1,COM2,etc.).

Example:COMDEV=COM2

For SUN:Any valid communication device (/dev/ttya, etc.).

Example:comdev=/dev/ttyb

### 2.7.1.2 BAUDRATE

*BAUDRATE*, the rate of serial communication between the target and host systems, is preset to 9600 bauds and may be overwritten by one of the following baud rates:

1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600, or 115200.

Example:BAUDRAUTE=19200

### 2.7.1.3 SHOWPROT

*SHOWPROT* determines the display of the communication protocol. The value is set to 0 or 1 (1 to show the communication protocol, 0 to hide it) at the first communication attempt with the target. You can check the *Show Protocol* checkbox in the Communication Device Specification dialog if you want to display the communication protocol in a Command Line window.

## 2.8   MMEVS TARGET STARTUP FILE

The startup command file startup.cmd is executed by MCUez immediately after the MMEVS target driver has been loaded. You can generate this file, but it must be located in the working directory. Any MCUez command can be used in this file. A wide set of commands are introduced in the MCUez manual.

Example of a startup.cmd file content:

wb 0x0035   0x00

wb 0x0012   0x11

## 2.9   MMEVS RESET COMMAND FILE

To execute the MMEVS reset command file (reset.cmd) select Reset in the Target/MMEVS menu. When selecting the menu entry MMEVS0508|Reset, the debugger executes the commands specified in the file reset.cmd located in the project directory. Any MCUez command can be used in this file.

Freescale Semiconductor, Inc.

# CHAPTER 3

# MMEVS COMMANDS

## 3.1 INTRODUCTION

This section describes MMEVS-specific commands. MMEVS-specific commands are typed in the Command Line component or inserted into a command file.

For further details about commands, refer to the *MCUez Debugger User's Manual*.

## 3.2 BAUD RATE COMMAND

### Description:

**BAUD**

| Description: | baud rate. |
|---|---|
| Syntax: | **BAUD** [<rate>] |
| | **rate**: Specifies the new baud rate, and must be an integer constant with one of the following specific values (decimal):<br><br>1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600, 115200. |

The BAUD command sets or displays the baud rate for communication between the MMEVS and the host computer. For maximum performance, the baud rate should be set as high as the host computer can accommodate. The maximum rate is 115200; the default baud rate is 9600.

If no baud rate is entered, the command displays the Communications Device Specification dialog window for interactive selection of the baud rate. The baud rate dialog window can also be selected from the menu.

### Example:

```
BAUD 57600
```
Changes the communication baud rate to 57600.

## 3.3    TARGET SIGNAL EMULATION COMMANDS

### SIG

| Description: | set emulator signals. |
| --- | --- |
| Syntax: | **SIG** [ [ENABLE] <signal> {<signal>}] [ DISABLE <signal> {<signal>}] |
|  | **signal**: The signal to be enabled or disabled. Available signals are **RESETIN** and **RESETOUT**. |
|  | **ENABLE**: Enable means to connect the signal from the target system. |
|  | **DISABLE**: Disable means to disconnect the signal from the target system. |

### Description:

The SIG command enables or disables control signals from the target MCU, and controls emulation. If a signal or list of signals is entered with neither ENABLE nor DISABLE, the signal(s) are enabled. If no signal is specified, the Target Signal dialog appears.

### Example:

```
SIG ENABLE RESETIN
```
Enables the 'reset in' signal from the target system.

Freescale Semiconductor, Inc.

## 3.4 RESET COMMANDS

### RESET

| | |
|---|---|
| Description: | reset. |
| Syntax: | reset [GO \| STOP] |
| | **GO**: Resets the MCU and does a Go from Reset. |
| | **STOP**: Resets the MCU and stops (default). |

### Description:

The **RESET** command resets the target MCU. RESET GO performs a reset of the EM and automatically start execution of the code starting at the address stored in the reset vector. RESET STOP performs a reset and wait for user commands.

### Examples:

```
RESET GO
```

Resets the MCU and does a Go from Reset.

```
RESET
```

Resets the MCU and stops (default).