Rize Microcomputer Model RMS102

Quick Start Guide

IMPORTANT INFORMATION

- RMS102 is sensitive to ESD (Electro-Static Discharge).
- Observe ESD precautions.
- For software license details refer to license.dat file
- To download more documentation, source code, etc, visit: www.revely.com

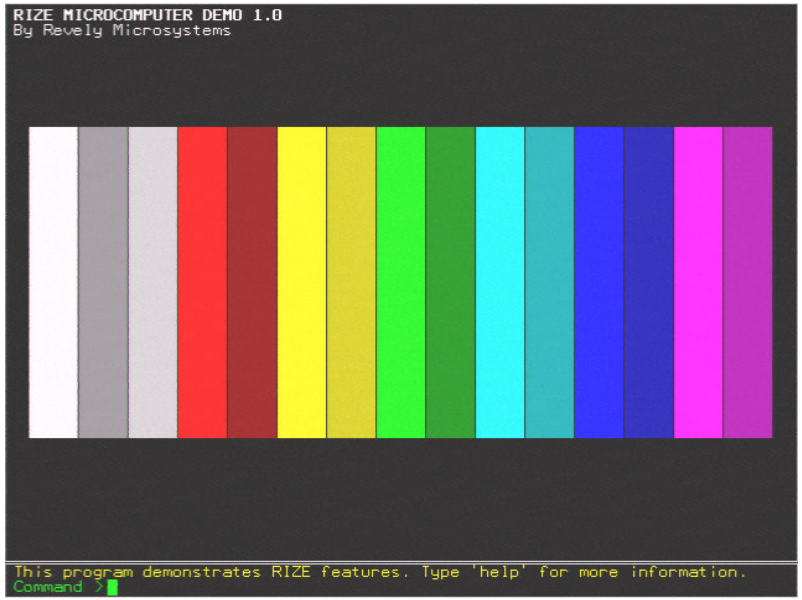
STEP ONE – VERIFY OPERATION

Before trying out your own code or development tools, fire-up RMS102 and try out the factory installed demo program.

You'll need the following:

- Rize RMS102 Microcomputer (-NTSC or –PAL it doesn't matter)
- 5-8Vdc power source
- PS/2 keyboard
- PS/2 Mouse (optional) and Y cable
- VGA monitor
- Television and composite video cable (also optional)

Set-up is simple. Just connect the TV or monitor, plug in the keyboard and apply power. The default video format is VGA, so if you have a TV connected press <F10> to toggle to a TV test screen. Type 'help' for a list of text commands.



Problems with this step? Email tech@revely.com

STEP TWO – REVIEW FOUR WAYS TO DOWNLOAD CODE / DEBUG

The RMS100 supports several debugger/download options:

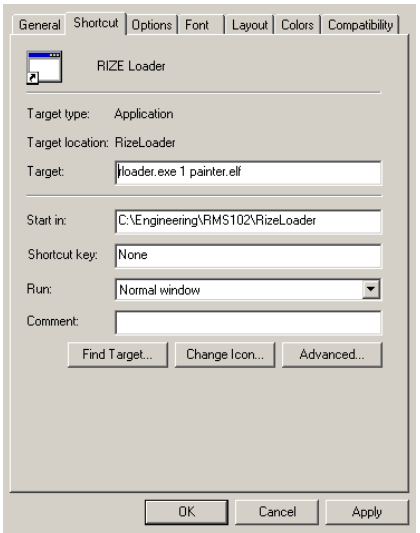
1. **Rizeloader and a SC102 serial cable**. Freescale was kind enough to build a tiny serial bootstrap program into the i.MX. Even if all memory is erased or corrupted, this little program accepts simple memory access commands using RS232. Rizeloader is a Windows console application that uses the i.MX bootstrap program to load a faster client program, which handles the actual transfer in Flash or SDRAM. Transfer rates are typically 5KB/sec.
2. **Embedded USB CrossConnect**. RIZE's circuitry includes a high performance JTAG debug interface. CrossConnect works with CrossStudio, a powerful IDE for ARM from Rowley Associates. Use CrossStudio to develop C/C++ program and perform source level debugging in Flash or SDRAM. Download/upload speeds are greater than 110KB/sec.
3. **Cardloader** is a Flash resident application that reads an ARM ELF executable program from SD Flash media into SDRAM and executes it. Transfer is approximately 1MB/sec. Programs cannot be loaded in Flash using this tool. Cardloader will be available soon from www.revely.com. It can be installed using Rizeloader.
4. **3rd Party Debuggers.** Other ARM development tools can be used with RIZE. The Debug Connector has all ARM JTAG signals present. Revely part JTAG102 adapts these connections to the standard ARM 20way 0.1" header configuration. This header has been tested with Macraigor Wiggler and Rowley CrossConnect.

This guide explains how to set up and use Rizeloader and Embedded CrossConnect.

OPTION ONE – USING RIZELOADER

Using Rizeloader requires the following items:

- Rize RMS102 Microcomputer
- 5-8Vdc power source
- Serial (SC102 or similar)
- Bootstrap Jumper
- Windows PC with a serial port
- rizeloader.zip (download from www.revely.com)



To set up rizeloader:

1. Unzip rizeloader.zip
2. Create a desktop shortcut to rizeloader.exe with the following parameters:
   rizeloader.exe 1 yourfile.elf
   The first argument specifies which PC serial port to use (1=COM1, etc). The second argument is the name of the ELF file to be loaded.

Note: You can use the command prompt to enter this command if you prefer.

To load an ELF file into memory:

1. Connect the PC serial port to RMS102's I/O Expansion connector using the SC102 serial cable.
2. Insert the bootstrap jumper in the System connector.
3. Apply power
4. Click on the Rizeloader shortcut to start the transfer.

```
*****************************************
* RIZE Loader V1.0 for RIZE Model RMS102 *
*   Copyright 2005 Revely Microsystems   *
*****************************************

Client file is rize_loader.elf (6KB)
Image file is painter.elf (48KB)
Connected to RMS102 target!
!! Target Link Confirmed !!
Programming Client..
.load          :    3104 bytes to be stored in SDRAM at 0x08FF0000
Connected to Client
Programming Image..
.vectors       :      60 bytes to be stored in Flash at 0x10000000
.init          :    1048 bytes to be stored in Flash at 0x1000003C
.text          :   15756 bytes to be stored in Flash at 0x10000454
.rodata        :    1483 bytes to be stored in Flash at 0x100041E0
Programming Completed !
painter.elf is now running
Press any key to exit_
```

Screen-shot of successful ELF transfer using Rizeloader

Problems with this operation?

- Check that you have specified the correct COM port and it is not in use elsewhere.
- Rize_loader.elf must be in the same directory as rizeloader.exe
- Check the bootstrap jumper and re-apply power to reset the i.MX
- Try loading example programs from www.revely.com

OPTION TWO – USING EMBEDDED CROSSCONNECT

To start high-performance source-level debugging, you'll need the following items:

- RIZE Microcomputer Model RMS102
- USB cable (mini B)
- Windows or Linux PC
- CrossStudio from Rowley Associates (www.rowley.co.uk)
- CrossConnect device driver
- 5-8Vdc Power source

Set up and installation:

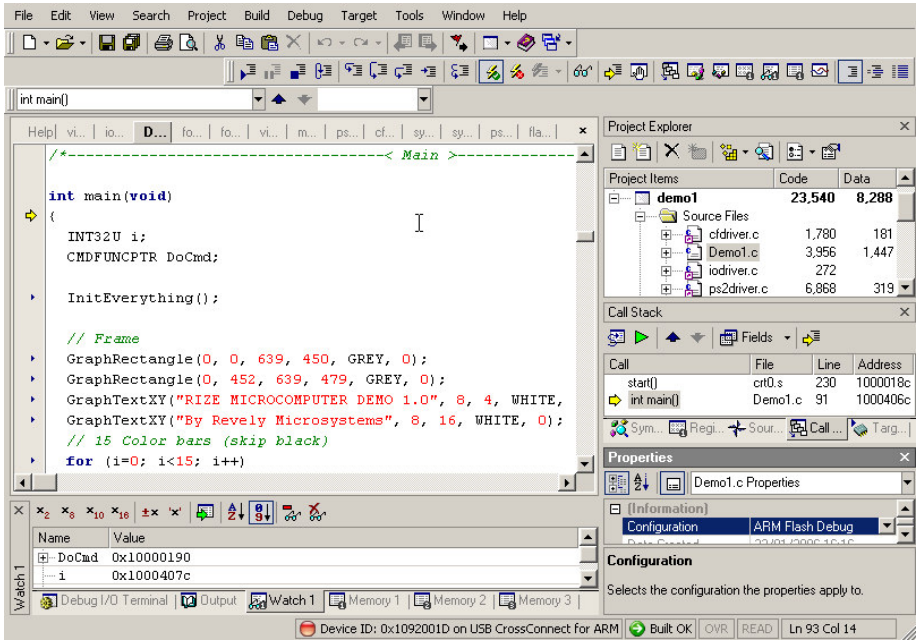
- Download and install CrossStudio (evaluation licenses are available from Rowley).
- Download and install the CrossConnect USB driver from http://www.rowley.co.uk/arm/CrossConnect.htm
- Connect RMS102 to you PC using the USB cable and follow the CrossConnect driver installation instructions.

Future releases of CrossWorks will include integrated support for RMS102. For V1.5 and earlier releases you will need to copy several files into the CrossWorks install directories.

1. Download rms102_crossworks.zip from www.revely.com
2. Unzip the file into the CrossWorks install directory. Typically C:\Program Files\Rowley Associates Limited\CrossWorks for ARM 1.5

Loading and debugging a program using CrossWorks:

1. Open a RIZE CrossWorks project file
   a. Download from www.revely.com (application source code only).
   b. From the RIZE Source Code CD (complete source included).
3. Select **Target→Select USB CrossConnect for ARM**
4. Select **Target→Start Debugging**
5. CrossStudio will download the program to Rize and halt at the start of 'main'. Select **Debug→Go** to run the program.



For CrossStudio documentation and support contact Rowley Associates.