

# GRMON

## Debug Monitor for LEON



### Introduction

GRMON is a debug monitor for the LEON Debug Support Unit (DSU), providing a non-intrusive debug environment on real target hardware. The LEON DSU can be controlled through any AMBA AHB master and GRMON therefore supports communication through a large number of interfaces.

### Features

- Read/write access to all LEON registers and memory
- Downloading and execution of LEON applications
- Built-in disassembler and trace buffer management
- Breakpoint and watchpoint management
- Command-line mode or graphical user interface
- Remote connection to GNU debugger (GDB)
- Single word or burst access to AMBA on-chip bus
- GRLIB Plug & Play support
- Auto-probing and initialisation of LEON peripherals and memory settings
- Error injection for fault-tolerant LEON processors
- Supported debug interfaces: PCI, USB, Ethernet, JTAG, UART and SpaceWire
- Common Flash Interface (CFI) compatible Flash PROM programming
- Loadable modules for custom commands and DSU interfaces

### Graphical user interface

The GrmonRCP graphical user interface (GUI) for GRMON is based on the Eclipse Rich Client Platform (RCP). It provides a comprehensive and customizable interface to all GRMON functions.

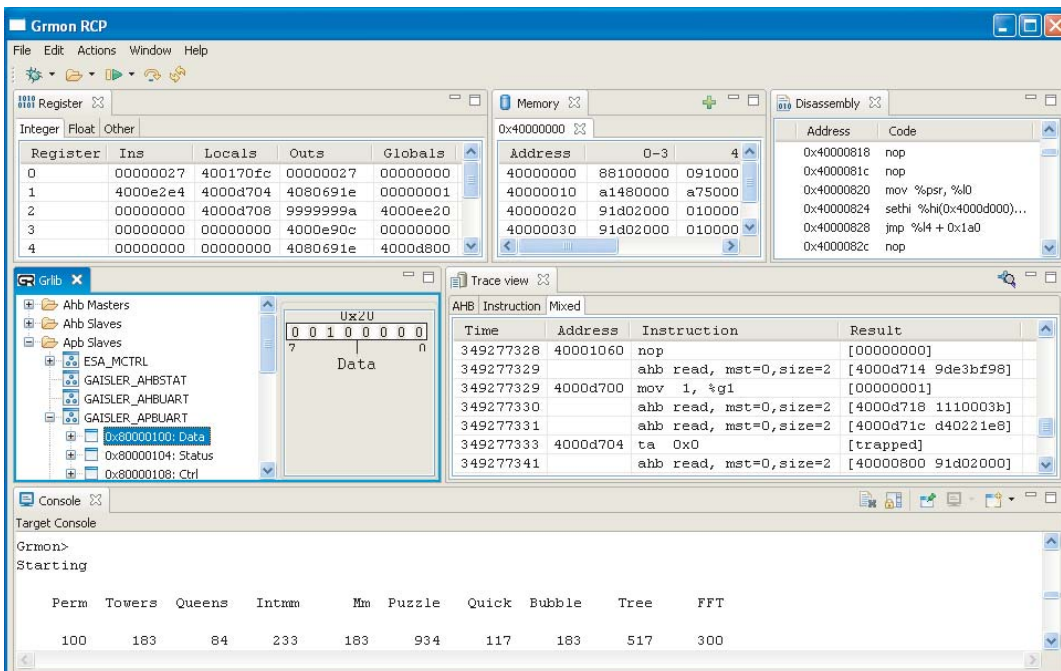


Figure: Screenshot of a GrmonRCP debug session

### Standalone operation

GRMON can operate in standalone mode or attached to the GNU debugger (GDB). In standalone mode, LEON applications can be loaded and debugged using a command line interface or via the graphical user interface. Numerous commands are available to examine data, insert breakpoints and advance execution.

### GDB operation

In GDB (GNU Debugger) mode, GRMON acts as a gateway and translates the GDB extended-remote protocol to debug commands on the target system. Applications are loaded and debugged through GDB (or a front-end such as DDD). Any tool supporting the GDB extended-remote protocol can be used to interface GRMON.

## Plug & Play

When GRMON first connects to the target system, it scans the system to detect which IP cores are present. This is done by reading plug & play information available on the AMBA AHB bus. A debug driver for each recognized IP core is loaded, which performs a core-specific initialization sequence as required. For a memory controller, the initialization sequence would typically consist of a memory probe operation to detect the amount of attached memory. For a UART, it could consist of initializing the baud rate generator and flushing the FIFOs.

## Trace buffer

Depending on the LEON configuration, the DSU trace buffer can store the latest executed instructions, the latest AMBA AHB bus transfers, or both. GRMON supports all three modes and provides individual listings or a common view of the traces.

## MMU support

If the LEON processor includes a Memory Management Unit (MMU), GRMON can automatically perform virtual to physical address translation when responding to GDB requests. This allows source-level debugging of programs and operating systems using the MMU, such as Linux and VxWorks.

## Multiprocessor support

In a LEON3 and LEON4 multiprocessor system each processor's state is maintained separately, including the trace buffer. The program execution on each processor can therefore be traced individually. GRMON provides commands to view the state and trace buffer of each processor one at a time. To test different system configurations, processors can be enabled or disabled independently. A disabled processor will remain halted when the program is started. Each processor can be set to halt when one of the other processors is halted, e.g. due to a breakpoint or a trap, to have as low impact as possible on the system during debugging.

## Error injection

All memory blocks in the fault tolerant LEON3-FT processor (i.e. cache memory and register file) are Single Event Upset (SEU) protected. The GRMON error injection function emulates SEU occurrences in memory blocks and allows the user to test the fault-tolerance by inserting random bit errors during program execution. An injected error flips a randomly chosen memory bit in one of the memory blocks, effectively emulating a SEU. The user defines the error rate and chooses between uniform and average error distribution. GRMON can also perform error correction monitoring and report error injection statistics including the number of detected and injected errors and the error coverage.

## Extending GRMON

GRMON can be extended by means of loadable modules. A custom DSU interface not already integrated in GRMON can be added through the use of a custom DSU communication module. This module only needs to provide functions to read and write to the AMBA AHB bus. Custom IP cores can be supported with new commands that are added through a custom command module, which is provided with read and write access to the AMBA AHB bus and the register file.

## Availability

GRMON is immediately available for LEON2, LEON3 and LEON4. GRMON supports the following platforms: Linux, Solaris, Windows (NT/2000/XP) and Windows with Cygwin.

### CONTACT INFORMATION

Aeroflex Gaisler AB  
Kungsgatan 12  
411 19 Göteborg  
Sweden

Tel: +46 31 7758650  
Fax: + 46 31 421407

Sales contact:  
sales@gaisler.com  
www.aeroflex.com/gaisler