

MemTest

Data Sheet

25 January 2011

PRODUCT DESCRIPTION

MemTest is a memory testing tool for systems designed with Freescale's CPM-based and QUICC Engine (QE)-based PowerQUICC devices. The tool is fully implemented in microcode and runs on the CPM or the QE. It can be used with Static, Dynamic, SDRAM, or DDRx, and works by flooding the memory under test with well distributed random data, and then verifies the integrity of this data.

The MemTest microcode module is much more than just a memory testing tool. It can also be used with the increasingly popular DDRx memory to both adjust and fine-tune its sensitive timing. Because of the module very small footprint it can be downloaded into the internal DPRAM by the COP/BDM port, and still leave enough free space on the DPRAM to include extensive error reporting if required. MemTest is "a must have tool" for any board designer/tester as it can be used to test the memory even before a single instruction is run by the host. It can be set up to test the memory immediately after power up and before booting.

The user can specify the number of different tests to be carried out (1 to 4,294,967,296). The module also provides an extensive error reporting.

MemTest enables the user to ensure that the memory controller and the timing are property configured under the specific board layout, making MemTest an essential tool for testing RAM functionality during board qualification.

The module's key features are listed below:

- Simulates very heavy/stressed bus activity during memory testing
- Has a very small footprint - can run from DPRAM!
- Helps to configure the DDR(s) memory timing even before Flash
- 32-bit preset value, enable 4,294,967,296 different pseudo random pattern sequences
- Quits after a specified number of detected errors
- Provides comprehensive reporting: Bad-address, Read-data, Expected-data, XOR (helps to detect possible cross-talk or bad lane)
- Extremely fast (on a MPC830E MDS board, QE=400MHz and CCB/DDR= 266MHz, the module is able to test DDR1 32MByte of memory in less than 680 MSec!)
- Can optionally perform random reads from a configurable address range while memory fill, to mix different bus operations





- Three modes of operation:
 - Fill Only
 - Check only
 - Fill & Check
- Easy to operate

Important Note: This new version of MemTest replaces our very popular and no-longer supported earlier version that uses tests such as Walk-One, Walk-Zero, By-Pattern, etc. The new random data test method proved to be much superior both in terms of error detection and speed of operation.

For customers who are using non-CPM/QE devices, (e.g. MPC5200, MPC74xx), we offer a host assembler version.

ABOUT DOGAV SYSTEMS

DoGav Systems is a leading provider of software and hardware consultancy and training services. It specializes in Freescale's processors, in particular the PowerQUICC family of communication processors. It has a proven track record of over 20 years supporting Freescale customers in developing market-leading products for the communications equipment market.

DoGav Systems is Freescale's most experienced and active microcode developer. Since receiving its license in 2000, it has developed numerous customized microcode packages for both small and large Freescale customers. These packages are now successfully deployed in commercial products. In addition, DoGav Systems also offers more than 30 off-the-shelf microcode products for the PowerQUICC I, PowerQUICC II, PowerQUICC III and PowerQUICC II Pro processors.



The following figures illustrate the operation of MemTest. Figure 1 shows a snapshot of memory filled with random data by MemTest. Figure 2 shows the same memory snapshot after 4 cells were manually modified, and the detailed error report generated by MemTest.

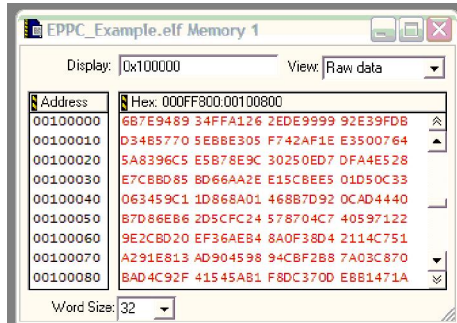


Fig.1: A snapshot of 256 bytes of memory filled by MemTest with random data.

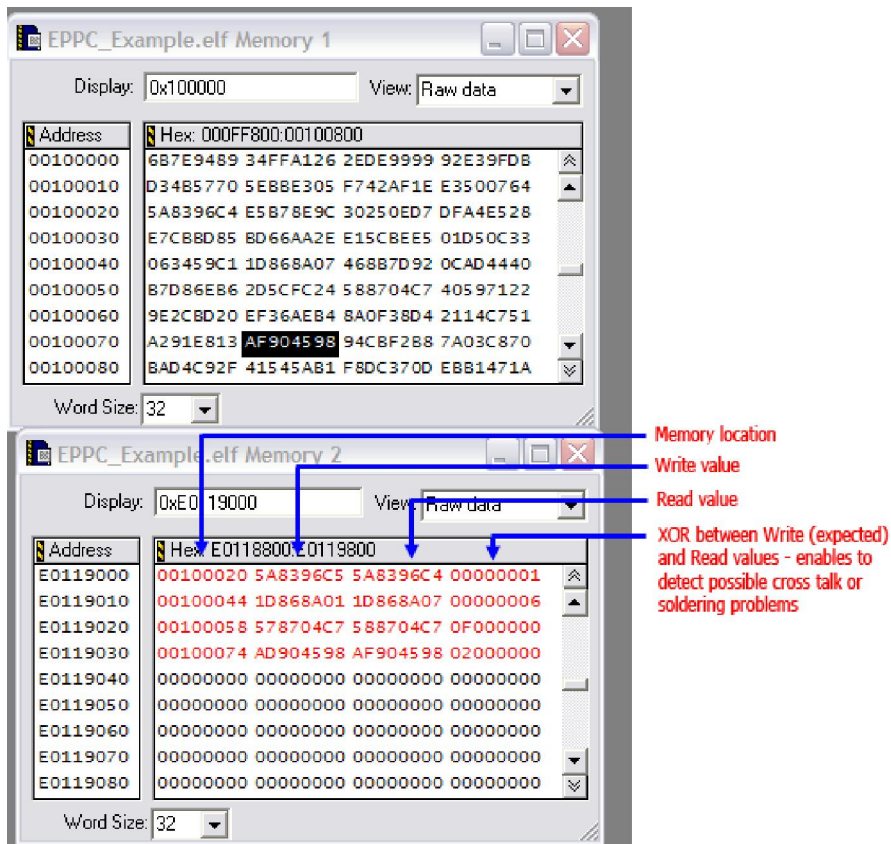


Fig. 2: Top half shows the same memory locations after 4 cells were manually modified. Bottom half shows the error report that was generated by MemTest as a result of the memory verification.