

Frascati, March 1, 1994

Note: **C-11**

DEVIL Tester

G. Baldini, G. Di Pirro

The DANTE control system for DAΦNE is based on a multiprocessor scheme that uses about 70 VME CPUs named "DEVIL". The DEVIL has been designed and built by a commercial firm on a prototype developed in the LNF.

A software program to check the functionality of a DEVIL in the shortest possible time has been implemented.

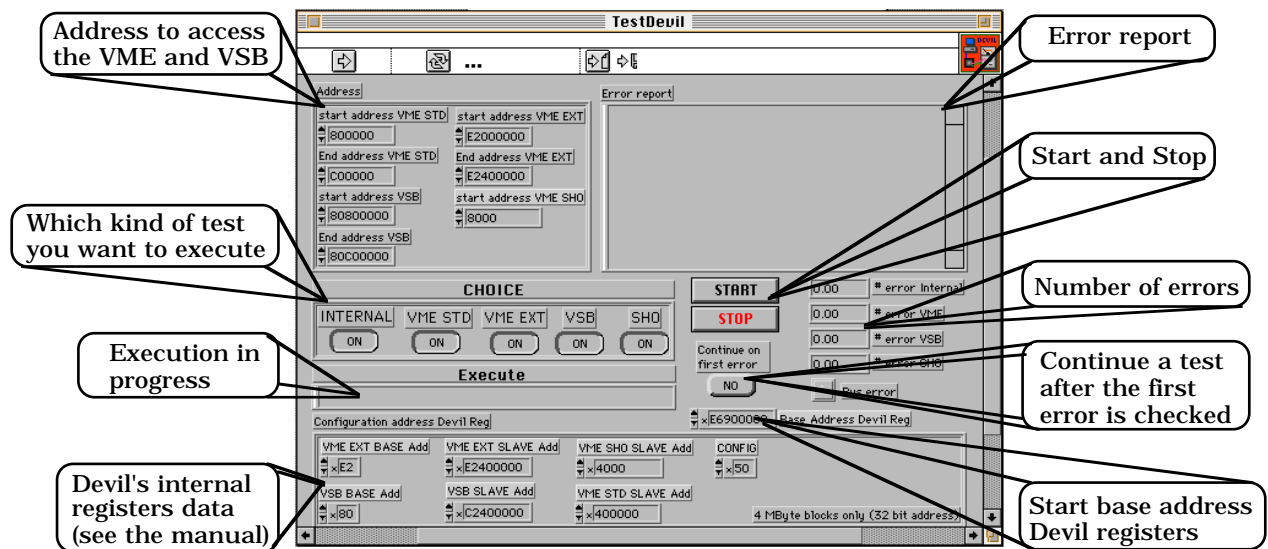
When we want to check the functionality of a DEVIL installed in one of the crates of the control system, an additional DEVIL, called TESTER, is installed in the crate. The TESTER carries a program, "Devil Tester", which must be copied also on the DEVIL under test through the AppleTalk network! Devil Tester runs on both CPUs, the TESTER and the DEVIL under test.

The Devil Tester program

When the program is launched we get a dialog window which reminds the correct hardware linking for arbitration and the position of the DEVIL in the crate.

If everything is correct we push OK and then we go to the control panel to choose the possible test options.

We can test the DEVIL's memory through VME Extended (EXT) or VME Standard (STD) or VSB bus, or their combinations. (It's also possible to test the internal memory).



It is possible to choose between stopping execution or continuing after an error is detected.

After these operations we can click on the start button to start the memory test.

The Devil Tester program starts with the "INIT devil" virtual instrument which initializes the VIC 68 chip and the VME-VSB interface addresses.

This operation must be performed on both DEVILs, the one under test and the Tester.

The actual test consists of 4 phases. In every phase one data path is checked: the paths are:

- DEVIL to internal memory;
- DEVIL to DEVIL via VME (standard A24 D32 VME);
- DEVIL to DEVIL via VME (extended A32 D32 VME);
- DEVIL to DEVIL via VSB.

In every phase 4 Mbytes of memory are completely written, read back, and tested for errors. This operation is repeated three times with the following data:

- writing in every location its own address;
- always writing the number 0x5AA55AA5;
- always writing the number 0xA55AA55A.

This method allows to test every location for address coherence and to test every bit with 0 and 1.

For each data type the test is performed three times, using Byte, Word and Longword addressing.

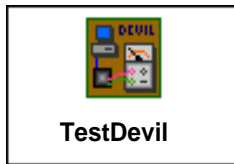
The Devil Tester :Time response

We have performed 6 tests and the time required durations are printed in the table below.

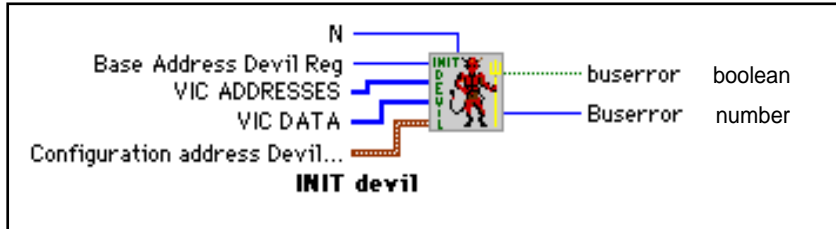
Through BUS	Data Type	Durations
Internal memory	Byte, Word, Long	150 sec
VME STD	Byte, Word, Long	173 sec
VME EXT	Byte, Word, Long	167 sec
VSB	Byte, Word, Long	163 sec
SHO	Byte	0.02 sec
All test	Byte, Word, Long	654 sec

This result is obtained using virtual instruments programmed in C language and inserted in the LabVIEW program as CINs (Code Interface Node).

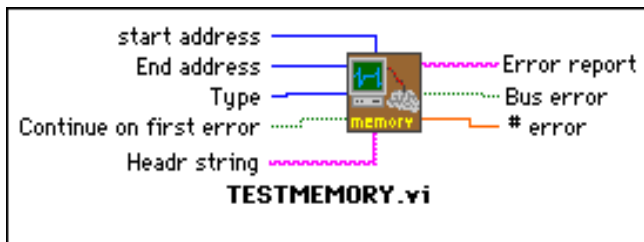
We include a list of the program SubVIs (Labview Virtual Instruments Subroutines) with a short description of their operation.



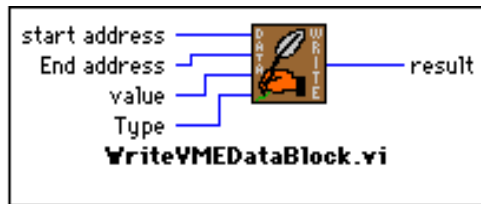
Main program icon



Initialize VIC 68 and DEVIL internal registers



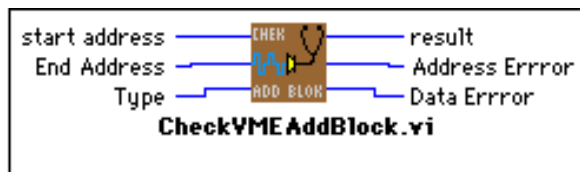
Test memory from start address to End address using Type (0 = Byte, 1 = Word, 2 = Long) . Header string is the header for the error report. # error gives the number of errors found if Continue on first error is true.



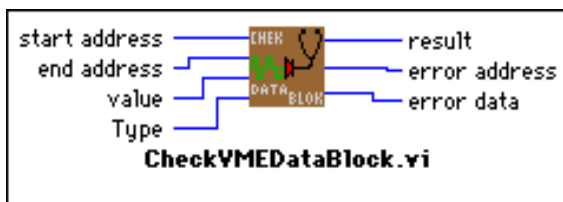
Write value from start address to End address.



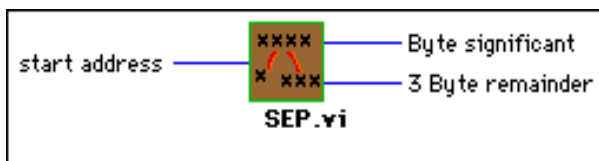
Write from start address to End address a value equal to the location address.



Read from start address to End address checking that the location data correspond to addresses.



Read from start address to End address checking that the location data correspond to value.



Separate a longword into two parts, the Most Significant Byte and the remaining three bytes. Used to write the DEVIL register for A32 VME access.