On-line help on Muon Fanout Card test software

B.Satyanarayana,
Department of High Energy Physics,
Tata Institute of Fundamental Research,
Mumbai, INDIA, 400005

February 27, 2001

Abstract

Since this is an on-line help document, it assumes that the Muon Fanout Card (MFC) test software (MFCTEST) is up and executing in the test stand. Please refer to the other related documents for the specification (Specification of the D0 MFC test stand software), installation (Get started with MFCTEST software) and complete technical reference (MFCTEST: Muon Fanout Card Test Software) of MFCTEST. The last document is under preparation.

1 General comments

MFCTEST requires files 'mfctest.hlp' - its help text file, 'inttrig.dat' - trigger data file for running the MFC internal sequencer and 'intrtrig.dat' - trigger data file for running the MFC internal sequencer in order to generate some of the interrupts are all located in its working directory.

MFCTEST is implemented in terms of simple menus, starting with a main menu (Muon Fanout Card Test Software Menu) and branching into the individual functions or utilities menus. The last option of any menu brings the control back to the next higher level menu.

MFCTEST works under certain constraints with regards to operator interface, operating system and/or its programming language. The input is limited through keyboard (no mouse) and the output to a simple alphanumeric text terminal (no graphics and GUIs). There is no default defined for any input that MFCTEST requests; entering a RETURN without typing any input may mess up the screen. Appropriate configurable in-line editing keys may be used while entering the inputs. User inputs are case insensitive. Suitable screen formats are implemented for all the output. When more than a screenful of data or messages are to be output, they are formatted into pages, prompting the user before displaying the next page. CTRL+C is the program break key.

Functions that work on the MFC and/or MFTC modules, probe for their presence in the test crate before performing the requested operation on them. The function quits with an error message if it cann't find the module(s) in the crate. Similarly most of the bus and other hardware errors are detected by MFCTEST and appropriate action is taken on these events based on the user inputs and other test conditions.

2 Muon Fanout Card Test Software Menu

The main menu screen is as shown below.

```
+---------------------------------------------+
|                                             |
|     MUON FANOUT CARD TEST SOFTWARE MENU     |
|                                             |
|     A. Initialization functions menu        |
|     B. Memory & register checks menu        |
|     C. MFC NVRAM configuration handler      |
|     D. MFC readout modes tests menu         |
|     E. MFC interrupts tests menu            |
|     F. MFC data browsers menu               |
|     G. Diagnostic utilities menu            |
|     H. Miscellaneous functions menu         |
|     I. Display MFCTEST log files            |
```

```
|      J. Short help on MFCTEST software       |
|      K. About MFCTEST                        |
|      L. Exit MFCTEST                         |
|                                              |
+----------------------------------------------+
```

Options 'A' to 'I' will be explained in the following sections. Option 'J' displays this help document with appropriate pagination. MFCTEST's credentials and acknowledgments are displayed by option 'K'. Option 'L' exits MFCTEST and takes control back to VxWorks command prompt.

2.1 Initialization functions menu

This sub-menu performs various initialization functions required on the MFC and Muon Fanout Test Card(MFTC) and their individual modules. The Initialization functions menu is as shown below:

```
+----------------------------------------------+
|                                              |
|         INITIALIZATION FUNCTIONS MENU        |
|                                              |
|      A. Reset Muon Fanout Card               |
|      B. Initialize Muon Fanout Card          |
|      C. Initialize MFC's SCLR                |
|      D. Reset MFC's HOTLink                  |
|      E. Reset Muon Fanout Test card          |
|      F. Exit to MFCTEST main menu            |
|                                              |
+----------------------------------------------+
```

2.1.1 Reset Muon Fanout Card

This function sends a software reset to the Muon Fanout Card in the crate. This action should be the same as pushing the RESET switch on the MFC's front panel. However, it is observed that current hardware logic is not quite simulating the same effect and needs to be corrected. This function now explicitly empties all FIFO banks, which when the logic is corrected will be done by the board's hardware or firmware itself.

2.1.2 Initialize Muon Fanout Card

This function stops the MFC's internal sequencer and confirms that it is really stopped and is ready. It resets MFC's Hotlink interface. MFC is switched to 'External by VME' mode. It finally initializes the Serial Command Link Receiver(SCLR) on the MFC.

2.1.3 Initialize MFC's SCLR

This function initializes the SCLR by following similar hardware procedure as it does in the previous function. In addition, this function displays appropriate bits from MFC Hotlink and control register and the error registers. The bits displayed are Ack, Err, Rdy, Sync_Err and Date_Err. The status of these bits is displayed before and after initialization, after waiting and after reset.

2.1.4 Reset MFC's HOTLink

This function resets HOTLink on the MFC. It first enabales BIST on MFTC's HOTLink and starts SYS mode of operation of the MFC. Finally the HOTLink receiver on the MFC is reframed.

2.1.5 Reset Muon Fanout Test Card

This function resets the MFTC and its internal sequencer. More operations may added under this function as the MFTC module evolves.

2.2 Memory & register checks menu

The Memory & register checks menu is as shown below. It performs various memory and register checks on the MFC and MFTC.

```
+---------------------------------------------+
|                                             |
|       MEMORY & REGISTERS CHECKS MENU        |
|                                             |
|     A. Check MFC non-volatile memory        |
|     B. Check MFC sequencer memory           |
|     C. Check MFC FIFO memory                |
|     D. Check MFC internal registers         |
|     E. Check MFTC internal registers        |
|     F. Exit to MFCTEST main menu            |
|                                             |
+---------------------------------------------+
```

2.2.1 Check MFC non-volatile memory

This function tests MFC's non-volatile memory. First it probes to check that all the NVRAM bank is physically present in the MFC. It then performs various types of write/read checks on the memory. The tests performed include those writing and checking incrementally increasing numbers in consecutive locations, running bit patterns and alternate bit patterns (such as 5's and a's). All the memory tests are non-destructive. If the memory is found to be bad, then information is displayed about the bad location, written and read data. If more than 256 locations are bad, then the memory test is aborted with appropriate message.

In addition, this function sets the last error record pointer in the NVRAM and also fills the entire error record buffer with 0xffff's.

2.2.2 Check MFC sequencer memory

This function is essentially similar to the one explained in the previous section except that it checks the MFC's internal sequencer memory instead of NVRAM.

2.2.3 Check MFC FIFO memory

Once again, the memory check algorithms used in the function are the same as those used in the previous two functions. However, this function has to handle certain issues which are specific to FIFO organization of the memory as well differences in the individual FIFO memory banks.

There are eight FIFO banks. While the Bank 1 and 3 are 8-bit by 32-word deep, bank 8 or the history bank is 16-bit by 256-word deep. Rest of the banks are 16-bit by 16-word deep. Bank 8 has bit 11 permanently tied to ground.

Accessing FIFO requires Computer Access Mode bit set in MFC control register. This is set after backing up the previous copy of the register contents. Previous data in FIFO banks is also backed up before starting the memory checks. Test are also performed while individual memory checks that the banks in deed could be emptied and filled to their designated capacity. As usual, all the memory tests are non-destructive.

2.2.4 Check MFC internal registers

As far as this test goes, MFC's internal register banks are assumed to be just another memory blocks. However, care is taken in skipping registers which are not completely write/readable. It may be pointed out that in case of some of the MFC registers, only some of the bits are write/readable or a particular pattern of data cannot written and so on. Another difference compared to the general memory check functions is that in this case, the test involving writing and checking incrementally increasing numbers in consecutive locations is only performed on the MFC internal register banks.

2.2.5 Check MFTC internal registers

Again, this function is very similar to the function described above, except that it deals with the MFTC's internal registers rather than MFC's.

2.3 MFC NVRAM configuration handler

Though these functions are not logically need to be supported by a test stand software such as MFCTEST, they are provided rather as generic diagnostic functions for the new as well as in use and broken down modules. The MFC NVRAM configuration handler menu is as shown below.

```
+---------------------------------------------+
```

```
|                                              |
|       MFC NVRAM CONFIGURATION HANDLER        |
|                                              |
|     A. Print current configuration           |
|     B. Set MFC NVRAM configuration           |
|     C. Exit to MFCTEST main menu             |
|                                              |
+----------------------------------------------+
```

2.3.1 Print current configuration

This functions displays in a suitable format, the entire MFC configuration file which is stored in its NVRAM. The display starts with geographic section ID and status followed by the configuration word in hex format. The stored checksum of the configuration file is checked with the the just computed one and the result is also displayed. Date and time of current configuration along with its author name is displayed next.

Last, but most important block of information that this function displays is the MFC configuration in a mnemonic picture format of the entire Muon readout crate. The entries include the VME module/section name, its slot number in the VME crate and its VME base address prefix. MRCs are represented by their two sections (A & B) separately. Presence or absence of the module/section is indicated by a 'one' or 'zero' respectively.

2.3.2 Set MFC NVRAM configuration

This function allows the user to setup all the entries of MFC's configuration file interactively. At the outset, the function requests the user name and the NVRAM configuration password. The password is authenticated and the function exits, if the valid password is not supplied. At this time, a prompt for the new geometric ID is requested. Then, the date and time from both the NVRAM and VME processor module are fetched and compared. Option is provided for setting the NVRAM's clock and calender with that of the processor. Lastly, provision is provided to configure the crate interactively by entering a 'one' or 'zero' for the present or absent module at the appropriate slot number in the crate, a pictorial template of which is provided for the purpose.

If any of the entries of the configuration file is changed, certain book keeping jobs are done by MFCTEST. These include updating the time stamp and author name of the configuration file, clearing the unused area of the configuration file. Fresh checksum is also computed (162bug's CS command algorithm is used for checksum) and written into the configuration file for future reference.

2.4 MFC readout modes tests menu

Functions listed under this menu offer tests on different readout modes that the MFC is designed to support. There are three main modes of MFC operation. In the internal sequencer mode, MFC mimics Trigger Frame Work(TFW) decisions using a sequence of pre-stored entries. The other two modes of operation are the TFW mode and the SYS mode in which the timing, trigger and control signals come from the TFW via SCLR or from Trigger Fanout Card via HOTLink interfaces respectively. Fourth and the final mode is called the Computer Access Mode which is set for accessing the FIFO memory banks. This mode is mentioned elsewhere in this document. This menu will expand as the MFTC evolves and currently it is as shown below.

```
+----------------------------------------------+
|                                              |
|         MFC READOUT MODES TESTS MENU         |
|                                              |
|      A. Setup and start internal sequencer   |
|      B. Stop internal sequencer              |
|      C. Perform HOTLink BIST                 |
|      D. Test MFC's SYS mode                  |
|      E. Test MFC's EXT mode                  |
|      F. Exit to MFCTEST main menu            |
|                                              |
+----------------------------------------------+
```

2.4.1 Setup and start internal sequencer

This function starts by determining whether to run in single or continuous mode from the user. It then loads the MFC sequencer memory with the TFW imitated decisions which are read into from a data file named 'inttrig.dat'. MFC is switched to internal sequencer mode and making sure that internal sequencer is stopped. The function then starts the sequencer. If a single mode operation is chosen, then the sequencer is stopped after a delay or else the sequencer is allowed to operate continuously. FIFO buffers may be examined to verify that internal sequencer has run on the data successfully. The data file 'inttrig.template' may be used to generate once own 'inttrig.dat' files and run MFC's internal sequencer on this data.

2.4.2 Stop internal sequencer

This function stops the MFC's internal sequencer and checks to see if it has indeed stopped after many attempts punctuated by some delay periods. It quits if it cann't stop the sequencer or else switches the MFC into the default test stand mode; external by VME mode. Please note that sometimes the MFC requires a couple of attempts through this function to stop the internal sequencer.

2.4.3 Perform HOTLink BIST

This function first reframes the MFC receiver with the HOTLink transmitter from MFTC. It then puts the transmitter in Built-In Self-Test mode and checks the MFC status register to see that there are no errors during the test. This function reports the test result.

2.4.4 Test MFC's SYS mode

In this test mode, a fixed trigger signals are sent over the HOTLink from MFTC to MFC, the signals (which are passed through the MFC) are sampled on the VME backplane and compared with the originally sent data by the MFTC. This function initializes HOTLink on MFTC and reframes the link between MFC and MFTC. It then starts data transmission and sit in a loop checking for completion of the hardware checking sequence, which is handled by MFTC internal sequencer. Finally, this function probes a flag maintained by MFTC to ascertain if the test is successful.

2.4.5 Test MFC's EXT mode

In principle, this function works very similar to the previous function, except that the data is transmitted over the SCLR instead of HOTLink. Details of the working of this function will evolve as the MFTC gets ready.

2.5 MFC interrupts tests menu

Functions listed under this menu again requires that MFTC is ready with its designed features and functions. So for the same reason, this menu will also have changes from time to come. It is as shown in the present form.

```
+---------------------------------------------+
|                                             |
|          MFC INTERRUPTS TESTS MENU          |
|                                             |
|     A. Setup interrupts in MFC              |
|     B. Test interrupt responses of MFC      |
|     C. Exit to MFCTEST main menu            |
|                                             |
+---------------------------------------------+
```

2.5.1 Setup interrupts in MFC

This function starts by setting up service routines for all the interrupts. Interrupts VBD done, L1Acc, L2Acc, L2Rej, SRQ, Init rise and Init fall are set on master chip interrupt vectors 0x50 to 0x56 respectively. Similarly L1Busy, L2Busy, L1 Err, L2Err, FIFO full and FIFI empty are setup on slave chip with interrupt vector 0x57 which is daisy chained to the master. It then goes to set the interrupt controller mode register for level triggering, reset interrupt controller edge flip-flop selective reset register, set interrupt controller edge polarity register for rising edge. This is followed by programming the interrupt master chip with edge-triggered, 8-byte interval, cascaded, with

ICW4. Base interrupt vector is set at 0x50 and slave connection on master's IRQ7. Master is set in fully nested, non-buffer, normal EOI, vector mode with its mask in place. Similarly the interrupt slave chip is set with its base interrupt at 0x57 and rest of the modes being similar to that of the master.

2.5.2 Test interrupt responses of MFC

There are a few interrupt generation schemes used by MFCTEST. In real system VBD done is generated by VBD. In the test stand, a VBD done bit is set in system status register of MFTC. MFC generates interrupt on this action. The ISR for this interrupt set the Slave ready bit on the HOTLink register, which can be read in through a MFTC register. This completes a cycle of checking VBD done interrupt.

L1Acc, L2Acc and L2Rej interrupts are checked by running the MFC in internal sequencer mode with psuedo TFW trigger data and capturing the respective interrupts as they are generated. The internal sequencer function in this case is run quiet mode, without generating any status messages as in the case of 'Setup and start internal sequencer' under section 2.4.1. This quiet version of sequencer function also uses a different trigger data file named 'intrtrig.dat'.

A simple scheme has been designed in the MFTC module for testing the MFC responses to interrupt signals L1Busy, L2Busy, L1Err, L2Err and SRQ. First four of these signals come from MRC in real system. A register with each bit assigned for a different interrupt is provided on MFTC. Interrupt signal is generated by flagging the corresponding bit in the said MFTC register. MFC response to this interrupt signal can then be recorded.

INIT rise and INIT fall are checked by setting the INIT-VME bit in the NOW register. FIFO full and FIFO empty interrupts are generated by writing and reading from the FIFO banks continously in CAM mode. Internal MFC error is simulated by writing erroneous data into MFC control register.

Global counters for various interrupts are maintained by MFCTEST. They are reset by this function. Each of the interrupts are taken up in turn. Interrupt is flagged as mentioned above. A wait loop is generated for allowing MFC to respond to this event. If indeed the MFC responds, then appropriate service routine setup by the previous function is activated. The service routines typically increment the corresponding interrupt number and displays the same. It resets the appropriate bits and registers for allowing further interrupts and so on. These operations could be interrupt type dependent. On the other hand, if an interrupt signal doesn't invoke any response from MFC, then appropriate message is displays after the wait period. Each interrupt type is checked with 8 iterations.

2.6 MFC data browsers menu

This menu provides functions to display various data buffers of MFC in their customized screen format. This menu is shown below.

```
+---------------------------------------------+
|                                             |
|            MFC DATA BROWSERS MENU           |
|                                             |
|      A. Display MFC FIFO buffers            |
|      B. Display MFC NVRAM error records     |
|      C. Exit to MFCTEST main menu           |
|                                             |
+---------------------------------------------+
```

2.6.1 Display MFC FIFO buffers

This functions displays contents of one or all of the FIFO banks in a convenient screen format. Part of the complication to this function arises from the fact that the data from all FIFO banks are NOT same wide and deep as mentioned in section 2.2.3. The function prompts for the FIFO bank number (0 for all banks). If one or more of the banks are empty, appropriate message is displayed. Data from banks 1 & 3 are 8-bit by 32 words. Data from these banks is packed into 16-bit by 16 words. So, data from banks 1-7 are displayed, one line per bank. Data from Bank 8 is displayed in 16-lines as the data size is 16-bit by 256 words. When all banks are opted, no data legends could be displayed due to non-graphic screen size limitations. In this case, first 7 lines are for banks 1-7 followed by the bank 8 data.

FIFO status registers' status, L1 & L2 accept and reject counters at error and total are displayed next. If bank 8 or all are opted, bank 8 entries are also displayed next with a 0x0700 mask.

2.6.2 Display MFC NVRAM error records

This function displays MFC's error records which are stored in its NVRAM. At the outset, the function checks to see how many error records are currently stored. Appropriate message is displayed, it the error record buffer is empty, else the user is prompted for the record number from the available records. The function displays the error record in a convenient screen format. The contents include data and time of error record, cumulative error number and error type and code, L1 & L2 counters and MFC registers at the time of error and finally MRC CSR registers, XNs and TNs for all the MRC sections.

2.6.3 Diagnostic utilities menu

This is one of most important menus of the MFCTEST with regard to debugging functions of MFC. Functions listed under this menu help one to localize the problem of a MFC to a section, region or even to a chip level. The diagnostic utilities menu is as shown below.

```
+---------------------------------------------+
|                                             |
|          DIAGNOSTIC UTILITIES MENU          |
|                                             |
|     A. Display MFC register contents        |
|     B. Modify MFC register contents         |
|     C. Display MFTC register contents       |
|     D. Modify MFTC register contents        |
|     E. Display a memory block interactively |
|     F. Modify memory interactively          |
|     G. Loop in a memory block read/write    |
|     H. Exit to MFCTEST main menu            |
|                                             |
+---------------------------------------------+
```

2.6.4 Display MFC register contents

This function displays the contents of 20 important MFC registers. For every register, the description of the register, its address and data in hex format are displayed. The register contents are also displayed in binary format with legends and binary bit positions displayed as the header line.

2.6.5 Modify MFC register contents

The same set of 20 registers are handled by this functions which allows the user to modify their contents. As already mentioned in section 2.2.4, it may be noted that all of these registers are not writable and in case of some of them, only some of their bits are writable. Complete set of 20 registers is prompted for modification however as a general case by this routine.

The prompt includes the description of the register, its address and its current contents. Options provided are to enter the new data in hex or binary format, to skip and goto next register or to quit the function.

2.6.6 Display MFTC register contents

This function is similar to the 'Display MFC register contents' function except for the fact that it handles MFTC registers instead of MFC's.

2.6.7 Modify MFTC register contents

This function is similar to the 'Modify MFC register contents' function except for the fact that it handles MFTC registers instead of MFC's.

2.6.8 Display a memory block interactively

This function may be used to display interactively blocks of memory from processor, MFC or MFTC modules; for that matter from any module in the VME crate. A block length of 128 words of 16-bit wide are used as default by this function. A probe for the presence of requested memory block is performed before actually trying to fetch data from this block and appropriate error message is issued in case an invalid memory address is supplied. Options provided by the prompt are to enter the next block address or choose continuation from the previous block or to quit the function.

2.6.9 Modify memory interactively

This function may be used to modify interactively the contents of a block of memory from processor, MFC or MFTC modules; for that matter from any module in the VME crate. Data is assumed to be 16-bit wide as default by this function. A probe for the presence of requested memory block is performed before actually trying to fetch data from this block and appropriate error message is issued in case an invalid memory address is supplied. Options provided by the prompt are to enter the next block address or to quit the function. Options for data entry are to enter data for the prompted location in hex, RETURN for skipping the current location or dot(.) for quitting data entry for a block.

2.6.10 Loop in a memory block read/write

This function is specifically useful while debugging a faulty MFC module. It provides a facility to loop over a block of memory in write or read cycles. The loop can be run over a given number of cycles or continuously. The function takes in the start address of the memory block, number of words to read from or write into and the required number of cycles. Possible inputs for the start address are entering a new address in hex or continue with the previous one or to quit. Valid inputs to the second prompt may be number of words in decimal to be read or written with a prefix 'r' or 'w' respectively. A 'p' input is provided for repeat with previous input. With regard to the number of cycles prompt, supplying '0' as input, makes the loop endless. Until a better scheme is worked out, the only way to break the endless loop is to employ the abort key sequence, CTRL+C. Needless to say that this will break MFCTEST in the process!

2.7 Miscellaneous functions menu

A set of functions which couldn't be classified under any of the menus mentioned above are listed under this menu which is shown below.

```
+--------------------------------------------+
|                                            |
|        MISCELLANEOUS FUNCTIONS MENU        |
```

```
|                                              |
|      A. Probe for MFC & MFTC cards in crate  |
|      B. Get MFC's ID and serial number       |
|      C. Exit to MFCTEST main menu            |
|                                              |
+----------------------------------------------+
```

2.7.1 Probe for MFC & MFTC cards in crate

This function probes and reports whether or not the MFC and/or MFTC modules are present in the test crate.

2.7.2 Get MFC's ID and serial number

This function fetches the card ID and serial number of the MFC under test from its internal register and displays the same.

2.8 Display MFCTEST log files

MFCTEST may be invoked by a 'log' command line option (mfctest "log"). It then creates a log file with the name derived from current date and the serial number of the MFC under test. All the test results and reports about this card on a given day will be written into this file. The log files may be used later for archival or other reference purposes. Previous log files may be viewed by employing this function.

The function displays all the log files from its working directory and prompts the user for a log file. It displays requested log file one screenful at a time.