# INSTRUCTION MANUAL

## MCS II

## MULTICHANNEL SCALER CARD

**Oxford Instruments Inc**
**Nuclear Measurements Group**

601 Oak Ridge Turnpike ■ Oak Ridge, Tennessee USA 37830

Telephone (615) 483-8405 Fax (615) 483-5891

MCSIIMT00F02

# WARRANTY

Oxford Instruments Inc warrants that the products or components manufactured by it shall be free from defects in material or workmanship for a period of one year from the date of delivery to purchaser. If such product or component is determined to be defective by Oxford Instruments Inc, its sole warranty obligation shall be limited to either replacing or repairing such defective product or component at Oxford Instruments' option. Such warranty is further conditional upon the purchaser giving prompt notice of any such defect and satisfactory proof thereof to Oxford Instruments' Customer Service Manager. Thereafter upon Oxford Instruments' approval, the purchaser shall return such defective product or component to Oxford Instruments' factory at Oak Ridge, Tennessee, all transportation charges prepaid. Oxford Instruments shall be responsible only for transportation charges incurred in returning such product or component to purchaser. All customs, brokerage and duty charges shall be at the expense of the purchaser. Damage in transit on items returned to factory due to inadequate packaging will be repaired at the purchaser's expense. Any repairs or replacements by the purchaser without Oxford Instruments' approval, any willful abuse or any evidence that the product or component was not properly used and maintained will automatically void this warranty.

Oxford Instruments Inc makes no warranty whatsoever in respect to products or components not manufactured by it, but instead, the applicable warranties, if any, of the respective manufacture thereof shall apply. Likewise fuses, batteries, and field-effect transistors in low-noise preamplifiers are specifically excluded from this warranty.

**SUCH WARRANTIES ARE IN LIEU OF OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO A PRODUCT AND THE ACCOMPANYING WRITTEN MATERIALS. IN NO EVENT WILL OXFORD INSTRUMENTS INC BE LIABLE FOR DAMAGES, INCLUDING ANY LOSS OF PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF USE OF OR INABILITY TO USE THE PRODUCT, EVEN IF OXFORD INSTRUMENTS INC OR AN AUTHORIZED OXFORD INSTRUMENTS INC REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. OXFORD INSTRUMENTS INC WILL NOT BE LIABLE FOR ANY SUCH CLAIM BY ANY OTHER PARTY.**

This limited warranty gives specific legal rights. Some states provide other rights, and as a result, the above limitations and/or exclusions may not apply. Furthermore, some jurisdictions have statutory consumer provisions which may supersede this section of the Agreement.

**APPLICABILITY**

If any provision of this Agreement shall be unlawful or void, then that provision shall be deemed severable from this Agreement and shall not affect the validity and enforceability of the remaining provisions of this Agreement. This Agreement is governed by the laws of the State of Tennessee.

TABLE OF CONTENTS

## 1.0 INTRODUCTION

The Tennelec/Nucleus, Inc. Multi-Channel Scaler (MCS-II) card is a flexible timer/counter with data rates to 200 MHz and near zero dead time between channels (10 nsec max). It is designed to be physically installed in most IBM-PCs, IBM-XTs, IBM-ATs, compatibles, and 386's.

The MCS card has two 24 bit 200 Mhz Counters, on board data memory, a 48 bit internal dwell timer, and a 48 bit pass counter. This card, with standard software, transforms the personal computer into a full featured Multichannel Scaler. A negative NIM or TTL input is the only external signal necessary for Multichannel Scaler(MCS) operation.

The software uses a windowing technique with pull down menus for user friendly operation. All important parameters are displayed on the monitor along with the spectral data.

A recommended minimum computer system consists of 512k of installed RAM memory, a hard disk, one floppy disk drive, a math coprocessor (optional), an EGA graphics card with an EGA monitor or VGA graphics card with a VGA monitor. The operating system should be DOS 3.0 or greater.

## 2.0 SPECIFICATIONS

### 2.1 PERFORMANCE

**MAXIMUM COUNT RATE** 200 Mhz for NEG-NIM input; 50 Mhz for TTL input.

**PULSE PAIR RESOLUTION** < 5 nsec. for NEG-NIM input; < 20 nsec. for TTL input.

**MEMORY** 8192 channel acquisition memory.

**MEMORY GROUP** Selectable in binary increments from 256 channels to 8192 channels.

**MEMORY CAPACITY** Maximum 16,777,215 counts per channel.

**TIME BASE** 10 MHz crystal controlled.

**INTERNAL DWELL** Dwell times (channel widths) may be selected between 2 usec and 325 Days.

**PRESET PASSES** 1 to 999,999,999 plus infinity.

**DEADTIME BETWEEN CHANNELS** < 10 nsec.

**DEADTIME BETWEEN PASSES** < 10 nsec.

**EXTERNAL RESTART DELAY** < 2 usec.

**DAC OUTPUT** Triangular or sawtooth.

**DISPLAY** A high resolution live display is provided through the personal computer monitor. All personal computers must have either an EGA graphics display card and EGA monitor or a VGA graphics display card and VGA monitor for proper operation of the MCS-II software.

### 2.2 REAR-PANEL CONNECTORS

**NEG-NIM DATA IN** A BNC connector that accepts negative NIM logic level pulses (14 mA into 50 ohms) as MCS data. The minimum input pulse width is 2 nsec, and the minimum pulse-pair resolving time is 5 nsec. This input may not be used simultaneously with TTL Data In.

**EXTERNAL DWELL IN** (DB9 Pin 1) When External Dwell is selected, rising edges of pulses determine channel widths. This TTL-level input requires a minimum pulse width of 200 nsec.

**EXTERNAL START IN** (DB9 Pin 2) If the unit is in acquire mode and the pass counter has not timed out, then a rising edge signal on this line resets the channel number to zero and the trailing edge initates a new MCS pass (even if the current pass has not terminated). This TTL-level input requires a minimum pulse width of 200 nsec.

**TTL DATA IN** (DB9 Pin 5) This input accepts TTL level signals as MCS data. The minimum input pulse width is 10 nsec, and the minimum pulse-pair resolving time is 20 nsec. It may not be used in conjunction with NEG-NIM data in.

**DAC OUT** (DB9 Pin 3) 12 bit resolution DAC analog output, jumper selectable as either 0 to 5 volts, or 0 to 10 volts. Software selectable as either triangular or sawtooth function.

**MSB OUT** (DB9 Pin 6) This TTL output signal is low (MSB = 0) for the first half of the selected group and high (MSB = 1) for the duration of a given pass.

**GROUND** (DB9 Pin 4) Logic and Signal Ground.

**2.3 POWER REQUIREMENTS** +5 Volts - 1700 mA
+12 Volts - 40 mA
-12 Volts - 20 mA

## 2.4 OTHER INFORMATION

**WEIGHT** (Shipping) 4.0 lbs. (1.8 kg.)
(Net) 1.0 lbs. (0.5 kg.)

**DIMENSIONS** Standard full-size IBM type XT card.

**INSTRUCTION MANUAL** One provided with each instrument ordered.

**ACCESSORY INCLUDED** None.

## 3.0 INSTALLATION

### 3.1 INITIAL INSPECTION

### 3.1.1 UNPACKING

Unpack your MCS card carefully, and save the shipping cartons with special packaging materials for reshipment, if necessary. Should you notice shipping damage, remember that only the consignee can collect damages from the carrier. Tennelec/Nucleus, Inc. will assist in establishing a claim and/or in replacing damaged equipment.

### 3.1.2 VISUAL INSPECTION

After the MCS card has been unpacked, it should be given a through visual inspection. Check to see that all integrated circuits and components appear to be sound and that the floppy disks have not been warped or bent. If damage is detected,contact the carrier and Tennelec/Nucleus, Inc.

### 3.1.3 INSTALLATION

The MCS card may be installed in your computer by the following procedure. BEFORE INSTALLATION, READ SECTION 3.1.4 SO THAT THE SWITCHES AND JUMPERS MAY BE CHANGED, IF NECESSARY, BEFORE THE CARD IS PHYSICALLY INSTALLED IN THE COMPUTER.

1) TURN OFF THE POWER to the computer to prevent damage to the computer and/or the MCS card.

2) Remove the cover from the computer. Consult the reference manual supplied with the computer for the proper instructions on removing the cover.

3) The MCS card may be inserted into any available slot.

Remove the retaining screw and lift out the blank panel from the chosen slot.

4) If needed, mount the plastic card guide in the place provided at the front end of the slot position.

5) To insert the MCS card, the panel end of the card must be tilted downward as the card is first placed into the computer slot. This allows the BNC connector to clear the top of the rear panel cutout. When the BNC connector has cleared, the rear of the board may be rotated downward until the card is parallel to the edge connector and centered in the card guide. Moderate downward pressure will now seat the MCS card in the edge connector.

6) Replace the blank panel retaining screw in the MCS card panel and tighten snugly.

7) Replace the computer top cover by reversing the procedure in step 2.

8) Reapply power to the computer and boot your operating system software.

9) When the DOS prompt appears, insert the Tennelec/Nucleus, Inc. disk 1 and type MCS, then press ENTER.

10) An initial introductory display should appear. Pressing any key removes the initial display and loads the main operating screen.

11) Proceed to SECTION FOUR for complete operating instructions.

### 3.1.4 SWITCH AND JUMPER SETTINGS

The MCS card will be shipped from the factory with the most common setting of all switches and jumpers. The optional settings are as follows.

SW1 The MCS-II card is shipped from the factory with the most common setting for the address switch, SW1. SW1 is an eight-position dip switch located above the XT bus connector of the MCS-II card. Optional address switch settings may be made with the following procedure.

When one of the SW1 switches is in the 'OPEN' position, a logic level '1' is implied, otherwise logic level '0' is implied. The

eight switches, numbered , select the base I/O port address for the MCS-II card. The MCS-II card is shipped with SW1 switches set for a base I/O port address of 528(decimal) or 210(hex).

If there is a conflict with another function in the PC at this I/O address, another address may be selected. The MCS-II card SW1 switches allow base I/O port address selection in the range from 0(decimal) through 2040(decimal) at any port address divisible by 8. A program is included on the distribution disk to assist in setting a different base port address. Type SWITCH followed by pressing <ENTER>. Enter the new desired base port address in decimal at the prompt. SWITCH then displays a simple diagram showing the settings for the dip switch SW1.

NOTE: WHEN THE ADDRESS SWITCHES ARE CHANGED, THE MCS-II SOFTWARE MUST BE CONFIGURED TO THE NEW PORT ADDRESS.

This is accomplished by typing

MCS p<base I/O port address>

and then pressing the <ENTER> key.

J1 J1 selects the DAC range. J1 is factory set for a 0 to 5 Volt range. Figure 1 shows +5 and +10 Volt range selection.

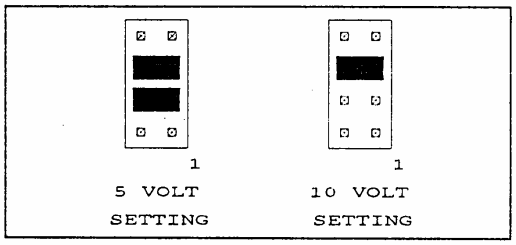



**Figure 1** DAC Voltage Range Jumper

### 3.1.5 DISTRIBUTION DISKS FILE LISTING

| File Name | Purpose |
|---|---|
| compile.exe | The MCS Batch File Compiler |
| cuser.c | C Language Source Code for User feature demo program |
| demo.nbe | Demo Executable Batch Program |
| demo.nbs | Demo Batch Source File |
| demo.spm | Demo Spectrum File (Cs-137, Co-57 & Co-60) |
| nte.exe | Batch File Editor |
| nteinst.exe | Install Program for the Batch File Editor |
| mcs.exe | the MCS data acquisition program |
| MCSedit.bat | DOS Batch File to specify what program to use to edit batch files |
| readme.doc | Documentation File for MCS-II |
| readme.exe | Program to view the Documentation File (or any text file) |
| hw.c | 'C' language program to directly control the MCS card |
| mcs.cfg | file where the system configuration is stored |
| tomcs.c | 'C' language program to convert the header in a PCA.SPM file to the MCS format |
| tomcs.exe | executable program to convert the header in a PCA.SPM file to the MCS format |

## 4.0 OPERATING PROCEDURES

### 4.1 SETUP OPERATIONS

The MCS card is supplied from the factory ready for normal Multichannel Scaler (MCS) operation. The only external input needed will be MCS data on either the NIM or TTL input lines. Refer to SECTION TWO for the required video card and monitor.

#### 4.1.1 LOADING THE SOFTWARE

The MCS software may be loaded initially in a customized state by use of the following command line parameters. e.g., To load the MCS software with no title screen and with the alternate color set in an EGA system, type the following after the DOS prompt: MCS n r. Using MCS software with Windows 386 is described in section 4.1.9.

/ICS-II command line parameters

n' No title screen.

This switch: bypasses the MCS-II title screen.

o' Ignore program overloading and skip the DOS 3.xx Check.

This switch: bypasses the check of whether MCS-II is already loaded in memory. This also removes the requirement for DOS version 3.00 or greater.

m' Set monochrome colors.

This switch: starts MCS-II in monochrome, black and white. If your display card has less than 128KB of RAM, the program will assume that you have a monochrome TTL display attached to your EGA or VGA card. MCS-II will then run in Bios video mode 15, which is specific to EGA/VGAs with attached monochrome TTL displays.

t' Set up monochrome text only display.

This switch: allows the use of an attached monochrome display adapter and CRT to display acquire time counters on the monochrome text CRT when MCS-II is run in the 'background'. For use on color systems only.

0' Set memory buffers for 512 channels.

This switch: sets full group size to 512 channels. This allows the program to run using less RAM than usual.

1' Set memory buffers for 1024 channels.

This switch: sets full group size to 1024 channels. This allows the program to run using less RAM than usual.

2' Set memory buffers for 2048 channels.

This switch: sets full group size to 2048 channels. This allows the program to run using less RAM than usual.

4' Set memory buffers for 4096 channels.

This switch: sets full group size to 4096 channels. This allows the program to run using less RAM than usual.

'a' Do not display (Alt) as a part of the pull-down menu bar on the upper left-hand corner of the MCS-II main screen.

'r' Alternate Color set.

This switch: sets the MCS-II colors to the pastel color set.

'?' Show command line options.

This switch: shows all the MCS-II command line parameters (switches). After the command screen is shown, MCS-II will exit to DOS.

's' Force Standard configuration.

This switch: causes MCS-II software to ignore the configuration file MCS.CFG.

'v' Video Reinitialization for mouse caused video problems.

This switch: resets the computer video mode to undo the change to the EGA or VGA registers by some mouse device drivers.

'T' Force use of EGA 640x350 monochrome graphics mode Use of this switch: will tell MCS-II that your computer has a monochrome TTL display attached to your EGA or VGA card. MCS-II will then run in video mode 15, which is specific to EGA/VGAs with attached monochrome TTL displays. Do NOT use this switch if your system does not have a monochrome TTL display attached to your EGA or VGA card.

'R' Skip the system RAM check.

This switch: bypasses the check of whether there is enough free RAM to run MCS-II software in. If on start-up, there is less than 64000 bytes of free RAM, MCS-II will warn you of this condition. If free system RAM falls below 58000 bytes (free RAM is shown in the upper right hand corner of the MCS-II main screen) features such as help screens or peak search can cause MCS-II to abort to DOS with an out of RAM

(fatal) error. Note that MCS-II will first save the contents of the MCS card to a disk file before exiting to DOS with any kind of fatal error.

**'h'** Keep more of the MCS-II software resident when exiting MCS. If you plan to make heavy use of the 'Edit a Batch File' or 'Compile a Batch File' features, you may find that the use of this switch will speed up your operations. 84k of RAM memory is used (in bytes) when this switch is used. When this command parameter is used, it does not matter what disk drive and/or directory you are in when 'exit' is typed to return control to the memory resident MCS-II program.

**'p<port>'**
Where port is the I/O port to force operation of a MCS-II card. This switch will force MCS software to operate on a MCS-II card using 'port' as the base I/O address of the MCS-II card. If this value 'port' does not correspond to the base address of a MCS-II card in your computer, your computer will exit saying, "No MCS card has been found."

**'y'** This switch reverses the vertical movement of the mouse.

**'z'** This switch allows an unconditional screen dump which may, under certain conditions crash the system. See section 4.4.5 for more information on screen dumps.

**'B'<filename>**
Run a batch process named filename, then exit to DOS. see section 4.6 for more information on batch processes.

**'b'<filename>**
Run a batch process named filename. See section 4.6 for more information on batch processes.

**'g'** This switch keeps MCS-II software from hooking up its screen dump code to the print screen key allowing special third party screen dump programs to operate correctly.

The MCS software uses a pull down windowing technique to set parameters and perform most operations. The remainder of this section will be used to describe the window menus.

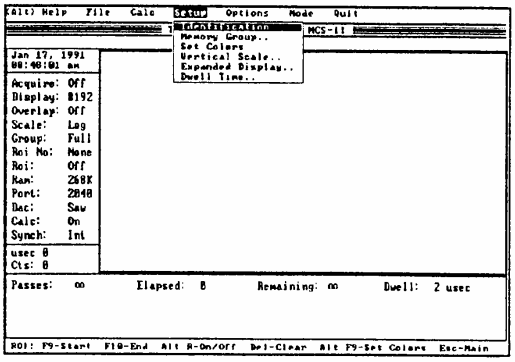



**Figure 2** Identification Code

### 4.1.2 IDENTIFICATION CODE

To reach the Identification Code selection, hold the ALT key down and press S for setup. Use the Arrow keys to highlight Identification and then press the ENTER key to set the Identification Code. You may now type up to 70 characters or letters as an Identification Code (See Figure 2). Function key F5 also accesses this feature.

### 4.1.3 MEMORY GROUP SELECTION

The Memory Group menu is selected by first entering the Setup window by holding down on the ALT key and pressing S for Setup. Using the Arrow keys, highlight the Memory Group line and press ENTER. If you choose any group other than Full, it is necessary to choose the division or section of the selected group (See Figure 3) Press the ENTER key after each selection.

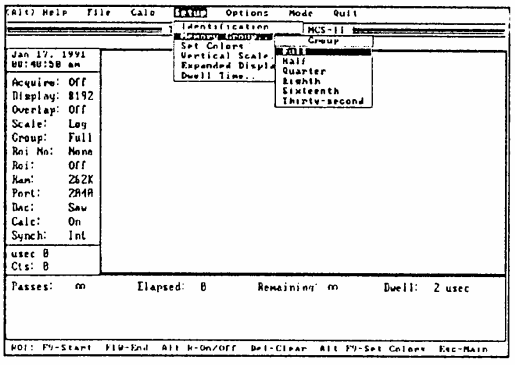



**Figure 3** Memory Group Selection

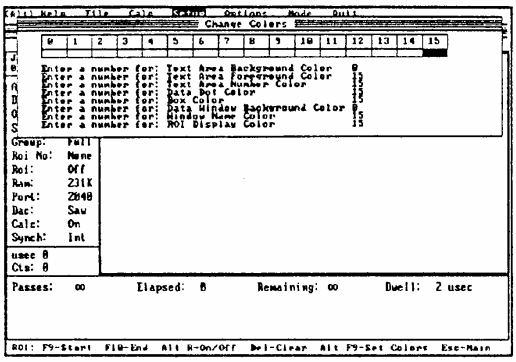


**Figure 4** Setting Display Colors

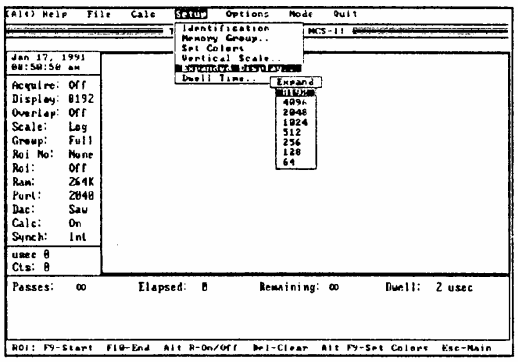


**Figure 6** Expanded Display Selection

### 4.1.4 SETTING DISPLAY COLORS

Choose the Setup window by pressing ALT S . Select the Set Colors mode by using the arrow keys to highlight the Set Colors line and then pressing ENTER. A 15 column set of colors will be displayed and you will be asked to select a color number for each area or text of the display. The present color number will be shown and you may keep this number by pressing ENTER or you may delete this number and type the new selection and then press ENTER (See Figure 4).

### 4.1.5 VERTICAL SCALE SELECTION

Select the Setup window by pressing ALT S . Move the highlighted line to Vertical Scale with the arrow keys. All possible vertical scale selections will be shown. Using the arrow keys, highlight the scale of choice and then press the ENTER key (See Figure 5). The UP and DOWN arrow keys provide a fast entry to the Vertical Scale selection.

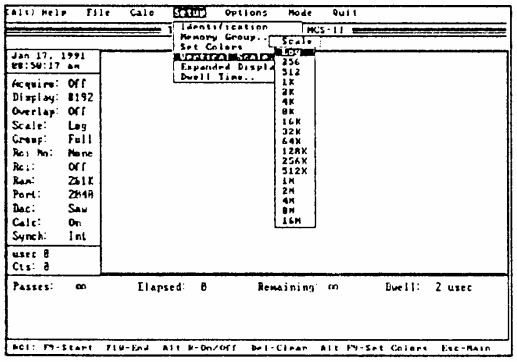


**Figure 5** Vertical Scale Selection

### 4.1.6 EXPANDED DISPLAY OPERATION

Press ALT S to select the Setup window. Use the arrow keys to highlight the Expanded Display line and then press ENTER . All possible full scale expansions will be shown. Choose the required expansion with the arrow keys and then press the ENTER key (See Figure 6). Function key F4 provides a fast entry to the expanded display mode.

### 4.1.7 CURSOR OPERATION

The cursor is maneuvered across the spectral display by the left and right arrow keys and the page up and page down keys. The lower part of the display parameters show the channel where the cursor resides and the number of counts in that channel. A list of the cursor commands is shown:

**Right Arrow** Move the cursor one channel to the right.

**Left Arrow** Move the cursor one channel to the left.

**Page Up** Move the cursor right 1/32 of the display.

**Page Down** Move the cursor left 1/32 of the display.

**Home** Move the cursor to first channel in window.

**End** Move the cursor to last channel in window.

**Ctrl Home** Move the cursor to channel zero.

**Ctrl End** Move the cursor to the last channel.

**Ctrl PgUp** Move cursor to next higher ROI.

**Ctrl PgDn** Move cursor to next lower ROI.

### 4.1.8 DWELL SELECTION

Press ALT S to select the Setup window. Use the arrow keys to highlight the Dwell line and then press ENTER . Time Units, Time Multiplier, and External will be displayed.

Time Units - All possible units for the internal dwell time prescaler will be shown. Choose the required units with the arrow keys and then press the ENTER key.

Time Multiplier - This selection allows the user to enter the amount of the selected units 1 - 65534. For example: If the time units are DAYS. Entering 200 as the time multiplier would select a dwell time of 200 days.

External - This selection allows an external source to generate the dwell time for the MCS board. The internal prescaler is disabled during this mode of operation.

### 4.1.9 WINDOWS 386

MCS software was not written to be a 'Windows' application.

MCS and Windows 386 do conflict with each other in one known area, graphics. This is because the assembly language graphics functions in MCS software use the EGA (or VGA) hardware registers (for speed).

Two conditions must occur for the conflict to show up:

1) MCS is running as a background task under windows 386.

2) And MCS is operating as a batch interpreter (running a batch process) that puts the system into Acquire mode.

The result is that EGA graphics code that is executed when MCS goes into Acquire mode causes Windows to suspend execution of the MCS program!

Here is what Windows 386 is good for:

1) Turn acquire on while MCS is a foreground task.

2) Exit to the MS-DOS Executive.

3) Shell COMMAND.COM.

Now, you can acquire MCS data totally in the background, with about 600 KB of free ram available to run other programs. MCS will be running in it's own 640KB page and other programs will use in their own 640KB pages. To do this will require a 386 type computer with 2MB or more of RAM.

The file MCS.PIF is provided to run MCS under Windows 386.

## 4.2 KEYBAR COMMAND OPERATIONS

The Keybar is the command ruler line at the bottom of the screen.

### 4.2.1 ACQUIRE AND STOP OPERATIONS

The first function key F1 starts and stops the acquisition of data. When the MCS card is acquiring data, the first position in the keybar will read <F1-Stop> and the first line on the display parameters will read <Acquire: On>. Pressing the F1 key will toggle the acquisition of data between On and Off.

### 4.2.2 ERASE DATA

The Erase function is a two key command for the purpose of eliminating accidental erasure of important data. Erasure of all displayed data is accomplished by holding the CTRL key down and pressing F2.

### 4.2.3 PRESET PASSES

Pressing the F3 key allows the user to select the maximum number of passes to allow through the selected memory group. Enter the passes desired and press ENTER. The preset passes will be shown below the bottom of the spectra area.

### 4.2.4 EXPAND MODE

The MCS display may be expanded to a full scale display of 64 channels. Pressing the F4 key will prompt a listing of all available expansion

;elections. Move the highlight to the selection desired and press ENTER to do the expansion. The expanded area will be shown at the bottom of the spectra area and the full spectra will be displayed at the top of the screen in compressed form.

### 4.2.5 IDENTIFICATION CODE

An Identification code of up to 70 characters may be entered when the F5 key is pressed. Type in the code when the prompt appears and then press ENTER.

### 4.2.6 LOAD OPERATION

Data that has been saved on disk may be loaded into the MCS spectral memory. Press F6 and you will be prompted to type in the File Name of the data that you wish to load. Typing in the file name and pressing ENTER will load the data.

### 4.2.7 SAVE OPERATION

Acquired data may be saved to disk for reloading at a later time or for back up purposes. When F7 is pressed, you will be prompted to type in the file name that you wish to give the data to be saved. When the file name has been typed, press ENTER to save the data.

### 4.2.8 ROI OPERATION

Pressing the ESC key changes the keybar command line to ROI commands. To set a Region Of Interest, move the cursor to the first channel of the desired ROI and press F9. Next, move the cursor to the last channel of the desired ROI and press F10. All channels between and including these two end channels are set as a Region Of Interest. To display the ROI channels, press ALT R. The ALT R keys function as a toggle to turn the ROI's on and off. The Delete key will erase the ROI. Press Ctrl F8 to delete all ROI's. To change the color of the ROI's, press the ALT F9 keys. Select the color desired and press ENTER. Pressing the ESC key again will return the keybar commands to the main menu.

## 4.3 MODE OPERATIONS

### 4.3.1 TRANSFER BETWEEN GROUPS

Data may be transferred between like memory groups with the smallest memory group being 256 channels. To do a Transfer operation, choose the Mode window by pressing ALT M. Move the highlight to the Transfer line by using the arrow keys. Press ENTER and then choose the group size that is being transferred. When the group has been chosen, you will be asked to choose the source division (the section that is being transferred) and then the target division (the section that is receiving the transfer). A prompt will be placed on the display describing the transfer that is to take place and asking if you wish to continue with the transfer. Type either Y for yes or N for no to complete the procedure (See Figure 7).

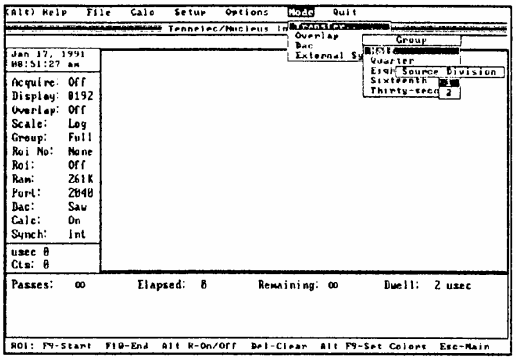



**Figure 7** Group Transfer Operation

### 4.3.2 OVERLAPPING MEMORY GROUPS

Sections of memory may be overlapped for comparison purposes. All available groups will be overlapped by this procedure. i.e. when in quarter groups, all four quarters will be overlapped on the display. To do an overlap operation, Select the Mode window by pressing ALT M. Move the highlight to the Overlap line with the arrow keys and press ENTER. To remove the overlap, enter the mode window again and select the No Overlap line and then press ENTER.

### 4.3.3 DAC OPERATION

Press ALT M to select the Mode window. Use the arrow keys to highlight the DAC line and then press ENTER.

Sawtooth mode - this mode causes an analog DAC output to begin at zero Volts and increase to the maximum set DAC Voltage as the channels advance through the selected group.

MSB mode - this mode causes the MSB output to be low during the lower half of the selected group channels and high during the upper half.

Triangle - this mode causes an analog DAC output to begin at zero Volts and increase to the maximum set DAC Voltage at the midpoint of the group then decrease back to zero at the end of the group.

### 4.4 FILE OPERATIONS

#### 4.4.1 LOADING A BINARY FILE

Select File window by pressing ALT F . Select the Load Binary File by using the arrow keys to highlight the first line. You will be asked to provide the file name. Type in the file name and then press ENTER to load the file (See Figure 8). Data is loaded into the current displayed memory group.

#### 4.4.2 PRINTING A DATA FILE

The data may be output to a printer from several menu commands. e.g. ROI Report, ASCII File Save, ROI ASCII File Save. To output the file to a printer, type PRN at the filename prompt.

#### 4.4.3 SAVING A BINARY FILE

Press ALT F to select the file window. Move the highlighted line to the Save Binary File with the arrow keys. You will be asked to provide the file name. Type in the file name and then press ENTER to save the file to disk. Data in the current displayed memory group is saved to disk.

#### 4.4.4 FILE DIRECTORY

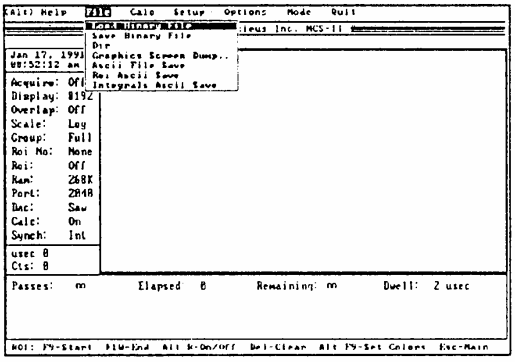



**Figure 8** Loading Binary File

The File window is selected by pressing ALT F . Select DIR by moving the highlighted line to DIR with the arrow keys. Press ENTER and you will be asked for a file mask. You may enter global characters (i.e., *.* or *.spm) or a particular file name. The Binary file may be loaded by highlighting the file with the arrow keys and then pressing ENTER.

#### 4.4.5 GRAPHICS SCREEN DUMP

Select the File window by pressing ALT F. Using the arrow keys, highlight the Graphics Screen Dump. Pressing ENTER will display the four selections for size or type of screen dump. Select the size or type or choose to save to disk and then press ENTER. Pressing the Shift PrtSc keys also selects this feature.

1) Standard Size:
   This is the smallest and fastest way to save the contents of the MCS-II screen to a printer. The printed image measures: 5.5 inches wide by 4.5 inches tall.

2) Double Wide:
   This method of screen dump generates an image that is 7.8 inches wide by 4.5 inches tall.

3) Sideways:
   This screen dump method generates a sideways image that very nearly fills a whole sheet of paper. The printed image measures: 9.0 inches by 6.0 inches.

4) Disk File:
   This option writes a disk file of the colored screen image. The file format used is called PCX. This is the file format used by the graphics program PC Paintbrush+. PC Paintbrush+ is produced by Zsoft Corporation, 450 Franklin Road, Suite 100, Marietta, GA 30067 [(404) 428-0008]. With PC Paintbrush+, a color image of the MCS-II screen can be edited in almost any way imaginable, and saved to disk or printed using any one of 74 printer or plotter types.

#### 4.4.6 ASCII FILE SAVE

Press ALT F to enter the file window. Highlight the ASCII File Save by using the arrow keys. Press ENTER and you will be asked to enter the file name. Type in the file name and press ENTER to save the spectrum to an ASCII file or device.

### 4.4.7 ROI ASCII SAVE

Select the File window by pressing ALT F. Move the highlighted line to Roi ASCII Save by using the arrow keys. Press ENTER to save the region of interest data to an ASCII file or device. An ROI must be defined for this function to operate.

### 4.4.8 INTEGRALS ASCII SAVE

The File window is selected by pressing ALT F. Select the Integrals ASCII Save line with the arrow keys and then press ENTER. The Integrals data may be saved to an ASCII file or device. A ROI must be defined before selecting this mode.

### 4.4.9 MOVING DATA TO SPREADSHEETS

The following options from the File pull-down window will generate comma delimited ASCII text files that can be read by popular spreadsheet programs.

a) ASCII File Save
b) ROI ASCII Save

The comma delimiters allow the spreadsheet program to separate the delimited test into spreadsheet rows.

Using the Quattro spreadsheet program to load an ASCII file.

1) call up the File Menu with: '/F'
2) call up the Import Menu with: 'I'
3) call up the Comma Delimited load feature with: 'C'
4) enter the filename that you saved with MCS-II then press the return key.

The Quattro spreadsheet program is produced by:

Borland International Inc.
1800 Green Hills Road
P.O. Box 660001
Scotts Valley, CA 95066-0001

## 4.5 CALCULATION OPERATIONS

### 4.5.1 SMOOTHING DATA

Enter the CALC window by pressing ALT C . Using the arrow keys, highlight the Smooth Data line and then press ENTER. You will be given 4 smoothing choices. Select one by using the arrow keys and then press ENTER (See Figure 9). Save the raw data on disk before smoothing if you will have a need for the raw data later. Smoothing destroys the original data.

### 4.5.2 CALIBRATION

To perform a Time Calibration, first acquire a spectrum and set Regions of Interest around the calibration area. Select the Calc window by pressing ALT C. Choose the Calibration line by using the arrow keys and then press ENTER. Choose the units of calibration (i.e. usec, msec) and then press ENTER. Set the cursor in the ROI of the first calibration area and then press ENTER. You are then asked to enter the data time in the proper units. When you press ENTER you will be asked to locate area 2 by moving the cursor to the ROI in which the second calibration area is located. Press ENTER and you will be asked to ENTER the time of the second area. Continue until all area times have been entered and then press ENTER again to complete the time calibration procedure.

### 4.5.3 ROI REPORT

A ROI Report may be made to the display screen, a printer or a disk file. Select the CALC window by pressing ALT C. Choose the ROI Report line by moving the highlight with the arrow keys and pressing ENTER. You will be prompted for a file name. Type EGA to send the report to the video display or type PRN to send the report to a printer. Press ENTER to complete the procedure. Entering anything besides EGA or PRN will generate an ASCII text file with that name.

### 4.5.4 SPECTRUM STRIPPING

A Spectrum Stripping operation may be performed between any like groups of the spectrum. Select the CALC window by pressing ALT C. Use the arrow keys to select the Strip line and press ENTER. You will be asked to choose the group or section of memory that you wish to operate with. Make this choice and press ENTER and you will be asked for the source division (the section that will be subtracted from the target division). Select the source and press ENTER and you will be asked to choose the target division. Make this choice and press ENTER. A prompt will be displayed to ask if you wish to perform an addition or subtraction. Make your selection and press ENTER to complete the operation (See Figure 10).

## 4.6 OPTIONS OPERATIONS

### 4.6.1 CALL USER PROGRAM

This feature allows any executable program to be run from inside MCS-II. For example: the DOS utility program CHKDSK could be run if you suspected a problem with a disk drive.

The primary purpose of the Call User Program option is to execute user programmed software that reads memory on the MCS Card.

The prompt asks for the name of the executable file to run.

### 4.6.2 EDITING A BATCH FILE

This feature is provided to allow batch process source files to be edited by the editor specified in the DOS BATCH file called MCSEDIT.BAT. The standard editor is NTE.EXE (from distribution disk 1). This editor has a WordStar Style command set and pull-down/pop-up windows to make it easy to use. When this feature is selected, you will be prompted for the name of the batch file to be edited.

To use your own editor instead of NTE.EXE, simply edit the DOS BATCH File MCSEDIT.BAT to specify the name of your editor. Make sure that your editor produces pure ASCII text files so that the batch process compiler can read your batch process source files correctly.

The user will be aware of three steps in the implementation of a batch process. First, a user must prepare a source file containing a set of instructions describing a process. This file will contain only ASCII text and will use the command syntax described later in this document. The suggested convention is that the source file use the extension '.nbs'. Secondly, the source file must be compiled or translated to an object file containing a machine-oriented description of the batch routine. This file will have the extension '.nbe'. And, finally, the user causes the batch instructions to be executed by running the object file.

Each of these steps will be implemented by the option pull-down menu in MCS-II. The preparation of the source file requires a text editor, for this purpose the Tennelec/Nucleus, Inc. provides a Wordstar clone text editor (NTE.EXE) as part of the MCS-II package. Editing and compiling also may be done external to MCS-II, using any text editor and the batch compiler provided on the MCS-II distribution diskette.

The first two steps, editing and compiling, need only be done once for a single batch process. Once the file is successfully compiled and checked for correct execution, it may be run any number of times. The completed object file also may be copied to other MCS-II systems using the same card and run there; therefore in some situations it may be desirable to have a more technically oriented person handle the editing and compiling of the batch file and distribute it to end users. The end user would only execute the process, and could be shielded from more detailed knowledge of the batch processing facility.

The following notation is used in describing the MCS-II batch commands.

CAPS Capital letters indicate portions of statements or commands that must be entered exactly as shown.

Capitalization is being used here as part of the syntax notation, in an actual command file these items are accepted on a case-blind basis.

In a few instances, additional characters are printed in lower case. These additional characters are optional, but may be included to make your batch command file more readable. These characters are for the user only, the command interpreter does not consider them in interpreting a command.

< > Angle brackets indicate user-supplied text. You must type in the entry defined by the text, for example <filename>.

[ ] Square brackets indicate that the enclosed entry is optional.

In addition, the following command parameters are defined.

string an ASCII text string enclosed in single or double quotes.

filename [drive][path] <name> [extension]
A filename specifier is limited to 64 characters.

MCS also provides a variable filename (%F) that may be updated by the batch process. The variable filename is indicated by the keyword FILENAME. It is null at the start of each batch process and must be initialized by the FILENAME command described below.

witch ON I OFF

J an integer parameter, integers are made up of the digits 0 through 9, exclusively.

hannel
a marker position 0 to 8191.

roup a memory group of the form <size><group>.

| SIZE | GROUP |
|---|---|
| F for full | 1 |
| H for halves | 1-2 |
| Q for quarters | 1-4 |
| E for eighths | 1-8 |
| S for sixteenths | 1-16 |
| T for thirty-seconds | 1-32 |

Examples are F1, H2, etc.

Note: If the group is not specified, a default of 1 will be assumed.

Jnless otherwise stated, the following rules apply n interpreting your command file.

) data is accepted case blind
) tokens are delimited by at least one blank space, additional spaces are acceptable
:) commands are accepted one per line and must not span lines

COMMANDS

I) **ID <idstr>**

idstr string of 0 to 71 characters representing identification code for current spectrum.

2) **SCALE <N I LOG>**

N counts full scale represented by

2(expN) * 256

N can range from 0 to 16 representing vertical scales from 256 to 16777216

3) **SELECT <group>**

Select the memory group for display purposes.

4) **ERASE**

Erase the current memory group. If necessary, the erase lock feature will first be disabled.

5) **EXPAND <switch> [<width>]**

switch toggle expansion on and off
width number of channels to be shown in expansion display ( valid values will be matched to MCS-II development )

6) **ROI <mode> [<region>]**

mode ON display regions of interest

OFF remove ROIs from display

SET include the specified region in the ROI

CLEAR clear all ROIs

region <start> <end> I ALL

start N value indicating first channel to include in ROI
end N value indicating last channel to include in ROI

7) **OVERLAP <switch> [<N>]**

switch toggle overlap display

N offset overlapped displays ( check on range for this value )

8) **TRANSFER <source group> <target group>**

9) **MCS <passes> <time> <units>**

passes number of passes

time time multiplier (1 to 65534 or 'e') (e is for external dwell)

units time units
MICRO (microseconds)

| | |
|---|---|
| MILLI | (milliseconds) |
| SECO | (seconds) |
| MIN | (minutes) |
| HR | (hours) |
| DAY | (days) |

10) **DAC <mode>**

where mode is equal to:

SAW = sawtooth
MSB = MSB
TRI = triangle

11) **SMOOTH <N>**

N number of points used in smoothing algorithm (N = 3, 5, 7 or 9)

12) **STRIP <grp1> <grp2> <mode> <percent>**

grp 1 memory group to be stripped
grp 2 memory group to be used for stripping

NOTE: group 1 and 2 must be compatible in size.

mode ADD strip by adding
SUB strip by subtracting

percent integer 1 - 99, percentage of group used to do strip

13) **REPORT PEAK <filename>**

filename where to store peak report

14) **SAVE <filename> [OVERWRITE]**

filename where to store screen data

OVERWRITE previously existing file of the same name is over-written, the default is to protect existing files

15) **LOAD <filename>**

Load spectrum from specified file. If an extension is not included in the specified filename, '.spm' is assumed.

16) **I/O <N> [<target>]**

function number as follows:

1 load binary file
2 save binary file
3 save ASCII file
4 save roi ASCII
5 save integrals ASCII
6 graphics screen dump disk file
7 graphics screen dump standard size
8 graphics screen dump double wide
9 graphics screen dump sideways

target <filename> output to specified file
< device > output to specified device, where device may be:

LPT1
LPT2
PRN
COM1
COM2
AUX

Screen devices are not supported from the batch processor, screen output implies an interactive session.

I/O options 7 through 9 send all output to the printer (LPT1) and do not accept a target parameter.

17) **SHOW <string> [TONE]**

This command displays a message to the user. A special screen area within MCS-II is designated for the batch display area. Messages up to 70 characters long are allowed.

The string may contain a status update or be a request for user action. The optional keyword TONE indicates a tone signal should accompany the message.

18) **CONFIRM <string> [TONE]**

similar to message command above except that the user must press a key to continue i.e.

"Place next sample, press any key to continue"

19) **MARKER <channel>**

channel place marker in this position

20) **LOOP BEGIN <count>**
**LOOP END**

This is a simple loop command, i.e. no nesting.

1) **STEP MARKER <direction> <N>**

direction BACK
FORWARD

N move the marker forward or backward N channels

NOTE: stepping the marker past the current display bounds is considered an error

2) **STEP ID <N>**

N increment the spectrum ID by N

NOTE: The current spectrum id will be trimmed of trailing blanks, then the increment applied. If a non-digit character is encountered before the increment is fully applied, a fatal error will be reported. For example:

June13_00000 can be incremented up to June13_99999

June13_testdata cannot be incremented

3) **FILENAME <filename>**

Initialize the filename variable.

The filename variable may be used with the LOAD, SAVE, and I/O commands. It is incremented by STEP FILENAME.

4) **STEP FILENAME <N>**

N increment the filename variable (%f) by this value

Trailing digits in the current filename are incremented. No trailing digits are taken as a 0 value. Overflow beyond a legal file specifier is considered a fatal error. For example, if we are incrementing by 1 then

file.spm ----> file1.spm
filexxx9.spm ----> filexxx10.spm (fatal error)

5) **TIME PAUSE <N>**

N The number of seconds to pause. Range: 1 to 999,999

A status message will be displayed over the program title bar. To break out of the TIME PAUSE feature, press the escape key.

26) **WAIT**

Wait for preset time to elapse.

27) **CALL <program[parms]>**

program name of spawn program, naming convention is similar to <filename> but should be name of an executable program, i.e. '.exe', '.com', or '.bat'

parms program parameters delimited by spaces, up to 4 may be specified

The called program should return a 0 value (unless a value is specifically reported, 0 is returned ). A nonzero value indicates an error condition and batch processing will be terminated.

28)
**COLLECT <string> <parm> [TONE] [LEN maxln]**

This command displays a prompt string, and collects a string to be assigned to the parm variable. The optional keyword TONE indicates a tone signal should accompany the prompt.

string prompt string, i.e. "Enter sample ID"

parm batch string variable %1 through %4 or the filename parm %F

maxln option maximum length for collected data, if not specified a default of 80 will be assumed

For example, a batch file might contain:

```
COLLECT "Enter spectrum id: " %1 TONE
ID %1
SAVE %1 OVERWRITE
```

29) **Comment Lines**

A comment is delimited by a semicolon. The comment may be on its own line or follow a valid command on the same line. When included on a command line, at least one space

should delimit it from the end of the command text. You may wish to examine the sample batch file 'demo.nbs' on disk 2 of MCS-II.

### 4.6.3 COMPILING A BATCH FILE

This feature is provided to convert batch process source files into batch process executable files. Mostly this is done so that the syntax of the commands in your source file can be checked for correctness. If there is a problem, the batch compiler will write a message about the nature of the syntax error and on what line the error was located. When this feature is chosen, you will be prompted for the name of the batch file you wish to compile.

The user will compose the source batch file, an ASCII text file made up of the batch commands described below. The suggested convention is that these files have the extension '.nbs'. These files are compiled to produce an object file. The object file will have the extension '.nbe'.

The compiler program will accept the MCS card size as a parameter, if it is not provided as a parameter, the user will be prompted. When MCS-II spawns the compiler, it will provide the size as a parameter.

The compiler will check for syntax errors and also check for functionality errors as possible (i.e. bounds checking appropriate to MCS card size, etc.). Parsing will continue until completion or until a threshold number of 10 errors are detected. Errors are reported to the file error.msg and the screen; each error message will include the line number and text in error and a message, The compiler will return a code indicating whether compilation was successful.

### 4.6.4 RUNNING A BATCH FILE

This feature is provided to run a successfully compiled batch process file.

The prompt asks for the name of the batch executable file to run.

'B<filename>' Run the batch process called 'filename' then exit to DOS.

An example :

MCS n BDEMO.NBE

The command line above runs MCS-II, skipping the title screen, running the batch process DEMO.NBE and then exiting back to DOS.

'b<filename>' Run the batch process called 'filename'.

Here is an example to show how to run a MCS-II batch process from the DOS command line or a DOS batch file:

MCS n bDEMO.NBE

The command line above runs MCS-II, skipping the title screen, and then running the batch process DEMO.NBE.

NOTE: <filename> must have the ".NBE" extension

The following features are important to the execution of a batch file:

1) Batch interruption. The Control-Break keys are designated for early termination of a batch process. Between commands, the keyboard will be checked.

2) All errors are currently considered fatal to the batch process. Since batch files are run repeatedly, once they are debugged errors should not occur. When an error does occur, the relevance of the remaining batch operations is in question.

## 4.7 SPECIAL COMMAND OPERATIONS

### 4.7.1 HELP COMMAND

The Help Window may be accessed at any time by pressing the ALT H keys.

### 4.7.2 QUIT COMMAND

To select the Quit window, press ALT Q. You will be given the choice of making MCS II resident or just exiting to DOS. Make a selection by moving the highlight with the arrow keys and then pressing ENTER. To exit while acquiring data, choose the memory resident option.

### 4.7.3 ESC COMMAND

The ESC key is used to change the keybar commands to ROI commands normally.

### .7.4 CTRL HOME

he Ctrl Home keys may be used to move the ursor to channel 0 at any time. This is true when he display is in normal display or expanded mode.

### .7.5 CTRL END

he Ctrl End keys may be used to move the cursor o the last channel in the spectral display. The lisplay may be in normal or expanded mode when his command is performed.

### .7.6 MOUSE COMMANDS

he screen is divided into two activity areas for nouse operation the spectral window and the nenu window. The spectral window is the ectangular area containing your spectrum display. he menu window is the area external to the pectral window and include both the pull down nenu bar at the top of the screen and the key-bar nenu at the bottom of the screen. To move the nouse between activity areas, click the right or niddle button.

Mouse Operation in the Spectral Window:

The mouse pulls your cursor along the spectrum.

Mouse operation in the Menu Window:

a) press the left mouse button to select an item for example to select an option from the pull-down or key-bar menus ( equivalent to the RETURN key )

b) press the right mouse button to back up one level for example to close a pull-down menu and return to the menu window ( equivalent to the ESCAPE key )

The mouse that has been used the most with MCS-I is the Logitech 3 button type (either Bus or Serial tyle).

### 1.8 COMMAND KEY SUMMARY

A condensed listing of standard key commands ollows:

**Right-Arrow** Move the cursor one channel to the right.

**Left-Arrow** Move the cursor one channel to the left.

**Up-Arrow** Increase the vertical scale. Increase display offset.

**Down-Arrow** Decrease the vertical scale. Decrease display offset.

**Page-Up** Move the cursor right 1/32 of the display.

**Page-Down** Move the cursor left 1/32 of the display.

**Home** Move the cursor to first channel in window.

**End** Move the cursor to last channel in window.

**Ctl-Home** Move the cursor to channel zero.

**Ctl-End** Move the cursor to last channel.

**Esc** Toggle between menu key bars. Aborts long calculations.

**Ctl Page Up** Move cursor one ROI to the right.

**Ctl Page Dn** Move cursor one ROI to the left.

**Ctl Left Arrow** Move cursor one pixel column to the right.

**Ctl Right Arrow** Move cursor one pixel column to the left.

**F1** Toggle data acquisition.

**F2** Toggle erase enable.

**Ctl F2** Erase spectrum if erase enabled.

**F3** Preset Passes.

**F4** Select expanded display mode.

**F5** Enter a spectrum identification code.

**F6** Load a binary spectrum file into memory.

**F7** Save a spectrum as a binary disk file.

**F9** Start ROI in current channel.

**F10** End ROI in current channel.

**Alt F9** Select ROI display colors.

**Ctl F8** Clear all ROIs.

**Del** Clear ROI containing current cursor channel.

**Alt R** Toggle ROI display mode.

Pull down Menu Keys

**Right Arrow** Move one menu to the right.

**Left Arrow** Move one menu to the left.

**Up Arrow** Move up one menu choice.

**Down Arrow** Move down one menu choice.

**Page Up** Move to first menu choice.

**Page Down** Move to last menu choice.

**Home** Move to first menu choice.

**End** Move to last menu choice.

**Esc** Exit pull-down.

**Return** Select current menu choice.

Line Editor Keys

**Right Arrow** Move the cursor one character to the right.

**Left Arrow** Move the cursor one character to the left.

**Home** Move to first character position.

**End** Move to end of edit data.

**Insert** Toggle between insert and overlay modes.

**Ctl Y** Delete entire line of data.

**Back Arrow** Destructive backspace.

**Del** Delete current character.

**Esc** Exit editor; void current work.

**Return** Exit editor; accept current work.

### 4.9 TENNELEC/NUCLEUS TEXT EDITOR

The Tennelec/Nucleus, Inc. Text Editor (NTE) is a Text Editor that has been modified to edit the MCS-II batch files. The command set for NTE is the same as that of Micropro WORDSTAR ver 3.30. (WORDSTAR is a registered trademark of Micropro International Corp.) In addition, most editor features can be accessed via the pull-down menu system (invoked with F6). The editor features: pull-down windows, fast memory mapped screen output, and the ability to operate on files as large as 240,000 bytes.

NTE is started from the DOS prompt by typing in "NTE" and pressing RETURN. Any text file can then be loaded by using the 'FILE' pull-down menu.

NOTE: NTE does NOT produce backup or '.BAK' files before saving the current work file, so never operate on files on the distribution diskette.

TEXT EDITOR COMMAND SUMMARY:

**F1** Save File
**F2** Exit
**F4** Read File
**F6** Pull Down Menus
**F7** Start Block
**F8** Copy Block
**F9** Delete Block
**F10** End Block

**Ctrl O Ctrl A** Go to Column NOW
**Ctrl O Ctrl B** SET Go to Column
**Ctrl O Ctrl C** Center Line
**Ctrl O Ctrl G** Go to Window
**Ctrl O Ctrl I** Jump to Column
**Ctrl O Ctrl J** Jump to Marker
**Ctrl O Ctrl L** Left Margin
**Ctrl O Ctrl M** Set Marker
**Ctrl O Ctrl N** Jump to Line
**Ctrl O Ctrl O** Open Window
**Ctrl O Ctrl R** Right Margin
**Ctrl O Ctrl W** Wordwrap
**Ctrl O Ctrl Y** Close Window

**Ctrl Q Ctrl A** Find/Replace
**Ctrl Q Ctrl B** To Top of Block
**Ctrl Q Ctrl F** Find String

| | |
|---|---|
| **:trl Q Ctrl I** | Autoindent |
| **:trl K Ctrl B** | Begin Block |
| **:trl K Ctrl C** | Copy Block |
| **:trl K Ctrl D** | Get User Color |
| **:trl K Ctrl E** | User Color the File |
| **:trl K Ctrl F** | User Color - Line |
| **:trl K Ctrl K** | End Block |
| **:trl K Ctrl R** | Read Block from a File |
| **:trl K Ctrl T** | Set Tab Width |
| **:trl K Ctrl V** | Move Block |
| **:trl K Ctrl W** | Write Block to a File |
| **:trl K Ctrl Y** | Delete Block |
| **:trl P** | Insert ASCII Code |
| **:trl PgUp** | Top of Window |
| **:trl PgDn** | Bot of Window |
| **lome** | Column 1 |
| **:nd** | End of Line |
| **:trl L** | Next Find |
| **:trl U** | Abort |
| **\lt D** | Delete Line |
| **\lt L** | Delete to EOL |
| **\lt A** | HELP |
| **;hft F4** | Edit Macro |
| **\lt N** | Replace File |
| **\lt M** | MACRO Start/Stop |
| **\lt E** | MACRO Execute |
| **:sc** | UNDO |

:IND/REPLACE PARAMETERS:

| | |
|---|---|
| **3** | Search Backwards |
| **G** | Global Search |
| **N** | Find Nth Occurrence |
| **J** | Ignore Case |
| ***N*** | Whole Words Only |

A program on the utility disk called NTEINST.EXE ıas been provided to allow some customization of :he Tennelec/Nucleus, Inc. Text Editor (NTE.EXE). Γo use this program, copy NTEINST.EXE onto the ;ame disk or subdirectory that you keep the \TE.EXE program on and then type NTEINST then <RETURN>. The NTEINST main screen will then be /isible on your CRT. The top box on the CRT will ;how the operation of the NTEINST menus (similar :o MCS-II menu operation). The bottom box will ;how the operations that the program can ɔerform.

### Data File Directory Path

Γhis specifies the name of the default data file that \TE.EXE will use if no data file name is specified on the DOS command line. The default data file name is DEMO.NBS.

### Write Backup Files

This tells NTE.EXE whether to write backup files (.BAK) to disk during file save operations. The default is no backup files.

### Fast Ega Screen Writes

This tells NTE.EXE whether to use a fast method to write text to the CRT for EGA, VGA or monochrome TTL type displays or: use a slow method of screen writing to prevent video 'SNOW' on CGA type displays. The default is the fast EGA write method.

### Default Color Set

This restores the colors that NTE.EXE uses to the factory defaults.

### Customize Colors

This allows the colors that NTE.EXE uses to be set to any of the 128 colors that can be used by text mode programs like NTE.EXE.

### Restore Default Configuration

This restores the configuration of all installable parameters to the factory defaults.

### Install Program Commands

This redisplays the small help screen that shows how to operate the NTEINST.EXE program.

### Quit to DOS

This does an direct exit to DOS. Any changes made using other features of the NTEINST.EXE program will NOT be saved for later use.

### Save New Configuration

This saves the new configuration into the NTE.EXE program on disk.

## 5.0 SPECIAL SOFTWARE

### 5.1 MCS FILE FORMATS

The following 'C' language source code is provided to illustrate the exact format of the MCS spectrum file header and the MCS configuration file.

```
#define DTLEN 13
#define MAXID 72

/* --- MCS DATA TYPES ------- */

/* --- Standard Data Types -- */

typedef unsigned char     BOOL;
typedef int                   SHORT;
typedef unsigned int      USHORT;
typedef long int          LONG;
typedef unsigned long int ULONG;
typedef char              CHAR;
typedef unsigned char     UCHAR;
typedef float             FLOAT;
typedef double            DOUBLE;

/* --- Standard Data Types -- */

/* --- CALIB TYPEDEF -------- */

typedef struct
  {
   CHAR units[4];          /* units strings (ie:Kev,feet,gallons ..)      */
   DOUBLE calx0;           /* coefficients for: 0th order term            */
   DOUBLE calx1;           /* coefficients for: first order term          */
   DOUBLE calx2;           /* coefficients for: second order term         */
   SHORT  numbpoints;      /* number of calibration points                */
   DOUBLE calctrd[5];      /* centroids for up to 5 points                */
   DOUBLE calvalue[5];     /* values for up to 5 calibration points       */
   } CALIB;

/* --- CALIB TYPEDEF --------                                            */


/* --- HEADER STRUCT --------                                            */

typedef struct
  {
   UCHAR  etime[3];            /* (0 - 2)                                */
   UCHAR  ltimeflag;           /* (3  ) UNUSED                           */
   UCHAR  rtimeflag;           /* (4  ) UNUSED                           */
   UCHAR  convergain;      /* (5  ) UNUSED                               */
   UCHAR  digoffset;           /* (6  ) UNUSED                           */
   UCHAR  idcodestr[15];       /* (7 - 21)  file name root               */
   UCHAR  pca_date[12];        /* (22 - 33) mmm dd yyyy                  */
   UCHAR  pca_time[12];        /* (34 - 45) hh:mm:ss PM                  */
   UCHAR  group;               /* (46 )                                  */
```

```
 /* (47) system calibration flag - see control.units                    */
UCHAR  units;
 /* (48 - 157)  110 byte calibration structure                          */
CALIB  calib;
UCHAR  expansion[99];       /* (158 - 256) UNUSED                      */
UCHAR  phamode;             /* (257) UNUSED                            */
UCHAR  mcs;                 /* (258) UNUSED                            */
UCHAR  mcstimelab;          /* (259) UNUSED                            */
 /* (260) (not used by MCS product)                                     */
UCHAR  mcsdwellnumb;
UCHAR  mcspasct[3];         /* (261 - 263) UNUSED                      */
UCHAR  centupflag;          /* (264)                                   */
UCHAR  fwhmupflag;          /* (265)                                   */
SHORT  numbchanspm;         /* (266 - 267)                             */
 /* (268) [10c] acquisition start day (byte)                            */
UCHAR  asday;
 /* (269) [10d] acquisition start month (byte)                          */
UCHAR  asmonth;
 /* (270) [10e] acquisition start year (word)                           */
SHORT  asyear;
 /* (272) [110] acquisition start hundredth  (byte)                     */
UCHAR  ashund;
 /* (273) [111] acquisition start sec (byte)                            */
UCHAR  assec;
 /* (274) [112] acquisition start minute (byte)                         */
UCHAR  asmin;
 /* (275) [113] acquisition start hour (byte)                           */
UCHAR  ashour;
 /* (276) [114] acquisition stop day (byte)                             */
UCHAR  astpday;
 /* (277) [115] acquisition stop month (byte)                           */
UCHAR  astpmonth;
 /* (278) [116] acquisition stop year (word)                            */
SHORT  astpyear;
 /* (280) [118] acquisition stop hundredth (byte)                       */
UCHAR  astphund;
 /* (281) [119] acquisition stop sec (byte)                             */
UCHAR  astpsec;
 /* (282) [11a] acquisition stop minute (byte)                          */
UCHAR  astpmin;
 /* (283) [11b] acquisition stop hour (byte)                            */
UCHAR  astphour;
 /* (284-355) ID string (72 byte null terminated ascii                  */
 /* string)                                                             */
UCHAR  id[MAXID];
 /* (356) program major version number (byte)                           */
UCHAR  majvers;
 /* (357) program minor version number (byte)                           */
UCHAR  minvers;
 /* (358 - 361) (4 byte long int)                                       */
ULONG  real_elap_time;
 /*(362-374)acquire stop time hh:mm:ss PM(12 byte ascii                */
 /* string)                                                             */
UCHAR  acqstop_time[DTLEN];
 /*(375-387)acquire stop date mmm dd yyyy(12 byte ascii                */
 /* string)                                                             */
```

```
  UCHAR  acqstop_date[DTLEN];
   /*(388-400)acquire start time hh:mm:ss PM(12 byte ascii                    */
   /* string)                                                                 */
  UCHAR  acqstart_time[DTLEN];
   /*(400-413)acquire start date mmm dd yyyy(12 byte ascii                    */
   /* string)                                                                 */
  UCHAR  acqstart_date[DTLEN];
   /* (414 - 417) MCS id string: = { 'M' 'C' 'S' '\0' }                       */
  UCHAR  mcs_id[4];
   /* (418 - 421 preset time in seconds                                       */
  ULONG  preset_time;
  UCHAR  dac;              /* (422) MSB, TRI or SAW                          */
   /* (423) 'e' 'u' 'm' 'S' 'M' 'H' or 'D'                                    */
  UCHAR  dwell_units;
   /* (424-425) MCS dwell multiplier (1..65534)                               */
  USHORT dwell;
   /* (426 - 509) for future expansion                                        */
  UCHAR  future[84];
   /* (510-  511) header end marker = 100 * version number                    */
   /* (word)                                                                  */
  SHORT  endheader;
 } HEADER;
/* --- HEADER STRUCT -------                                                  */

/* --- CONFIG STRUCT -------                                                  */
typedef struct
 {
   /* Log == (0), 256 == (1), 512 == (1), ... 16M == (17)                    */
  SHORT   vscale;
   /* an index into the expand level: 0->8192, 1->4096,..                     */
   /* 6->256                                                                  */
  SHORT   expand;
  SHORT   group;                  /* group number: 1 -> 63                   */
                                  /* (ie:control.group)                      */
  UCHAR   pattern[65];            /* DIR (file_box) search pattern           */
  UCHAR   filename[65];           /* active spectrum file                    */
  UCHAR   loadname[65];           /* file name to load                       */
  UCHAR   report_name1[80]; /* ascii out file name                           */
  UCHAR   report_name2[80]; /* ROI out file name                             */
  UCHAR   report_name3[80]; /* integrals out file name                       */
  UCHAR   peak_rpt_name[80];    /*file name for peak report                  */
  UCHAR   pcxname[80];            /* file name for PCX file name             */
  UCHAR   batchname[80];          /* batch file name                         */
  UCHAR   username[80];           /* user program name                       */
  SHORT   dwell;                  /* dwell time for MCS mode                 */
  UCHAR   dwell_units;            /* 'u' 'm' 's' 'M' 'H' 'D' or 'e'          */
  UCHAR   dac;                    /* MSB, TRI or SAW                         */
  USHORT  initport;               /* I/O address of the MCS card             */
 } CONFIG;
/* --- CONFIG STRUCT -------                                                  */
```

## .2 THE USER FEATURE

ser feature programs use software interrupts to call subroutines inside MCS-II.

rograms that use the user feature should only be run with MCS-II in memory. The following 'C' language rogram reads how many channels are on the MCS card.

```
include "stdio.h"          /* Header file for Turbo or Microsoft C                    */
include "stdlib.h"             /* Header file for Turbo or Microsoft C                */
include "dos.h"            /* Header file for Turbo or Microsoft C                    */
include "string.h"         /* Header file for Turbo or Microsoft C                    */

define MCSUSERINT  0xf3    /* interrupt vector to use                                 */
har   MCS_prompt[] = "[MCS-II is Memory Resident]$p$g";

* ------------------------------------------------------                               */
oid main(void)
{
 union REGS regs;          /* interrupt union                                         */
 char far *cptr;               /* pointer to byte                                     */

* +++++++++++++++ Test for MCS-II in RAM +++++++++++++++++                             */

  cptr = getenv("PROMPT");
  if ( strcmp(cptr, MCS_prompt) != 0)
* Try to see if MCS-II is memory resident                                             */
    {
      fputs("ERROR: MCS-II does not appear to be in
       memory.\n",stdout);  exit(1);
    }

* Get Total numbers of channels on the MCS Card function                             */
  regs.x.ax = 129;

* Next: do the software interrupt (call code in MCS-II)                              */
  int86(MCSUSERINT, &regs, &regs);

* Next: output the results to the screen                                             */
  printf("Number of Channels on the MCS Card:
    %d\n",regs.x.ax);
}
```

he Calling Parameters for MCS User Programs Follow:

unction: 128 — Get MCS-II DGROUP
Input: AX = 128
Output: AX = MCS-II DGROUP

unction: 129 — Get Total numbers of channels on the MCS Card
Input: AX = 129
Output: AX = numbers of channels on the MCS Card

unction: 183 — Get id string
Input: AX = 183
Output: ES:SI is a pointer to the id string.

Function: 256 Get time structure
Input: AX = 256
Output: ES:SI is a pointer to the time structure

Function: 257 Get color structure
Input: AX = 257
Output: ES:SI is a pointer to the color structure

Function: 258 Set color structure
Input: AX = 258
Output: ES:SI is a pointer to the color structure

Function: 259 Get address of the count array
Input: AX = 259
Output: ES:SI is a pointer to the count array

Function: 260 Get address of the plot array
Input: AX = 260
Output: ES:SI is a pointer to the plot array

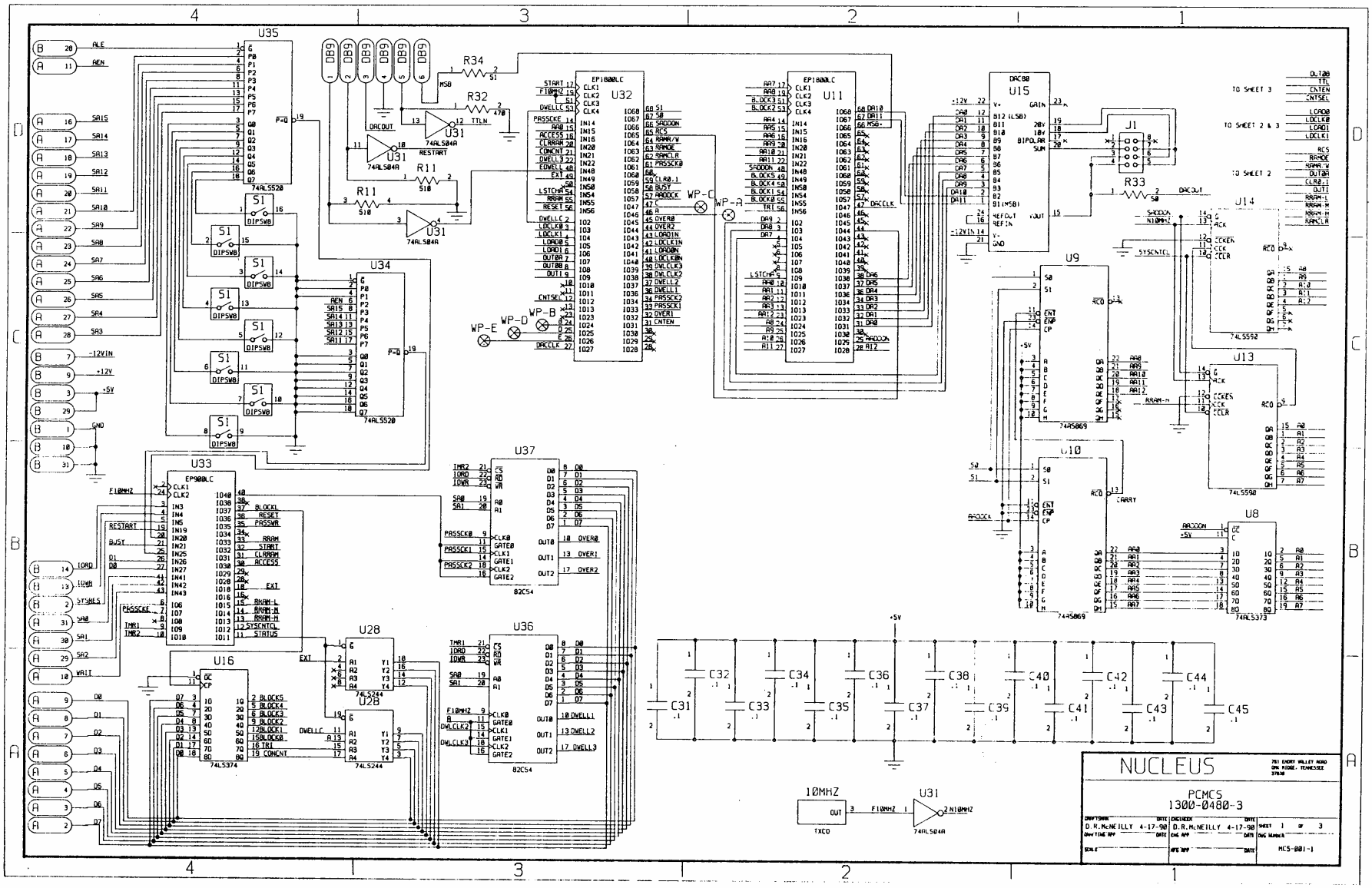
Function: 0FFFEh Get MCS-II version string
Input: AX = 0FFFEh
Output: ES:SI is a pointer to the MCS-II Version string
AX = number of bytes in the version string

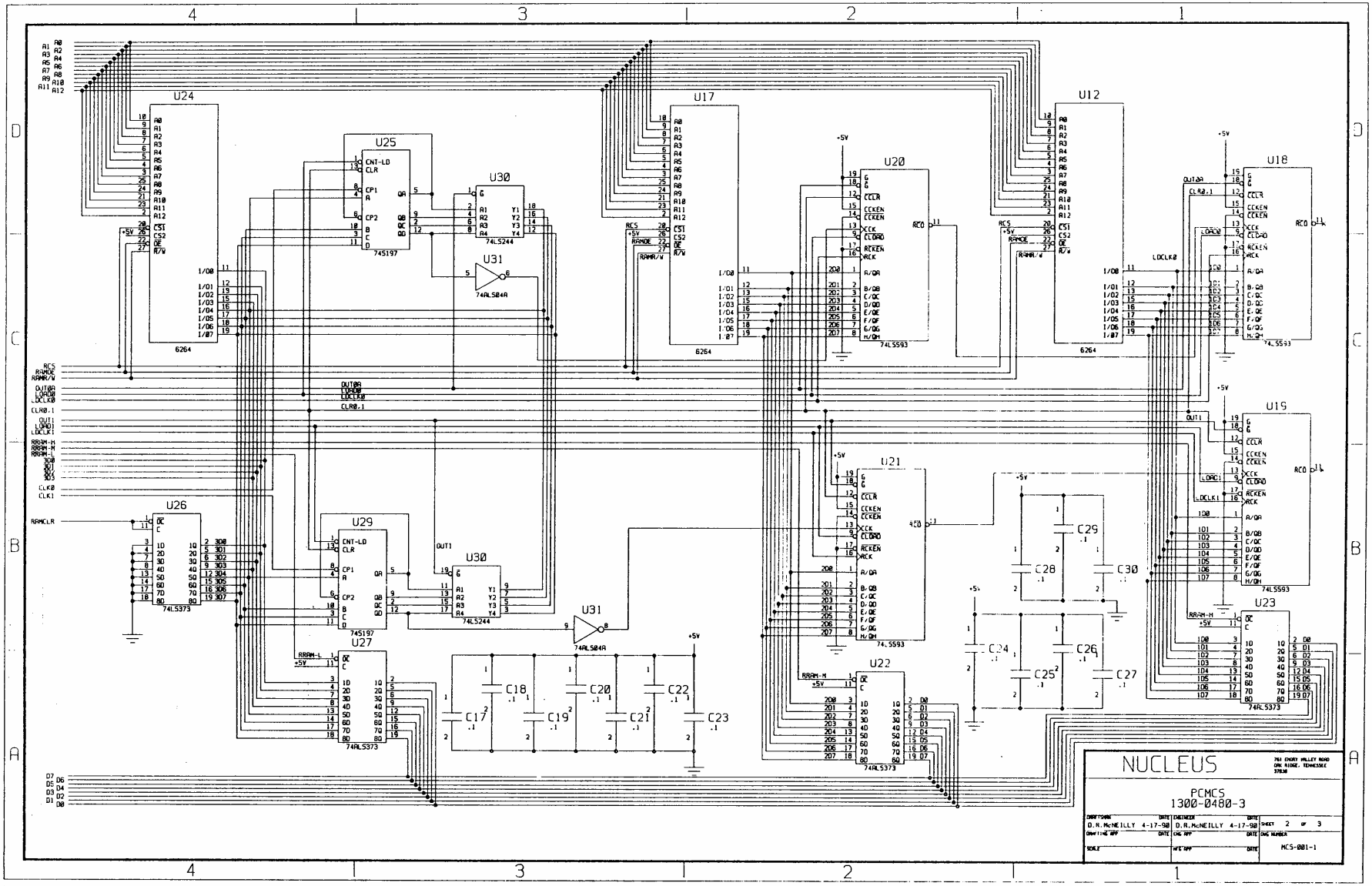
Function: 261 Get acquire start date string
Input: AX = 261
Output: ES:SI is a pointer to the acquire start date string.

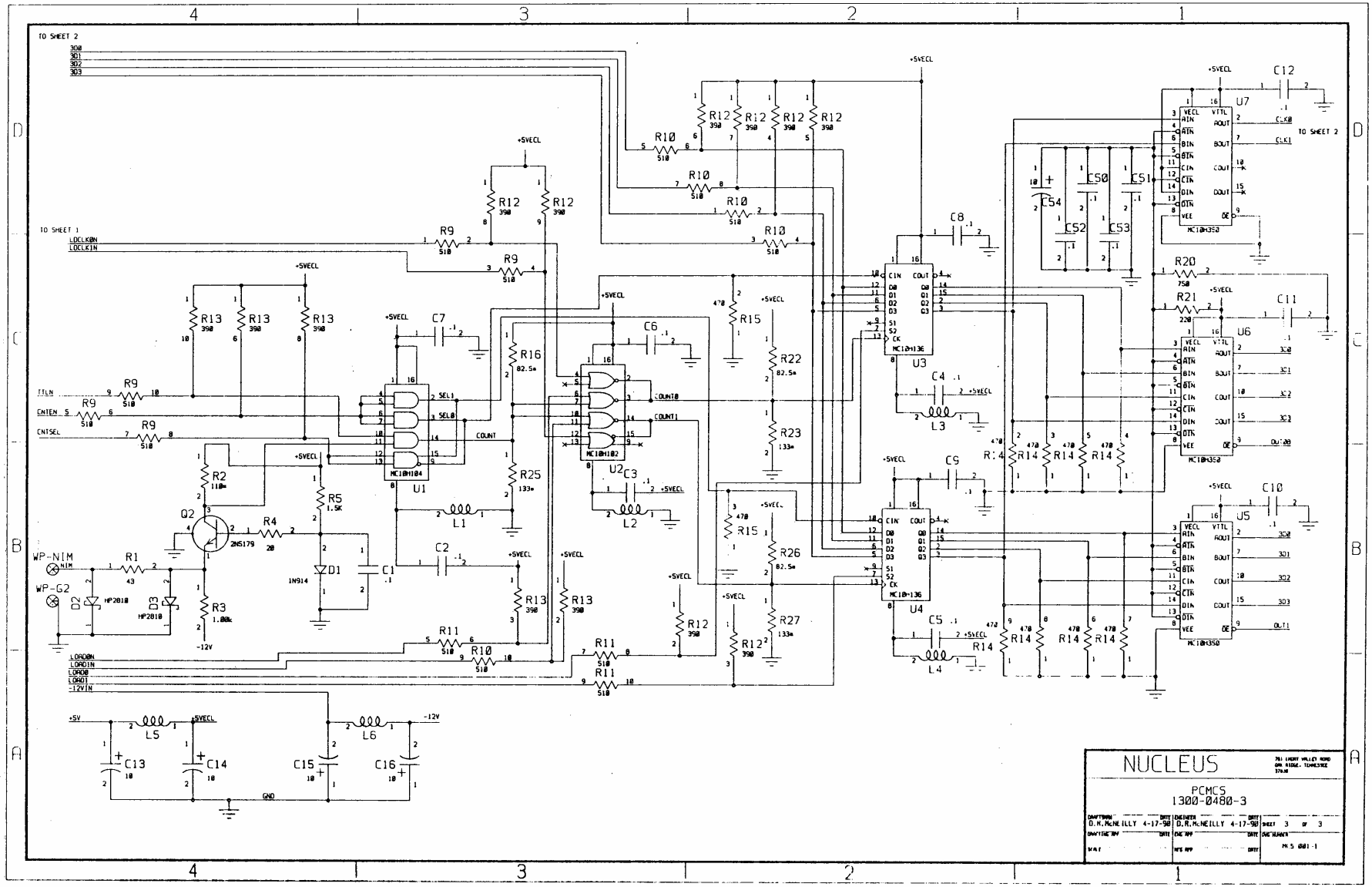
Function: 262 Get acquire start time string
Input: AX = 262
Output: ES:SI is a pointer to the acquire start time string.

Function: 263 Get acquire stop date string
Input: AX = 263
Output: ES:SI is a pointer to the acquire stop date string.

Function: 264 Get acquire stop time string
Input: AX = 264
Output: ES:SI is a pointer to the acquire stop time string.


U35
74ALS520
U34
74ALS520
U33
EP900LC
U32
EP1800LC
U11
EP1800LC
U37
82C54
U36
82C54
U16
74LS374
U28
74LS244
U28
74LS244
U31
74ALS04A
U31
74ALS04A
U31
74ALS04A
R34
R32
R11
R11
DACOUT
RESTART
WP-A
WP-B
WP-C
WP-D
WP-E
U15
DAC80
U9
74AS869
U10
74AS869
U13
74LS592
U14
74LS592
U8
74ALS373
R33
J1
C31
C32
C33
C34
C35
C36
C37
C38
C39
C40
C41
C42
C43
C44
C45
10MHZ
TXCO
NUCLEUS
PCMCS
1300-0480-3
D.R.McNEILLY 4-17-90
MCS-001-1


NUCLEUS
PCMCS
1300-0480-3
D.R.McNEILLY 4-17-98
D.R.McNEILLY 4-17-98
SHEET 2 OF 3
MCS-001-1


TO SHEET 2
TO SHEET 1
LOCLKØN
LOCLKIN
TTLN
CNTEN_S
CNTSEL
WP-NIM
WP-G2
LOADN
LOADIN
LOAD0
LOAD1
-12VIN
+5V
GND
+5VECL
-12V
U1
U2
U3
U4
U5
U6
U7
MC10H104
MC10H102
MC10H136
MC10H350
2N5179
1N914
HP2810
COUNT
COUNT0
COUNT1
SEL1
SEL0
NUCLEUS
PCMCS
1300-0480-3
D.R.McNEILLY 4-17-90
SHEET 3 OF 3