PC208W DATALOGGER SUPPORT SOFTWARE
INSTRUCTION MANUAL

REVISION: 4/99

COPYRIGHT (c) 1996-1999 CAMPBELL SCIENTIFIC, INC.

LIMITED WARRANTY

CAMPBELL SCIENTIFIC, INC. warrants that the magnetic diskette on which the accompanying computer software is recorded and the documentation provided with it are free from physical defects in materials and workmanship under normal use. CAMPBELL SCIENTIFIC, INC. warrants that the computer software itself will perform substantially in accordance with the specifications set forth in the Operator's Manual published by CAMPBELL SCIENTIFIC, INC. CAMPBELL SCIENTIFIC, INC. warrants that the software is compatible with IBM PC/XT/AT and PS/2 microcomputers and 100% compatible computers running Windows 3.1, Windows 95, or Windows NT only. CAMPBELL SCIENTIFIC, INC. is not responsible for incompatibility of this software running under any operating system other than those specified in accompanying data sheets or Operator's Manuals.

The above warranties are made for ninety (90) days from the date of original shipment.

CAMPBELL SCIENTIFIC, INC. will replace any magnetic diskette or documentation which proves defective in materials or workmanship without charge.

CAMPBELL SCIENTIFIC, INC. will either replace or correct any software that does not perform substantially according to the specifications set forth in the Operator's Manual with a corrected copy of the software or corrective code. In the case of significant error in the documentation, CAMPBELL SCIENTIFIC, INC. will correct errors in the documentation without charge by providing addenda or substitute pages.

If CAMPBELL SCIENTIFIC, INC. is unable to replace defective documentation or a defective diskette, or if CAMPBELL SCIENTIFIC, INC. is unable to provide corrected software or corrected documentation within a reasonable time, CAMPBELL SCIENTIFIC, INC. will either replace the software with a functionally similar program or refund the purchase price paid for the software.

CAMPBELL SCIENTIFIC, INC. does not warrant that the software will meet licensee's requirements or that the software or documentation are error free or that the operation of the software will be uninterrupted. The warranty does not cover any diskette or documentation which has been damaged or abused. The software warranty does not cover any software which has been altered or changed in any way by anyone other than CAMPBELL SCIENTIFIC, INC. CAMPBELL SCIENTIFIC, INC. is not responsible for problems caused by computer hardware, computer operating systems or the use of CAMPBELL SCIENTIFIC, INC.'s software with non-CAMPBELL SCIENTIFIC, INC. software.

ALL WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED AND EXCLUDED. CAMPBELL SCIENTIFIC, INC. SHALL NOT IN ANY CASE BE LIABLE FOR SPECIAL, INCIDENTAL, CONSEQUENTIAL, INDIRECT, OR OTHER SIMILAR DAMAGES EVEN IF CAMPBELL SCIENTIFIC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CAMPBELL SCIENTIFIC, INC. is not responsible for any costs incurred as result of lost profits or revenue, loss of use of the software, loss of data, cost of re-creating lost data, the cost of any substitute program, claims by any party other than licensee, or for other similar costs.

LICENSEE'S SOLE AND EXCLUSIVE REMEDY IS SET FORTH IN THIS LIMITED WARRANTY. CAMPBELL SCIENTIFIC, INC.'S AGGREGATE LIABILITY ARISING FROM OR RELATING TO THIS AGREEMENT OR THE SOFTWARE OR DOCUMENTATION (REGARDLESS OF THE FORM OF ACTION; E.G., CONTRACT, TORT, COMPUTER MALPRACTICE, FRAUD AND/OR OTHERWISE) IS LIMITED TO THE PURCHASE PRICE PAID BY THE LICENSEE.

LICENSE FOR USE

This software is protected by both the United States copyright law and international copyright treaty provisions. You may copy it onto a computer to be used and you may make archival copies of the software for the sole purpose of backing-up CAMPBELL SCIENTIFIC, INC. software and protecting your investment from loss. All copyright notices and labeling must be left intact.

This software may be used by any number of people, and may be freely moved from one computer location to another, so long as there is not a possibility of it being used at one location while it's being used at another. The software, under the terms of this license, cannot be used by two different people in two different places at the same time.

CAMPBELL SCIENTIFIC, INC.

815 W. 1800 N.
Logan, UT 84321-1784
USA
Phone (435) 753-2342
FAX (435) 750-9540
www.campbellsci.com

Campbell Scientific Canada Corp.
11564 -149th Street
Edmonton, Alberta T5M 1W7
CANADA
Phone (403) 454-2505
FAX (403) 454-2655

Campbell Scientific Ltd.
Campbell Park
80 Hathern Road
Shepshed, Leics. LE12 9RP
ENGLAND
Phone (44)-50960-1141
FAX (44)-50960-1091

WHAT'S NEW!

If you are upgrading from PC208W 2.3 or earlier, you'll notice a few changes in PC208W 3.0.

SETUP

Improved Modem Communication - Setting up remote communications is easier with PC208W's new modem configurations. PC208W uses hardware flow control, verbose result codes (looking for “CONNECT” back from the modem rather than a number), and error correction which is consistent with the newer, faster modems. Just pick your modem from a list, and you should be ready to communicate. If your modem is not in the list, the default modem will work in most instances.

> **NOTE:** If you used a custom modem string in a previous version of PC208W, please refer to the special upgrade information provided in Section 1.4, Installation.

User-determined FSA 2 Data File Names - PC208W now lets you specify different names for data files from Final Storage Areas 1 and 2.

New Clock Offset - A clock offset feature has been added to make it easier to set the datalogger time for different time zones or daylight savings.

Updated Scheduling Capability - Datalogger schedules can now be applied to one or more datalogger stations. When you have one station's schedule set up, simply click a button to apply those settings to other stations. This helps you set up your data collection schedules in less time and with less effort.

Datalogger Call-back - If you use call-back with your dataloggers, you'll notice that PC208W automatically tries to initialize the COM port. Check the Help on “Call-back” for details on properly setting up the network.

CONNECT

New Numeric Monitor - The Numeric Monitor is now a stand-alone window that is opened separately from the other Connect screen views. This 17-column by 15-cell display has room for assigning 254 input locations. It can be sized to view one to four columns (up to 60 input locations) at one time. This new design allows you to monitor your real-time data with a small screen while minimizing the rest of PC208W so you can work on other tasks.

Enhanced Graphing Capability - You can now save any of the three real-time graphs to disk in a WMF (Windows metafile) format for printing or importing into other programs.

VIEW

New Graphing Capability - View now plots up to two array elements on one graph. This is especially useful to quickly identify potential problems in your data (or sensors) either in the field or office. Use the zoom feature for a closer look at the graph, or click on a data point to display its numeric value.

Enhanced Ease of Use - Graphical icons have been added to the user interface for easier and more efficient use of the program. For instance, you can view your data in comma separated, columnar, or hexadecimal format just by clicking a button.

Enhanced Printing Capability - You can now preview your data or graph before printing. Headings have been added to the printed page for quicker identification of printed data. Margins can be adjusted by entering numerical values or by simply "dragging" them in Print Preview mode.

NEW START-UP FEATURE!

Command Line Capabilities - Command line arguments can be entered from a Windows command prompt or Windows short cut to open PC208W and perform a specific task automatically. Collecting data, setting the clock, invoking the Numeric Monitor or graphs, and toggling ports or flags can all be accomplished using command line arguments. Administrators can set up complex tasks for associates to run without training them on the entire package.

You can find a comprehensive list of these command line entries in the Help (search for

“command line”) and in Section 1.5 of this manual.

REPORT

Start Offsets - A negative value can now be entered as an offset on the Input File tab. This directs Split to begin reading X bytes from the end of the file (positive values begin reading from the beginning of the file). This makes using the time sync functions to process the tail end of a data file much faster when that data file is very large.

SplitR - Multiple copies of SplitR, the command-line version of Split, can be run at once using the /M switch.

PC208W DATALOGGER SUPPORT SOFTWARE
TABLE OF CONTENTS

PAGE

PC208W 3.0 INTRODUCTION

I.1 PLEASE READ THIS 1
I.2 OVERVIEW 1
I.3 OTHER SOFTWARE OF INTEREST 2
I.4 INSTALLATION 2
I.5 UNINSTALL 4
I.6 SOME NOTES ON SYSTEM RESOURCES 4
I.7 GETTING HELP 5

1. SETTING UP THE NETWORK AND COMMUNICATING WITH THE DATALOGGER

1.1 GETTING STARTED 1-1
1.2 SETUP 1-1
1.2.1 General Information 1-1
1.2.2 Setup Procedures 1-1
1.2.3 Device Map Notes 1-2
1.2.4 Example Device Maps 1-3
1.2.5 Notes on Data Collection and Formats 1-5
1.2.6 Setup Safety Tips 1-6
1.3 CONNECT 1-6
1.3.1 Handy Things That May Not Be Obvious 1-7
1.3.2 Connect Safety Tips 1-8
1.4 STATUS 1-8
1.4.1 Device Map and Station Map 1-8
1.4.2 Low Level I/O 1-9
1.4.3 COM SWF Messages 1-9
1.4.4 Pause 1-9
1.5 COMMAND LINE PARAMETERS 1-9

2. PROGRAM: EDLOG

2.1 INTRODUCTION 2-1
2.2 GETTING STARTED 2-3
2.2.1 Creating a New EDLOG Program 2-3
2.2.2 Editing an Existing Program 2-3
2.2.3 Documenting a DLD File 2-3
2.2.4 The Program Development Screen 2-3
2.3 COMMENTS 2-4
2.4 INSTRUCTIONS 2-4
2.4.1 Entering Parameters 2-4
2.4.2 Data Entry Warning 2-5

PAGE

2.5 EXPRESSIONS ... 2-5
2.5.1 Rules for Creating Expressions ... 2-5
2.5.2 Examples ... 2-6

2.6 EDITING EDLOG PROGRAMS ... 2-6
2.6.1 Editing Comments, Instructions, and Expressions ... 2-7
2.6.2 Cut, Copy, Paste, and Clipboard Options ... 2-7
2.6.3 Library Files ... 2-7
2.6.4 Renumbering the Instructions ... 2-7
2.6.5 Compress View ... 2-7

2.7 INPUT LOCATIONS ... 2-7
2.7.1 Entering Input Locations ... 2-7
2.7.2 Reps ... 2-8
2.7.3 Input Location Editor ... 2-8
2.7.4 Input Location Anomalies ... 2-9

2.8 INDENTION ... 2-10

2.9 FILE TYPES ... 2-10
2.9.1 Program Documentation, *.CSI ... 2-10
2.9.2 Download Files, *.DLD ... 2-10
2.9.3 Final Storage Label Files, *.FSL ... 2-10
2.9.4 Program Trace Information Files, *PTI ... 2-10
2.9.5 Printer Output Files, *TXT ... 2-10
2.9.6 Library Files, *.LBR ... 2-10

3. REPORT: SPLIT

3.1 OVERVIEW ... 3-1

3.2 GETTING STARTED ... 3-1

3.3 SPLIT PARAMETER FILE ENTRIES ... 3-5
3.3.1 Input File(s) ... 3-5
3.3.2 Start Condition ... 3-7
3.3.3 Stop Condition ... 3-9
3.3.4 Copy ... 3-12
3.3.5 Select ... 3-12
3.3.6 Output Files ... 3-24

3.4 HELP OPTION ... 3-26

3.5 EDITING COMMANDS ... 3-26

3.6 COMMAND LINE ENTRIES ... 3-26

4. VIEW

4.1 INTRODUCTION ... 4-1

4.2 MENU ITEMS ... 4-1

4.3 ICON ITEMS ... 4-1

4.4 RUNNING VIEW AS A STAND-ALONE PROGRAM ... 4-3

4.5 ASSIGNING DATA FILES TO VIEW ... 4-3

PAGE

5. STG MODULE: SMS - STORAGE MODULE SOFTWARE

5.1 INTRODUCTION 5-1

5.2 GETTING STARTED 5-1
5.2.1 The Setup Screen 5-2
5.2.2 Establishing Communications 5-3

5.3 STATUS INFORMATION 5-5
5.3.1 Status Box/Update Status 5-5
5.3.2 Advanced Status Information 5-5

5.4 PROGRAMS 5-7
5.4.1 Program Location 5-8
5.4.2 Clear 5-8
5.4.3 Store 5-8
5.4.4 Read 5-8

5.5 DATA 5-8
5.5.1 File Format 5-9
5.5.2 File Naming Options 5-9
5.5.3 Show Card/Module Directory 5-10
5.5.4 Get All 5-10
5.5.5 Get New 5-10
5.5.6 Get One 5-10

5.6 ERASE 5-10
5.6.1 Erase Data 5-10
5.6.2 Erase Data and Programs 5-11
5.6.3 Erase and Test Card/Module 5-11

5.7 THE MENU BAR 5-11
5.7.1 File 5-11
5.7.2 Options 5-11
5.7.3 Data 5-11
5.7.4 Help 5-12

5.8 ABORT 5-12

6. PROGRAM CONVERSION

6.1 DOC2CSI 6-1
6.1.1 Copying Program Files 6-1
6.1.2 Running DOC2CSI 6-1
6.1.3 Example Usage 6-1
6.1.4 Notes 6-2

6.2 CR10 PROGRAMS TO CR10X 6-2

6.3 XLATE 6-2
6.3.1 File Menu 6-2
6.3.2 Options Menu 6-2
6.3.3 Using Translated Files 6-3

PAGE

APPENDICES

A. FREQUENTLY ASKED QUESTIONS

A.1 SETUP, CONNECT, AND STATUS A-1

A.2 EDLOG.EXE A-2

B. MODEMS

B.1 PHONE MODEMS B-1

B.2 GENERIC MODEM DIALING STRINGS B-1

B.3 MD9 MULTIDROP INTERFACE B-2

B.4 DATALOGGER INITIATED COMMUNICATIONS (CALL-BACK) B-2

C. PC208W ERROR MESSAGES

C.1 SPLIT ERROR MESSAGES C-1

C.2 EDLOG EXPRESSION ERRORS C-2

C.2.1 Causes C-2

D. CARD SERVICES DRIVER SOFTWARE NOTES

D.1 GENERAL D-1

D.2 DRIVER VERSION NUMBERS D-1

D.2.1 Microsoft Windows D-1

D.2.2 CSL-CS.DLL D-2

D.3 SETTING UP CARD SERVICES D-2

D.3.1 How to Choose Which Area of Memory to Allocate to Card Services D-3

E. GLOSSARY OF COMMON WINDOWS TERMINOLOGY E-1

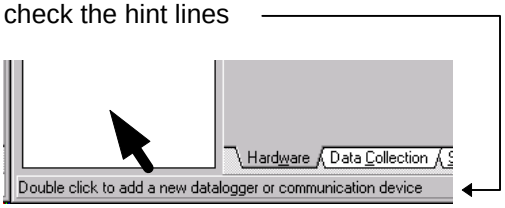
GLOSSARY

PC208W 3.0 INTRODUCTION

I.1 PLEASE READ THIS

Welcome to PC208W 3.0, Campbell Scientific's Windows® compatible support software for the CR10X, CR10, CR500, 21X, CR7, and CR23X dataloggers. ***Please*** take time to read the Introduction and Frequently Asked Questions (Appendix A) sections of this manual—we've tried hard to keep them short and direct. We have designed the PC208W windows and help screens to make learning the software as intuitive as possible, and have intentionally kept this manual short.

We hope that the use of most controls and features will be obvious, but for those that are not, there are several ways to get help. First check the hint lines



which are displayed on the bottom of most screens. These provide a one line summary of the control under the mouse cursor. For more help, select a control with the mouse pointer or <Tab> key and press the <F1> key. In some instances clicking the right mouse button will bring up options that will help.

This manual assumes that the user is familiar with the Microsoft Windows interface. If you need help with some of Windows' features (double-clicking, cutting and pasting, etc.) refer to Appendix E.

I.2 OVERVIEW

There are eight buttons on the PC208W toolbar. Here is what they are used for:

- **Setup**: Configure the devices (including modems, dataloggers, COM ports, and tasks) that you use, with separate tabs for hardware, data collection, and collection schedules.
- **Connect**: Go on-line with a datalogger to set the clock, send programs, collect data, view and graph measurements, etc.
- **Status**: Check the communication and data collection status of all devices you work with, and trigger manual data collection.
- **Program**: Create and edit datalogger programs with EDLOG.
- **Report**: Process the data files and create reports using SPLIT.
- **View**: View ASCII files in comma separated, columnar, or hexadecimal format. Quickly graph up to two elements in your data file.
- **Stg Module**: Retrieve files from (and reset, program, etc.) Storage Modules either directly or via a CR10(X), CR500, or CR23X, and from PCMCIA (PC) Cards via the CSM1/MCR1 interface or directly from a PCMCIA slot in the PC.
- **Help**: Enter the PC208W help system.

Each of these buttons launches its own window which gives you access to all of PC208W's functions all the time. The details of each screen are described further in the rest of this manual and in the Help.

> **NOTE:** Reading Card Storage Module cards directly in a PCMCIA slot requires compliant drivers. Some computers not running the Windows 95 operating system may require the purchase of drivers from a software vendor.

I.3 OTHER SOFTWARE OF INTEREST

There are other programs available from Campbell Scientific that you may find useful. These packages are free and are available from our Internet site (www.campbellsci.com) or on disk (call 435-753-2342 and ask for an Applications Engineer):

- SHORT CUT: An alternative to EDLOG for creating simple programs. SHORT CUT guides the user through the selection and setup of sensors (including wiring) and data storage for many meteorological, hydrological, and industrial applications. SHORT CUT supports the CR10(X) , CR500, and 21X dataloggers. (You can then download the program to the datalogger and retrieve data with PC208W.)

- PC TOUR: A computer-based tutorial which introduces a new user to basic datalogger and datalogger support software use. It provides an overview of a CR10X datalogger wiring panel, discusses the concepts behind a data acquisition system, and presents an overview of PC208W.

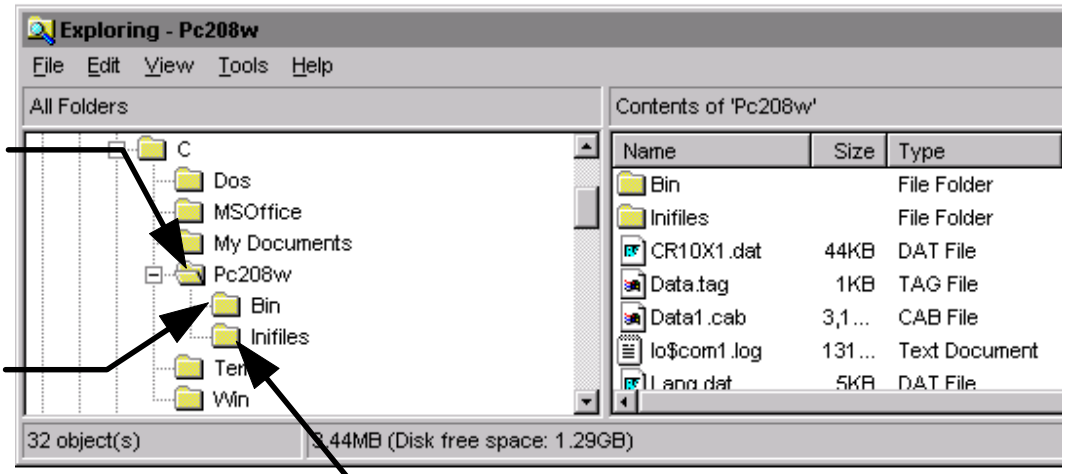
I.4 INSTALLATION

CAUTION: As with all software, Campbell Scientific strongly recommends that a system back-up be performed before software installation. In addition, periodic back-ups should be performed on files in the PC208W directory. These files include: PC208W.dnd (datalogger network map), Wmodem3.ini (modem setup information), Edlog program files (*.DLD, *.CSI), Split Parameter files (*.PAR), and collected data files (*.DAT).

PC208W requires Windows 3.1®, Windows 95®, Windows 98®, or Windows NT®. We recommend at least a 486 computer with 16 M RAM. Though PC208W may run on slower systems, better performance will be seen with faster systems. A mouse is also required. A minimum of 10 M of hard disk space is required for software installation. Additional disk space may be required for I/O log files if you choose to log these files to disk (refer to Section 1.4.2 and Appendix A for more information on log files).

Working Directory:
- Programs
- Data files
- etc.

PC208W Software and templates:
- pc208w.exe
- edlogw.exe
- etc.



"ini" files for the main PC208W screens and connect screen setting for each datalogger.

Insert the first PC208W disk in drive A: or B:. In Windows 95 or NT, select RUN from the **Start** button. In Windows 3.1, Select FILE and then RUN from the Program Manager Menu. Enter A:SETUP or B:SETUP and select the **OK** button. Follow the instructions given on the screen. PC208W lets you choose a working directory name and installs most of the software into a subdirectory named BIN.

In the example below, PC208W was installed to the C:\PC208W directory.

1.4.1 FOR USERS UPGRADING FROM PC208W VERSIONS 2.3 AND EARLIER

Modem Configurations

PC208W 3.0 uses a new configuration for modem setups, which should provide for more reliable communication and fewer problems in establishing remote communication links. If your station setup uses a modem that is included in PC208W's modem list, you won't need to make any changes after installing PC208W 3.0.

However, if you previously set up a custom modem instead of using an existing modem configuration, PC208W 3.0 will leave the modem pick list field blank. Choose the default modem; in most instances, this setup should work. If the default settings will not work with your modem, you will need to refer to your modem's user's manual to enter valid strings (refer to Appendix B). Some items to look for when developing a custom modem string are: hardware flow control enabled, verbose result codes (looking for “CONNECT” back from the modem rather than a number), and error correction enabled.

> **NOTE:** PC208W won’t find your custom modem on its WMODEM3.INI list so it will display a blank field for the Modem Pick List field and a warning in the SWF log until you actually pick "default modem".

Directory Structures

In version 2.1 and greater, the main PC208W directory is PC208W (unless you manually change the default directory during software installation). Two subdirectories exist under PC208W: BIN and INIFILES. The BIN directory is used to store nearly all of the files essential to the operation of the PC208W program (executable files, help files, etc.). The INIFILES directory is used to store all of the *.INI files associated with datalogger setups and PC208W software execution.

Datalogger *.INI files are created/updated each time a communication session with a datalogger is closed, and PC208W's INI file is updated each time it is closed. The files provide the PC208W program with specific information so your devices or program defaults do not have to be configured before each PC208W session.

Versions of software prior to 2.3 stored the PC208W .INI file in the working directory. This INI file included information on all of the dataloggers in the device map and setup information on PC208W. If you are upgrading from Version 2.2 or earlier, when the program is installed, your existing PC208W .INI file will be moved into the INIFILES directory. Then, the first time a connection is made to each datalogger that was set up in your original INI file, the program accesses the INI file and creates a new file from the information in the old INI. This change in directory structure was made to accommodate an INI file size limitation inherent to the Windows operating system.

Datalogger Network Configuration Files (*.DND files)

PC208W keeps the setup information in the working directory in a DND file. (In PC208W 1.0 and earlier, this file was named TCOM.DND. In versions later than 1.0, it was named PC208W.DND.) If an existing DND file is found during an upgrade installation, this file will be renamed, if necessary, and updated to the current DND format.

During the upgrade installation, you may choose to install the new PC208W in a different directory. If so, you can copy the old DND file into the new directory. If the old file is named TCOM.DND, rename it to PC208W.DND. By copying this DND file, you will not have to set up new network configurations (the device map) for your existing datalogger stations.

Existing Data, Edlog, and Split Files

Your existing PC208W files can be used without modification in PC208W 3.0 These file types include: *.DAT, *.PAR, *.CSI, *.FSL, *.DLD.

(Refer to Section 2.9 for more information on Edlog file types.)

For users upgrading from DOS PC208

You will have to recreate the setup information and batch routines. Edlog .CSI and .DLD files and Split .PAR files will work without modification. However, if your previous version of Edlog used .DOC files, they will need to be converted using DOC2CSI (see Section 6).

If you are connecting to one or more dataloggers via an MD9 network, and have been using PC208 for DOS, you will have to change the ID switch settings on your base MD9. The DOS software uses an address between 1 and 254; PC208W uses address 255.

If the base MD9 ID is set to 255, you can still use PC208 DOS software, if needed, without changing the setting back to a number between 1 and 254. Simply add a pound sign (#) to the end of the MD9 dialing path in the station file.

I.5 UNINSTALL

Should you need to remove PC208W from a computer, an uninstall program is provided. With Windows 95 or NT, select SETTINGS from the **Start** button then select CONTROL PANEL. Double click on ADD/REMOVE programs. Select PC208W from the list and select the **Add/Remove** button.

With Windows 3.1, double click on the UNINSTALL PC208W icon in the PC208W program group.

I.6 SOME NOTES ON SYSTEM RESOURCES

COM Ports and Interrupts

PC208W is capable of running in the background while you are doing other work with other programs. To do this, it uses the system's idle time for some of its serial communications. A few other applications are so resource intensive that PC208W does not receive sufficient time for its communications. PC208W will not run reliably with these applications loaded.

The Windows environment provides access to the COM ports via a COM port driver. Many computers have COM ports which share interrupts with other COM ports. (Typically COM1 and COM3 share IRQ 4 and COM2 and COM4 share IRQ 3.) This can be checked by examining the 'Ports' (Settings/Advanced) setting in the Windows Control Panel. A COM port sharing an interrupt with another device can prevent Windows from reliably performing communications using that port, particularly if the COM port and device are used concurrently. This shared interrupt problem would also affect operation of PC208W.

> **NOTE:** If you appear to be experiencing communication problems when using external modules, always check the COM ports setting in the Windows Control Panel, using the 'Ports' (Settings/Advanced) option, and check for any shared interrupts.

Other programs may also try to use the COM ports, and some may leave the COM ports "open" even when not active. This is especially true of DOS based communications programs. If PC208W has trouble connecting to a datalogger when you believe the setup and connections are correct, check to make sure other programs that use the same COM ports are closed.

For Windows 3.1 and 95 Users

A problem which is inherent to Windows 95 may surface when scheduling Tasks which call DOS programs. When a DOS command is executed, a certain amount of system resources are allocated to this command. Windows 95 does not free up this resource when the command has been executed and control is returned to Windows. This memory is freed, however, once your computer system is rebooted.

If your Windows 95/3.1 system remains on-line for long periods of time (such as a week or more without being powered down), it is suggested that the system be rebooted every few days if DOS commands are being executed. If not, your system may run out of system resources and fail to respond. The only solution to this system "lock-up" is to reboot the computer. Any files which have not been saved to disk will be lost.

I.7 GETTING HELP

We've tried hard to make PC208W 3.0 as intuitive as possible and follow common Windows conventions. There are several sources of help for you including:

- This manual—we've kept it short, please read it.
- Hint lines which appear at the bottom of each screen.
- Context-sensitive Help in each program available when you press <F1>.
- General help from the **Help** pull down menu.
- The datalogger manuals. These are thick and more for reference than to be read cover-to-cover. The Overview chapter, however, is a good one, and you should be sure to read it carefully.

From Anywhere:

Campbell Scientific, Inc.
815 West 1800 North
Logan, UT 84321-1784
Tel: (435) 753-2342
Fax: (435) 750-9540
E-mail: support@campbellsci.com
WWW: http://www.campbellsci.com

From Australia, Fiji, Indonesia, Malaysia, Papua New Guinea, Philippines, Singapore, and Thailand:

Campbell Scientific Australia Pty Ltd
PO Box 444 Thuringowa Central
Townsville QLD 4817 AUSTRALIA
Phone: 61-77-254-100
Fax: 61-77-254-155
E-mail: csa@ultra.net.au
WWW: http://www.ultra.net.au/~csa/

From Canada:

Campbell Scientific Canada
11564 - 149th Street
Edmonton, Alberta T5M 1W7
Phone: 403-454-2505
Fax: 403-454-2655
E-mail: dataloggers@campbellsci.ca

From Austria, Belgium, Denmark, Finland, France, Germany, Greece, Ireland, Italy, Luxembourg, Netherlands, Northern Ireland, Norway, Poland, Portugal, Spain, Sweden, Switzerland, and United Kingdom:

Campbell Scientific Ltd.
Campbell Park
80 Hathern Road
Shepshed, Leics. LE12 9RP
UNITED KINGDOM
Phone: +44-(0)1509-601141
Fax: +44-(0)1509-601091
E-mail: sales@campbellsci.co.uk
Technical Support:
support@campbellsci.co.uk
WWW: http://www.campbellsci.com/europe/

From South Africa:

Campbell Scientific Africa
Box 2450
Somerset West 7129
Republic of South Africa
Phone: 2721-8800885
Fax: 2721-8800240
E-mail: jvisagie@mbv.co.za

SECTION 1. SETTING UP THE NETWORK AND COMMUNICATING WITH THE DATALOGGER

1.1 GETTING STARTED

With the software installed, you are probably anxious to start using the datalogger. If at all possible, prior to deploying the equipment in the field, make sure you can establish a telecommunication link with your datalogger—even if you have to create a temporary setup in your office. Getting familiar with datalogger programming and the support software in the comfort of your office will be a lot less frustrating than trying to resolve problems in the field. If your datalogger is already deployed and working, then establishing communications with the datalogger is the place to start.

The first step in interfacing your PC and the PC208W software with your equipment is to set up your datalogger and other devices. **SETUP** is used to accomplish this. Once your equipment is configured, direct connection can be made to the logger through the **CONNECT** screens. After telecommunication has been established you can use the other features of PC208W to create and download programs, collect data, view raw data or prepare more sophisticated reports, monitor equipment status, and interface with storage devices.

1.2 SETUP

1.2.1 GENERAL INFORMATION

opens a window in which you edit the Device Map, a diagram of all the various dataloggers, modems and other communications devices which you are using. Three different pages on this screen show up as tabs to let you configure Hardware, Data Collection, and Scheduled Communications settings. Whether you work with one datalogger (or "station") on your desk or hundreds of dataloggers over elaborate telecommunications networks, PC208W's Device Map keeps track of the type of devices, addresses and phone numbers, data collection pointers, and collection schedules. You can also add "Tasks" to the Device Map to execute other programs, such as DOS batch files, SPLIT (PC208W's data processing utility), or any other executable program.

1.2.2 SETUP PROCEDURES

The first step in setting up telecommunications is to add the necessary component to the Device Map by choosing one of the **Add...** options from the toolbar. Some of the necessary entries include which serial communication port (e.g., COM1) you will use, modems (if any), type of datalogger, etc.

The Setup window consists of three tabs: **HARDWARE**, **DATA COLLECTION**, and **SCHEDULE**. Each device will have these three tabs but only parameters relevant to that type of device will be shown on the tabs.

PC208W inserts default values for most parameters, but some it can't know. Therefore, once you add devices to the Device Map, click on the **HARDWARE** tab for each device to enter settings unique for that device, such as baud rates for COM ports, brand of telephone modem (for initialization strings), telephone numbers, radio IDs or paths for dataloggers, etc.

You will also need to set up what data you want to collect and how you want to collect and store it. This information goes on the **DATA COLLECTION** tab. Note that PC208W will suggest a filename for the data, but you can Browse to choose a different name, directory, or even a different computer if you're on a network.

If you want PC208W to collect data automatically, you will need to specify data collection and retry information on the **SCHEDULE** tab. Automatic data collection can be toggled on/off by enabling the Schedule On checkbox. Once a schedule has been set up for a device, this schedule can be copied to other devices by selecting **Apply to Other Stations** and highlighting the desired devices.

PC208W must be running to make scheduled calls, but it will make these calls even when minimized. For example, some people set up

scheduled data collection from their dataloggers, and put PC208W in the Windows Startup group. That way, it loads when they turn on their PC in the morning, and it proceeds to collect data or do other Tasks as scheduled throughout the day.

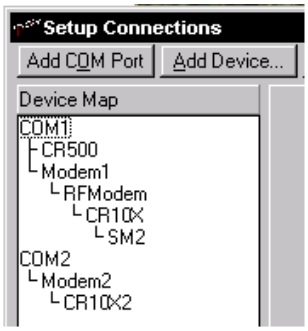
PC208W also allows you to set up other DOS or Windows programs to be run at scheduled times. These programs are referred to as **TASKS**. To set up a task to be run, press the **Add Task** button. On the **HARDWARE** tab, you can enter a unique name for the task to be run in the "Task Name" field. In the "Name of Program to Start" field, enter the command line for the program, batch file, or DOS command you want to execute. On the **SCHEDULE** tab, enter the "Calling Interval" for the task and ensure the "Schedule On" box is checked.

You will notice a list-box on all of the **SHEDULE** tabs called “After Call Do”. If you select the name of a device or Task from the Device Map, that device will be called or Task executed after the first Device is called (only for Scheduled Calls). In this way, you can set up a sequence of calls to dataloggers, or run other programs after that call (such as data processing applications like SplitR). Note that the After Call Do function is executed regardless of whether the first call is “successful”.

1.2.3 DEVICE MAP NOTES

The Device Map provides a pictorial representation of how each device is connected in your communications link. "Parent" and "child" connections are shown by right angle lines which connect the components, and the child components are indented beneath the parent component to help depict the logical order of the connection.

Review the example below:



The first item shown on the Device Map above is COM1—it is considered a parent component. The CR500 under it is a child. This communications link could be for direct connection from a desktop PC's COM1 port to the CR500.

The Modem1 connection is a link from the PC's COM1 port, that has a phone modem connected to it, with an RF modem connected to the phone modem, and a CR10X with a storage module connected to the RF modem. COM2 is a similar example, with a CR10X connected to COM2 through a phone modem.

Following are some hints which may help when setting up your Device Map.

- When you choose **Add Device**, a two column popup screen appears. The device which you want to add should be chosen from the left-hand column; the component which you want for the parent device should be chosen from the right hand column. PC208W will only allow you to choose valid parent components for each device; other devices are grayed out.

- Add your COM port first, and then attach each component in the order that it physically appears in your communications link.

- PC208W recognizes that modems are used in pairs—one at the computer and one at the datalogger. Therefore, once you attach a modem to the COM port, you do not have to add another modem for the datalogger site.

- Each device can be given a unique name. This can be helpful if you have a large network of datalogger sites and you want to remember which connections are for which sites. Device names can be edited by typing in a new entry directly in the component's Name field on the **Hardware**, **Data Collection**, or **Schedule** tab.

- If you need to change the logical order of a site's communication link, you can select a component with the mouse pointer and drag it to a new location. If PC208W is unclear of the connection to be made it will ask you to select whether the device should be

connected as a child component or on the same level. If the device you are moving to a new location has child components, these will be moved along with their parent device.

- If you delete a component with child components attached, the child components also will be deleted.

- If you inadvertently delete a communications link, it can be refreshed by choosing Cancel Edits (as long as you have not chosen Save Edits since the deletion).

1.2.4 EXAMPLE DEVICE MAPS

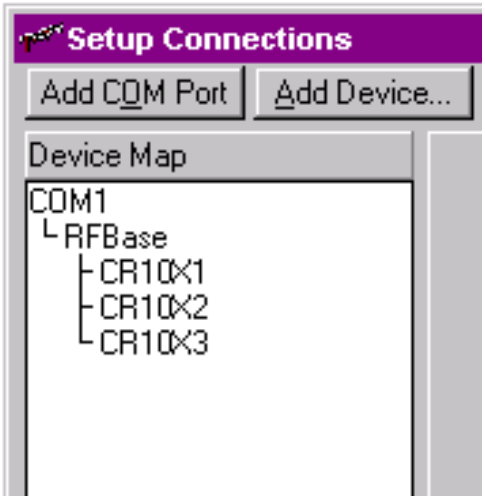
Refer to the examples below for help in setting up your own Device Map.

Step-by-step examples are provided in the PC208W help file. These can be accessed from the content page, by a right click on the device map in the Setup screen, or by searching for Cue Cards in the help file index.

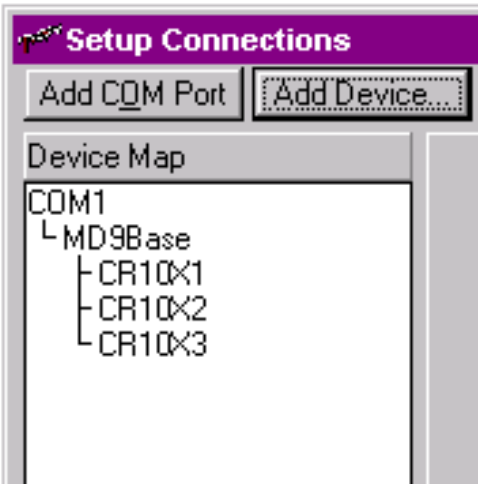
Direct Connect via an SC32A or RAD Short Haul Modem Pairs to a CR10X:

NOTE: Older short haul modems (SC95) are set up as generic modems with the datalogger attached. Refer to Appendix B for more information.

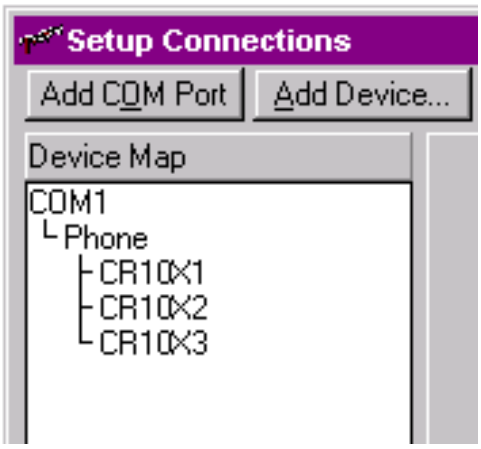
RF (RadioTelemetry) Connection to Three CR10Xs:



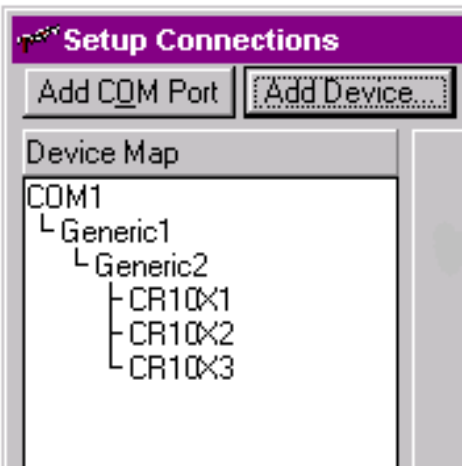
MD9 (Coax Multi-drop) Connection to Three CR10Xs:



Phone Modem Connection to Three CR10Xs:



Generic Modem Connection to Three CR10Xs:

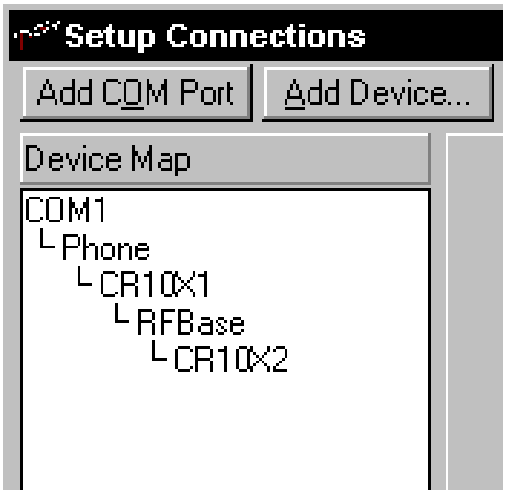


NOTE: Generic modems are used for connections or modems otherwise not supported by PC208W (such as spread-spectrum radio links or SC95 short haul modems). Generic modems require that the users specify the dialing and response strings needed to dial the modem. The user must also specify the delays and resolve timing issues. (See Appendix B.) It is possible to chain several generic modems if multiple strings and responses are required.

Phone to RF (or MD9) Connection to Three CR10Xs:

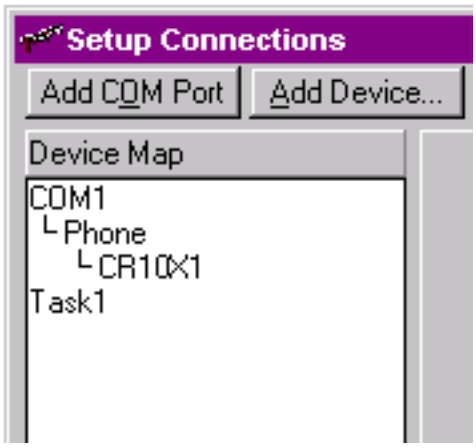
Modem Connection to CR10X with Storage Device:

Phone Modem to RF connection to a CR10X, with a CR10X at the RF Base:



NOTE: In reality, you can connect storage modules or PC card drives to most dataloggers. However, PC208W only allows valid connections in the Device Map where data can be retrieved from a storage module through a datalogger over telecommunications. PC208W allows SM192 and SM716 storage modules to be connected to CR10(X), CR500, or CR23X dataloggers.

Tasks (Scheduled Running of DOS or Windows Programs):



1.2.5 NOTES ON DATA COLLECTION AND FORMATS

As noted in Section 1.2.2, you must define how the datalogger should collect data and what file format to store the data. This information is entered on the Data Collection tab of the Setup window.

There are two basic options for specifying what data is collected: collect all the new data since the last collection or collect a fixed number of arrays on each call. Use the **Data Logged Since Last Call** option to build a complete archive. Use the **Most Recently Logged Arrays** option to collect a fixed number of the most recent arrays for each call.

When the **Most Recently Logged Arrays** option is chosen, the number of arrays to be collected is entered in the **Arrays to Collect** on each call field. Enter 0 if you want to collect all data on each call.

Data File Names for Areas 1 and 2

All dataloggers have a Final Storage area 1. The CR10, CR10X, and CR23X can be configured with two final storage areas. The default filenames for a device's Final Storage areas are taken from the device name. If a CR10X is named LOG1, the default files names are LOG1.DAT and LOG12.DAT. These default names can be changed by selecting the **Browse** button and entering a new name.

NOTE: By default, only Final Storage area 1 is set up for CR10s, CR10Xs, and CR23Xs. For data to be stored in Final Storage area 2, memory must be allocated in the datalogger, and the program must write data to this area.

File format

PC208W supports three formats for the collected data:

- ASCII, Comma Separated - This is the best choice for most applications. Data values are separated by commas with no leading zeros and with each array on a new line. This is the easiest format for importing into other software; it's reasonably compact and can be viewed directly.
- ASCII, Printable - Numbers are aligned in columns and preceded by a 2-digit ID indicating the array position. The column IDs repeat after 100 columns. Each row in the data file has up to eight columns before a new line is started. This format is easiest to read manually (i.e., printed copy) but is difficult to import into spreadsheets and other applications. SPLIT can be used to convert to other formats. This format is ambiguous for data arrays of 100 or more and should be avoided for purposes other than printing.
- Binary - This format is the most compact but must be converted before it can be used. SPLIT can be used to convert to other formats.

1.2.6 SETUP SAFETY TIPS

You can freely move between the different PC208W windows. This means you can open the **SETUP** screen while you are communicating with a datalogger or using one of the other PC208W programs. However, if you change the setup, PC208W will hang up all calls in process and will not call again until the changes are saved. It does this to prevent data storage pointers from being corrupted. Therefore, it is better to avoid changing the setup while communicating with a datalogger that will be affected by the change.

You must click on Save Edits for the device settings to take effect and continue any scheduled communications. You will also be prompted to Save Edits if you close the Setup window or PC208W if you have not yet saved the new device settings. If you find you have made a mistake while changing device settings but have not yet saved the edits, you can return to the previous configuration by clicking Cancel Edits. This only works for one previous setup because otherwise, PC208W may have called a datalogger and changed data storage pointers, making a return to previous settings unreliable.

You may drag and drop devices from one connection to another. If the place where you drop a device is ambiguous, PC208W will ask whether to place the moved device as a child or at the same level.

PC208W uses the computer clock to control some of its timing. Changing the PC clock while connected to a datalogger may cause unexpected results in PC208W.

1.3 CONNECT

brings up a window which gives you real time access to any of the dataloggers in your Device Map.

You can transfer programs to a datalogger, set the datalogger clock, collect data from the datalogger, collect data from the storage module, and display or graph datalogger measurements. From the **CONNECT** window you may connect to only one datalogger or storage module at any given time.

Only dataloggers and storage modules are shown on the **CONNECT** screen. You can switch to a different device by clicking on it with the mouse pointer. If you are connected to a device, PC208W will warn you and ask if it should close the current connection.

PC208W assumes a double click on a different datalogger means you want to switch regardless of the current connection. It will close the connection in an orderly fashion and switch to the new **CONNECT** screen. Note that PC208W saves the location and appearance of the screen for each datalogger, so switching to a new one may appear to make the screen "jump" from one position to another.

1.3.1 HANDY THINGS THAT MAY NOT BE OBVIOUS

TOOLS Tab

- Check the **Pause Clk Update** to stop the clock display. This can speed up manual data collection on a slow communication link. Be careful; if you pause the clock display AND the numeric display, there will be no visual clues as to when you are connected. In fact, PC208W may not stay connected as there is nothing to do. If you pause everything for more than about 40 seconds, the datalogger may time out and hang up the link from its end to conserve power.

- If you associate a datalogger program (.DLD file) in which you labeled your Input Locations, then you will have labels in the Numeric Display and Graphs.

- The **Pause Both** checkbox stops updating the numeric displays and graphs. This is useful to speed up manual data collection of large data files on slow communication links.

Ports/Flags

- A port or flag can be given a unique label. Right click on the port or flag and type in a new label. Labels are saved individually for each datalogger in the Device Map.

Numeric Display

- Right click on any cell in the Numeric Display to invoke a pop-up box. From this box you can choose to **Add** locations to the display; **Delete** locations from the display; **Setup** the Numeric Display's update interval, delete options, or decimal place settings.

- Double click on value labels to edit them. Changing the labels only affects the Numeric display and will not change the datalogger program.

- Double click on a value, then edit it and press <Enter> to store the new value in the datalogger's Input Location.

- You can drag and drop individual Input Locations to different locations on the display.

- Press and hold the <Ctrl> key before starting to drag a label, and the label will be inserted when you drop it. If <Ctrl> is not held, the label will replace the current label if one exists.

- You can drag a label from the ADD Inloc List and drop it on the numeric display instead of using the **Paste** button. You can drag and drop multiple locations at once by first highlighting labels using conventional <Ctrl>-click and <Shift>-click keystrokes.

- A cell must be highlighted on the numeric display in order to paste new input locations.

- <Ctrl>-click and <Shift>-click allow you to highlight multiple Input Locations for a subsequent DELETE.

Graphs

- If you are using the "power of 10 autoscaling" and a trace goes off scale, select the **Rescale** button to scale the value back to the Y-axis again.

- You can leave one or more graphs up with all other PC208W windows minimized, but if you close the **CONNECT** screen or the main PC208W toolbar, the connections are dropped and the graphs are also closed.

- Use the **Capture** button to save a graph to a WMF (Windows Metafile) format. This file can be printed or imported into other programs.

TERMINAL Tab

- If you are not connected to a datalogger, the **Open Port** button will allow you to manually dial and work with the communication devices. If you are connected, selecting the **TERMINAL** tab will stop all other communication with the datalogger until you return to the **TOOLS** tab. To enter the Remote Keyboard Mode on the datalogger press <Enter> to get an "*" and then press "7H <Enter>" (CR10(X), CR500, or CR23X) or "2718H <Enter>" (any datalogger). **BE CAREFUL!** This mode allows you to delete program instructions and may result in erasing the datalogger's memory!

- If you enter the Remote Keyboard Mode, be sure to press the ' *0 ' keys sequence to get a LOG1 message from the datalogger before you leave the **TERMINAL** screen so the datalogger is left in a known state.

1.3.2 CONNECT SAFETY TIPS

- Leaving PC208W connected to a datalogger keeps the communication link to the datalogger open and may prevent scheduled collection or callback from other dataloggers which use the same COM port.

- If you pause the clock display and the numeric display, there will be no visual clues as to when you are connected. In fact, PC208W may not stay connected as there is nothing to do. If you pause communication for more than about 40 seconds, the datalogger may time out and hang up the link.

- If you are connected, selecting the **TERMINAL** tab will stop all other communication with the datalogger until you return to the **TOOLS** tab. If communication between the computer and datalogger remains idle for more than about 40 seconds, the datalogger may time out and hang up the link. When the datalogger times out, you may get odd characters on the screen.

- During data collection if the disk or other storage devices becomes full, an error will be written to the SWF log and the data collection attempt will be halted. Once space is freed up on the disk or a new storage device is attached and data collection is resumed, some duplicate data may be written to the file.

1.4 STATUS

displays information for all the dataloggers on the Device Map at one glance. You can check the status of data collection including scheduled calls, errors, retries, and how much data was collected. You can also launch windows displaying SWF (Status, Warning, and Fault) messages or Low Level I/O (input/output) to monitor the binary traffic between the PC and datalogger. Buttons are also provided to conveniently collect data, stop data collection, turn on scheduled calls or reset retries.

1.4.1 DEVICE MAP AND STATION MAP

The top portion of the Status window shows a summary for all the devices you have set up. Information is displayed on data collection attempts and the next scheduled call. A checkbox in the lower left portion of the screen allows you to choose between viewing information on all the devices or just the stations (dataloggers). You can also highlight one or more dataloggers to manually collect data or manually terminate a collection in progress.

When a device is highlighted, you can press the right mouse button to bring up a list of data collection related options. You can **Toggle On/Off** the device's schedule; **Collect Data** from the device; **Collect Data** and **Do Task** (that is, perform any "After Call Do" items); **Reset Retries** to 0, or **Discontinue** the current data collection process.

This display gives a variety of diagnostic information summarized below.

On-line

O - the device is currently on-line or being dialed.

T - the device, usually a modem, has been dialed and is currently transparent (active but not the end of the link).

M - the COM port is being monitored for datalogger initiated communication (callback).

Error Rate

This is a running average of the errors in the last 20 communication attempts. **(n)** will appear following the error number if the last attempt to communicate failed. "n" is the number of successive failures. Typically the higher the error rate number, the worse the communication link.

Last Call

Indicates when the last call or attempt to call finished (even if it failed).

Collected

Indicates the last area collected, how many Final Storage locations were collected, and how many Final Storage locations should have been collected. (The CR10, CR10X, and CR23X are the only supported dataloggers with a Final Storage area 2.) For example, Area1 7/7 indicates that seven Final Storage locations were collected from Area 1 on the last call. Area1 0/0 indicates that no locations where collected on the last call, probably because there was no new data. Area1 0/10 would indicate that 10 locations were attempted but none were collected, possibly due to a link failure.

Next Call

Indicates the next time to call. This is based on the scheduling and retry parameters entered on the **SCHEDULE** tab for the datalogger in the SETUP window. If no date and time are displayed, scheduling is not enabled. If a date and time are displayed, but are grayed out, the schedule has been paused by checking the Pause Schedule check box at the bottom of the status screen. A device's schedule can be toggled on or off by selecting the device, pressing the right mouse button, and choosing Toggle On/Off or by pressing the Toggle On/Off button.

1.4.2 LOW LEVEL I/O

Logging low level I/O is typically used to troubleshoot a communication problem.

Select the **View I/O** button to display the actual serial communications for all the serial ports. The columns on the left contain the characters transmitted by PC208W and those on the right indicate characters received. Binary (unprintable) characters are displayed as a period "." character. Since communication for all serial ports is displayed and can occur concurrently, it may be helpful to turn scheduling off for all other sites when troubleshooting a particular site.

The Log I/O to Disk checkbox is used to enable logging to a file in PC208W's working directory. A separate file is used for each serial port. The files are named IO$COM1.LOG for COM1, IO$COM2.LOG for COM2 and so on. These files use a hexadecimal format so binary characters can be viewed.

1.4.3 COM SWF MESSAGES

Messages pertaining to datalogger communications can be displayed by selecting the **View Messages** button. Warning and Fault messages are always displayed and logged to disk in a file named SWF$.LOG. Status messages are also displayed if you enable the Show Status Messages checkbox and are logged if you enable the Log Status Msgs box. The log files will grow to a given size and then will start to overwrite (ring) themselves preventing them from consuming all disk space.

The first character of the message indicates the type of message—Status, Warning, or Fault. The time the message occurred is given next. The format is DD HH:MM:SS, where DD is the day of the month, HH is the hour, MM the minutes, and SS the seconds. The name of the relevant device and actual message follow the time.

1.4.4 PAUSE

When PC208W is first started, all scheduled data collection is paused for 15 seconds. The text for the box in the lower right corner of the status window reads Startup Pause. After 15 seconds, the Pause Schedule box will be either checked or unchecked, depending on its state when you last closed PC208W. When Pause Schedule is enabled, the Next Call information for any devices with scheduled data collection will be grayed out.

1.5 COMMAND LINE PARAMETERS

Command line parameters are used for further automating data collection. Using command line parameters, you can start PC208W and automatically connect to stations, collect data, and exit the program.

Command line parameters are entered from the command prompt, from the Start | Run menu, or specified on the executable line of a shortcut. Up to 150 characters (including spaces) can be specified on a command line. Each command begins with a forward slash, /. A dash, -, should separate the command and any parameters it may have. If multiple commands are used on one command line, they should be separated by a space. Following are a list of command line parameters that can be used with PC208W.

Parameter	Description
/A-*filename*	The /A command is used to specify an alternate file name for data collected from the active station's Final Storage Area 1. *Filename* is the user-specified alternate file name. Syntax: /S-*station* /A-*filename* /C
/A2-*filename*	The /A2 command is used to specify an alternate file name for data collected from the active station's Final Storage Area 2. *Filename* is the user-specified alternate file name. Syntax: /S-*station* /A2-*filename* /C
/C	The /C command is used to collect data from the active station.
/D-*seconds*	The /D command is used to insert a delay, in seconds, before the next command is executed. *Seconds* is the number of seconds to delay.
/F-*Flag#*	The /F command is used to toggle the specified flag of the active station. To specify an extended flag for the CR23X, precede the *Flag#* with an X.
/G-*graph#*	The /G-*graph#* (where *graph#* is 1, 2, or 3) is used to invoke Graph number 1, 2, or 3 for the active station.
/I-*InLoc#*:*Value*	The /I command is used to insert a value into the specified input location of the active station.
/K	The /K command is used to set the clock for the active station.
/M	The /M command is used to invoke the Numeric Monitor for the active station.
/N	The /N command is used to open PC208W and run it in a minimized window.
/P-*Port#*	The /P command is used to toggle the specified port of the active station.
/Q	The /Q command is used to terminate execution of PC208W.
/S-*Station*	The /S command is used to open a session with the specified station. This command must precede all other commands.
/T-*Taskname*	The /T command is used to execute a Task that has been set up in the device map. The /T command can also be used to collect data from a datalogger (see discussion below).

EXAMPLE:

Entering the command line:

PC208W.exe /S-CR10X1 /D-5 /K /A-newdata /C /Q

will

1. Open PC208W
2. Connect to the station named CR10X1
3. Delay for 5 seconds
4. Set the datalogger clock
5. Set name for data collected from final storage area 1 to “newdata”
6. Collect data
7. Exit PC208W

NOTES:

Because commands are separated by a space, the commands will not work if there are other spaces in the command line.

Command line functions will not work from Windows 3.1's "shell" to DOS. They must be run from a desktop icon.

Data Collection Using /C versus /T

The /C command is used to collect data from the datalogger specified by the preceding /S command. Once data collection is completed, PC208W remains connected to the datalogger until it is closed or until another /S command is executed. Though a datalogger may have an

After Call Do item associated with it, the After Call Do is not executed.

The /T command is used to execute a task from the command line. However, it can also be used to collect data from a datalogger. When the /T command is executed, PC208W connects to the datalogger, collects data, and then terminates the connection. If the datalogger has an associated After Call Do item, that item, and any subsequent After Call Do items, are executed.

If a resource intensive task is executed after collecting data from a datalogger, it may be advantageous to use the /T option to collect the data. If the /S-station /C combination is used, PC208W could return telecommunication errors if system resources are not available to maintain uninterrupted communication with the datalogger.

A Note About Directories and Command Line Entries

PC208W's default installation creates the **PC208W** directory and two subdirectories, **Bin** and **Inifiles**. PC208W stores most of the files it needs for program execution, including the PC208W.exe file, in the **Bin** directory. The **PC208W** directory is set up as the working directory; most of the files created by the user are stored in this directory.

One file that is integral to the operation of PC208W is the PC208W.dnd file. This dnd file contains the information for your Device Map. When you add a datalogger or other device to the Device Map, or make changes to an existing device, the information is stored in the **PC208W** working directory by choosing the **Save Edits** button.

When PC208W is started it reads the dnd file from the active directory. If you are running command line entries from a command prompt, PC208W considers the active directory to be the directory from which the commands were invoked. Therefore, if you type the following command at the C:\ prompt:

C:\PC208W\bin\pc208w.exe /S-CR10X1 /C /Q

PC208W will look for the dnd file on the C:\ drive. When it doesn't find a dnd file, it will create a new one that contains only COM1 (none of your devices will exist on the new device map).

One way to resolve this is to run the above command from a batch file that directs your computer to the appropriate directories. For example:

```
cd pc208w
bin\pc208w.exe /S-CR10X1 /C /Q
```

This batch file, which would be run from the C:\ prompt, would set the active (working) directory as **PC208W** and run PC208W.exe, which is stored in the **Bin** directory, using the specified command line parameters.

SECTION 2. PROGRAM: EDLOG

2.1 INTRODUCTION

Choosing PROGRAM from the main PC208W toolbar invokes the program EDLOG. EDLOG is a tool for creating, editing, and documenting programs for the CR7, CR10, CR10T, CR10X, 21X, and CR23X dataloggers.

EDLOG annotates the numeric instructions and parameters that are normally entered into the datalogger using the datalogger keyboard/display. Instructions and parameters can be picked from lists or entered manually, and are easily edited, moved, or deleted.

Some highlights of EDLOG's features are listed below.

Context-Sensitive Help

If you need help with an instruction or parameter, *press the right mouse button***.** The right mouse button brings up editor options, pick lists or pop-up help for where you're pointing. Detailed help is available for the cursor location by pressing <F1> or from a button on the pick list. Help, pick lists, and edit functions are also available from the menus or toolbar.

Cut and Paste

Several programs can be open simultaneously, simplifying cutting and pasting between programs for the same datalogger.

Input Location Labels

Measurement results and calculations are stored in datalogger Input Locations. EDLOG allows labeling these locations. For many applications, EDLOG will manage the use of Input Locations. A new Input Location label is automatically assigned the next available Input Location number (address). That Input Location can be picked from a list for further calculations or output. The Input Location Editor (Section 2.7.3) allows the Input Locations to be edited (moved, inserted, deleted); the Input Location numbers are automatically changed wherever the labels appear in the program.

When a section of code is pasted into a program, EDLOG will automatically use existing locations for matching labels and assign locations to new labels. All location numbers in the pasted code are updated accordingly.

The programmer that wants to assign labels to specific Input Locations may find it easiest to enter the labels in the Input Location Editor first and then pick them from a list as the program is written.

Expression Interpreter

Mathematical calculations can be written algebraically using Input Location labels as variables. For example, the following expression could be used to create a location that holds the temperature in Fahrenheit (TempF) which is calculated from the temperature in Celsius (TempC):

TempF=TempC*1.8+32

EDLOG will compile the expression into datalogger instructions, creating locations for intermediate steps if necessary (Section 2.5).

Compiler

EDLOG compiles the program to check for errors and create the file that is downloaded to the datalogger. The download file (*.DLD) excludes all comments and incorporates the instructions required to calculate the expressions. During the compile step, a Final Storage Label file (*.FSL) that lists the labels used by the program and a Program Trace Information file (*.PTI) are also created (Section 2.9).

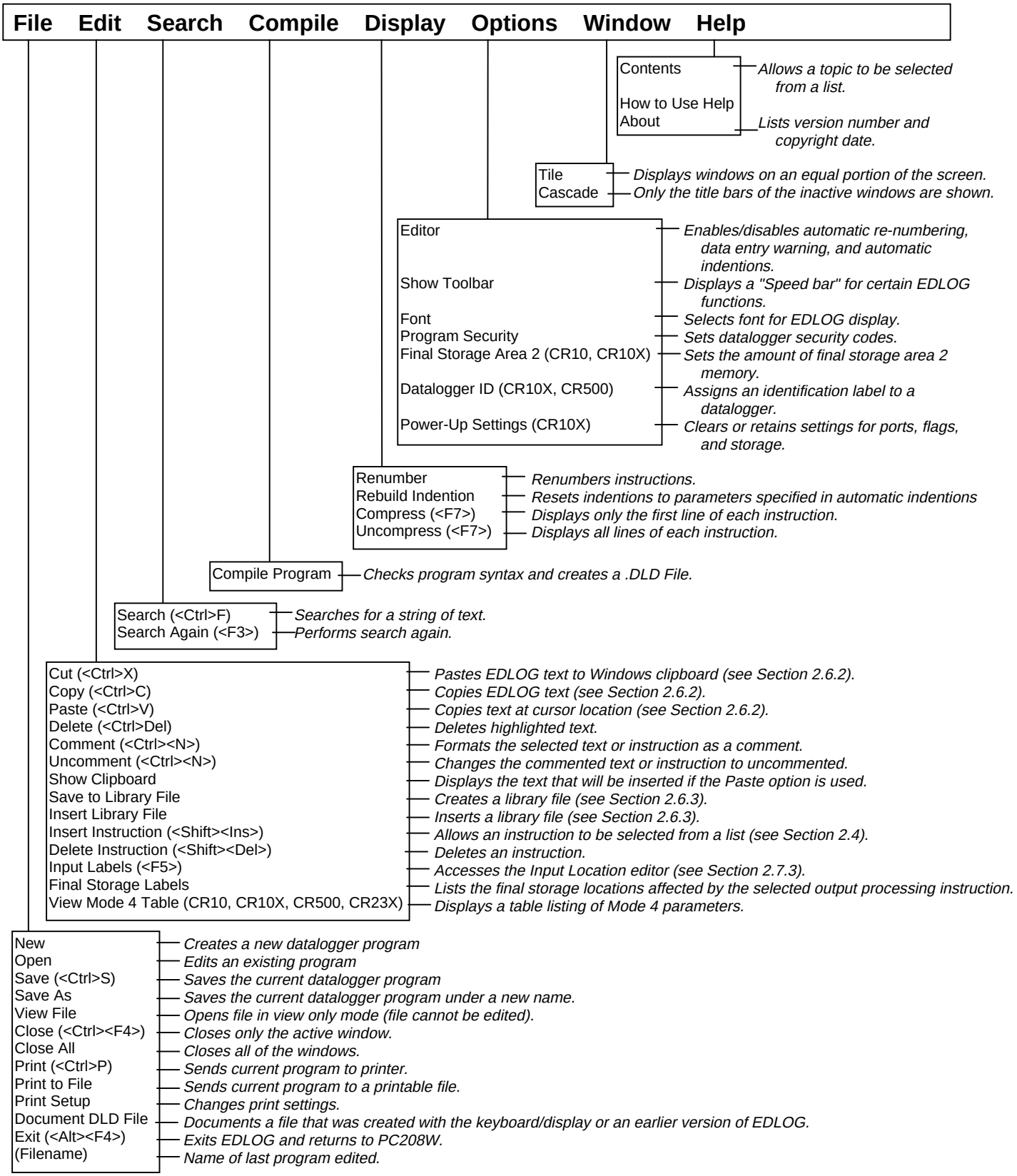


FIGURE 2.1-1. EDLOG Menu

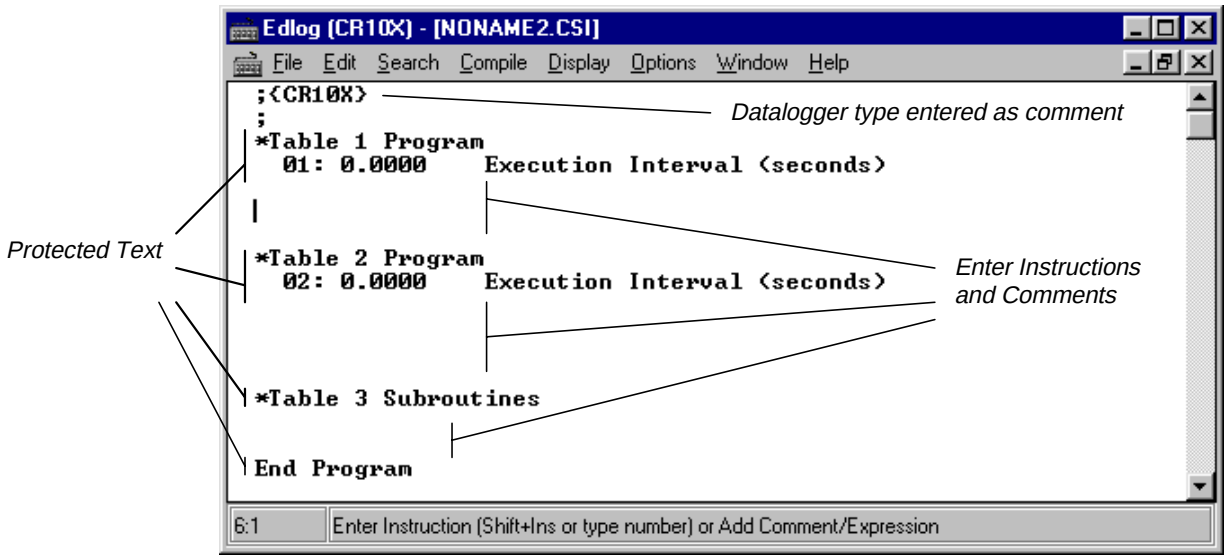


FIGURE 2.2-1. Program Development Template

2.2 GETTING STARTED

2.2.1 CREATING A NEW EDLOG PROGRAM

The steps for creating a new datalogger program are:

1. Start EDLOG from PC208W by clicking on the toolbar icon labeled PROGRAM.
2. Select FILE | NEW.
3. When prompted, select the datalogger type.
4. Develop and edit your datalogger program (Sections 2.3 and 2.4).
5. Select FILE | SAVE to save and name the program.
6. When prompted, select YES to compile the program.
7. Exit EDLOG by selecting FILE | EXIT.

2.2.2 EDITING AN EXISTING PROGRAM

Instead of using FILE | NEW, load an existing program by selecting FILE | OPEN.

2.2.3 DOCUMENTING A DLD FILE

FILE | DOCUMENT DLD FILE is for creating a documented (.CSI) file from the instructions in a .DLD file. This might be used if a program was keyed directly into the datalogger (or modified with the keyboard) and then retrieved in a datalogger session. After creating a documented file, make sure to save it using FILE | SAVE.

Programs created with versions of EDLOG earlier than 6.0 were stored with the instruction description and comments in a *.DOC file and the download code in the .DLD file. These programs, *without the programmer's comments,* can be imported into this version of EDLOG with FILE | DOCUMENT DLD FILE. To convert these programs and retain the programmer's comments, use DOC2CSI (Section 6).

2.2.4 THE PROGRAM DEVELOPMENT TEMPLATE

When FILE | NEW is selected, the template shown in Figure 2.2-1 appears in the **PROGRAM DEVELOPMENT** screen. A program consists of instructions, comments, and expressions inserted into Program Tables 1 and 2 and the Subroutine Table 3.

The Table Headers and Execution Intervals are protected text which cannot be deleted or commented out. When the cursor is moved to the Execution Interval line, the value for the execution interval is highlighted. A numeric value must be entered or the instructions in the table will not be performed. On the blank lines in each of the program tables, instructions can

be inserted or comments and equations can be freely typed in.

Like the execution interval, instructions consist of protected text and fields where parameters are entered. With data entry warnings enabled (default), EDLOG prevents leaving a field with an incorrect value.

2.3 COMMENTS

Comments are entered to describe the program for the programmer or future users. Comments are ignored by the compiler; they can be entered anywhere in the EDLOG program. A semicolon (;) is used to mark comments. When the compiler finds a semicolon in a line, it ignores the rest of the line. Instructions can be commented out to test various versions of a program or to delete a measurement.

In addition to typing a semicolon at the beginning of each line while entering comments, there are several ways to comment and uncomment lines, instructions, or blocks of code:

- Select “comment” or “uncomment” from the **right button menu** (highlighted block only).
- Select FILE | COMMENT or FILE | UNCOMMENT from the EDLOG toolbar.
- **Press <ctrl> n.** The line, instruction or highlighted block will become a comment.
- **Press <shift><ctrl> n.** The line, instruction or highlighted block will be uncommented.
- **Press <End>** to move to the right of the protected text of an instruction and enter a comment.

EDLOG will not allow a portion of an instruction or the table execution intervals to be commented out.

2.4 INSTRUCTIONS

Instructions must be entered in one of the program tables. Instructions are added to a program with the Insert Instruction Dialog box. (Programmers with a great memory can type the instruction number on a blank line and press <enter>.) To get the Insert Instruction box, either:

- Point the mouse cursor at the blank line where you want the instruction and press the right button.
- Select EDIT | INSERT INSTRUCTION
- Press <shift> <Insert>

Select the instruction with the mouse or by moving the cursor keys and click OK. If you need more information on an instruction, highlight the instruction and press the **Help** button.

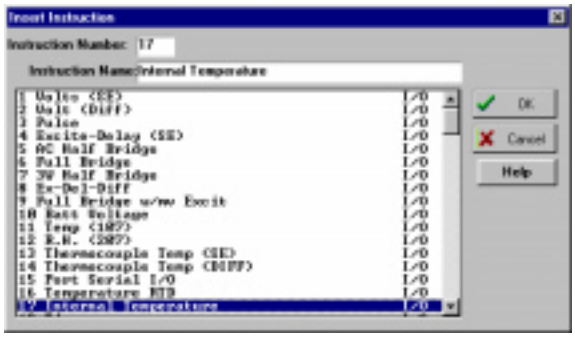


FIGURE 2.4-1. Insert Instruction Dialog Box

2.4.1 ENTERING PARAMETERS

When an instruction is inserted, the cursor moves to the first parameter. Type the parameter's value and press <enter> to move to the next parameter. For help with the parameter:

- Point the mouse cursor at the parameter and press the right button. This brings up a Dialog box for selecting the parameter or a pop up description of what needs to be entered.
- Press <F1>. This opens the help system to the detailed description of the instruction and parameters.

Repeat the process until all of the parameters have been entered.

> **NOTE:** The instructions and their parameters are datalogger specific; to find the description of an instruction, refer to the specific datalogger manual.

2.4.2 DATA ENTRY WARNING

When the Data Entry Warning is active, a warning is displayed immediately after an invalid input has been entered for an instruction's parameter. The warning lists the valid inputs. A valid input must be entered before advancing to the next parameter.

By default, the Data Entry Warning is enabled. To disable the Data Entry Warning, select OPTION | EDITOR. Use the <tab> key to place the cursor on the brackets and press the <space> bar or select the brackets with the computer's mouse. A space instead of an 'X' between the brackets indicates this option is disabled.

2.5 EXPRESSIONS

Expressions are a shortcut for processing numbers and input data. They must be entered into one of the program tables. When a datalogger program that contains an expression is compiled, the appropriate instructions are automatically incorporated into the .DLD file.

2.5.1 RULES FOR CREATING EXPRESSIONS

- Expressions must be set equal to the label of the Input Location that will store the result. The label must be to the left of the expression. See Section 2.7 for more information about Input Location labels.
- Expressions can have both fixed numbers and Input Location labels. Input Locations can ONLY be referenced by their label; each number is always assumed to be a constant.
- Floating point numbers are limited to five (or six for the CR10X) digits plus the decimal point and sign.
- The operator(s) and/or function(s) used in the expression must be on the operator or function list (Table 2.5-1).
- Numbers and labels that are immediately after a function must be enclosed in parentheses.
- Several operators and/or functions can be used in one expression. Operations and functions that are enclosed in parentheses are calculated first; the inner most parentheses are evaluated first.
- To continue an expression to the next line, end the first line with an underscore (_).

TABLE 2.5-1. Operators and Functions

Operators

*	(multiply)
/	(divide)
+	(add)
-	(subtract)
^	(raise to the power of; enclose negative values in parentheses)
@	(modulo divide)
e	(for CR10(X) only, scientific notation; 6e-1=0.6)

Functions

COS	(cosine; angle in degrees)
SIN	(sine; angle in degrees)
TAN	(tangent; angle in degrees)
COTAN	(cotangent; angle in degrees)
ARCTAN	(arctangent; angle in degrees)
ARCSIN	(arcsine; angle in degrees)
ARCCOS	(arccosine; angle in degrees)
ARCCOT	(arccotangent; angle in degrees)
SQRT	(square root)
LN	(natural logarithm)
EXP	(exponent of e; EXP(2) = e^2)
RCP	(reciprocal; RCP(4) = 1/4 = 0.25)
ABS	(absolute value)
FRAC	(takes the fraction portion; FRAC(2.78)=.78)
INT	(takes the integer portion; INT(2.78)=2)

2.5.2 Examples

Below are examples of valid expressions:

```
Zee    =  Vee+Ex
es     =  tee^(-2)
Root   =  SQRT(ABS(data))
avg    =  (data1+data2+data3+data4+data5)/5
length =  SQRT((adj^2)+(opp^2))
TempF  =  (TempC*1.8)+32
```

The following section of an EDLOG program for the CR10X uses an expression to convert Temperature from Celsius to Fahrenheit:

Execution Interval = 10 sec

```
1:      Temperature (107)  (P11)
  01:       1          REPS
  02:       2          Channel
  03:       1          Excitation Channel
  04:       2          Loc [TempC]
  05:       1          Mult
  06:       0          Offset
```

TempF = (TempC*1.8)+32

When this program is compiled, the .DLD file contains the following instructions. The last 5 instructions calculate the expression.

```
1:      P11          (Temperature, 107)
  1:       1
  2:       2
  3:       1
  4:       2
  5:       1.0
  6:       0.0

2:      P31          (Z=X)
  1:       2
  2:       5

3:      P30          (Z=F)
  1:       1.8
  2:       0
  3:       3

4:      P36          (Z=X*Y)
  1:       3
  2:       5
  3:       5

5:      P30          (Z=F)
  1:      32
  2:       0
  3:       3

6:      P33          (Z=X+Y)
  1:       3
  2:       5
  3:       6
```

2.6 EDITING EDLOG PROGRAMS

Table 2.6-1 provides a list of keystrokes which can be used in editing programs and moving around in the **PROGRAM DEVELOPMENT** screen.

TABLE 2.6-1. Editor Keystrokes

PgUp	Page Up
PgDn	Page Down
Up Arrow	Move Up One Line
Down Arrow	Move Down One Line
Right Arrow	Move One Character Right
Left Arrow	Move One Character Left
<ctrl> Home	Move Cursor to Beginning of File
<ctrl> End	Move Cursor to End of File
<ctrl> PgUp	Move Cursor to Top of Screen
<ctrl> PgDn	Move Cursor to Bottom of Screen
<Enter>	Move to Next Field or Create New Line
<shift> Ins	Select an Instruction from a Dialog Box
<ctrl> Right Arrow	Move Instruction 1 Tab Right (Cursor on Parameter)
<ctrl> Left Arrow	Move Instruction 1 Tab left (Cursor on Parameter) or Move from Input Location label to Input Location number.
<ctrl> n	Comment out a Line or Instruction
<shift> <ctrl> n	Uncomment a Line or Instruction
<end>	Move to end of line, Add a comment if on an Instruction
<ctrl>C	Copy highlighted text
<ctrl> X	Cut highlighted text
<ctrl>V	Paste clipboard
Del	Delete character to right or highlighted text
<shift> Del	Delete the Instruction or Line Under the Cursor
<Esc>	Close Dialog Box

2.6.1 EDITING COMMENTS, INSTRUCTIONS, AND EXPRESSIONS

To edit Comments, Expressions, and Instruction parameters, move the cursor to the appropriate text and retype it. To delete an instruction when the cursor is somewhere within the instruction, select EDIT | DELETE INSTRUCTION or press <shift> Del. An instruction or block of instructions can also be highlighted and deleted with the delete key.

2.6.2 CUT, COPY, PASTE, AND CLIPBOARD OPTIONS

EDIT | CUT, EDIT | COPY, and EDIT | PASTE allow sections of the program to be moved or copied to another area of the program or between programs. EDIT | SHOW CLIPBOARD shows the contents of the clipboard.

> **NOTE:** You cannot move, copy, delete or comment out protected text (Tables, Execution Intervals). To move, copy or delete an Instruction, the entire instruction, including all of the parameters, must be selected.

Cutting and pasting between datalogger programs should be between programs for the same datalogger type only. Instructions and parameters may differ between dataloggers. The compiler will catch many of these errors; however, this may be at the expense of much time and confusion.

2.6.3 LIBRARY FILES

Library files can be created to store portions of programs. The library file can be inserted into a program at a later time. To create a library file, select the text to be stored and then EDIT | SAVE TO LIBRARY FILE. When the window appears, type in the library filename.

To insert a library file in a program, move the cursor where the file is to go and select EDIT | INSERT LIBRARY FILE. Library files created for one type of datalogger type should not be used in programs for a different type; i.e., a file for a CR10X should not be used in a program for a 21X.

2.6.4 RENUMBERING THE INSTRUCTIONS

When Automatic Renumbering is enabled, the instructions are automatically renumbered whenever instructions are inserted or deleted. By default, Automatic Renumbering is enabled.

To improve performance on slow computers, the Automatic Renumbering can be disabled by selecting OPTIONS | EDITOR. A blank in the box next to Automatic Renumbering indicates this option is disabled. When Automatic Renumbering is disabled, the user can periodically renumber the instructions by selecting DISPLAY | RENUMBER.

2.6.5 COMPRESS VIEW

When DISPLAY | COMPRESS VIEW is selected, the first line of each instruction is displayed. The compressed view makes it easier to see the program structure and to move around in the program.

Instructions cannot be edited in the compress View's window. Use DISPLAY | UNCOMPRESS to switch back to the full view or use the <F7> function key to toggle between the compressed and full views.

2.7 INPUT LOCATIONS

Each Input Location has an Input Location number and a label that appear whenever the Input Location is referenced in the program. EDLOG automatically assigns Input Location numbers as labels are entered.

2.7.1 ENTERING INPUT LOCATIONS

When a parameter requires an Input Location, the cursor automatically advances to where the label is keyed-in. When a new label is entered, the next available Input Location number is automatically assigned to that label. To select an existing label from a list, press the right mouse button or <F6>.

Labels can have up to nine characters. The first character must be a letter. The allowed characters are letters, numbers, and _. The following labels are reserved for expressions and should not be entered by the user: CSI_R, CSI_2, CSI_3, . . . CSI_95.

To move to the parameter to enter the Input Location number instead of the label, use the mouse or press <ctrl> left arrow.

2.7.2 REPS

Many input/output and output processing instructions have a repetitions parameter. Repetitions (REPS) allow one programming instruction to measure several identical sensors or to process data from several Input Locations. When REPS are greater than 1, the Input Locations are assigned consecutive numbers (e.g., with REPS of 2 and LOC of 5, the Input Locations are 5 and 6). Each rep label is the initial label with a "_" and the next consecutive number.

For example: With 3 REPS and a label of "data" the labels for each REP are: data_1, data_2, and data_3.

Only the first location of an output processing instruction is linked to the label. Reps of input/output instructions and output processing instructions are not linked.

For example:

In the following section of an EDLOG program, the TempC and BatteryV Input Locations are sampled with one sample (P70) instruction, with the REPS parameter of 2.

```
10:    Temperature (107)  (P11)
  01:        1          REPS
  02:        2          Channel
  03:        1          Excitation Channel
  04:        1          Loc [TempC]
  05:        1          Mult
  06:        0          Offset

11:    Battery, Volt  (P10)
  01:        2          Loc [BatteryV]

12:    If time is  (P92)
  01:        0          minutes into interval
  02:       60          minute interval
  03:       10          Set high Flag 0 (output)

13:    Sample  (P70)
  01:        2          Reps
  02:        1          Loc [TempC]
```

NOTE: If an Input Location is inserted between the TempC and BatteryV location, the inserted location will be sampled instead of BatteryV.

2.7.3 INPUT LOCATION EDITOR

Input Location labels can be entered and edited by using the Input Location Editor (Figure 2.7-1). To access the Input Location Editor, select EDIT | INPUT LABELS.

The following commands are available from the Input Location Editor Edit menu:

INSERT (<F2>) — inserts blank Input Locations. This is used to provide space for new input labels between existing labels. This automatically changes the Input Location numbers for all of the labels that are after the inserted location.

DELETE (<F3>) — deletes the Input Location label, flags, number of reads and writes, and block information for a designated location number. Whenever the datalogger program references a deleted location label, the Input Location's number automatically becomes 0.

MOVE (<F4>) — moves the Input Location to a different number. This might change several Input Location numbers.

TOGGLE MANUAL (<F5>) — allows the programmer to manually mark a location as in use, and to remove this manual designation. This is used for burst mode, indexed loops, or other situations where it's not clear to EDLOG that the locations are being written to. Input Locations not marked as read, write, or manual are deleted by the optimize command.

OPTIMIZE (<F6>) — deletes Input Locations that aren't read, written to, or marked manually. Optimize tries to reduce the total number of locations used by moving Input Location labels down (in location number) to fill in unused locations. This might change several Input Location numbers. Any changes in location number made by the optimize command are reflected in the EDLOG program.

INSERT BLOCK (<F7>) – inserts a block of Input Locations with the manual flag set. The label for the locations is entered in the Base Name input area. The locations are labeled in the same manner as reps.

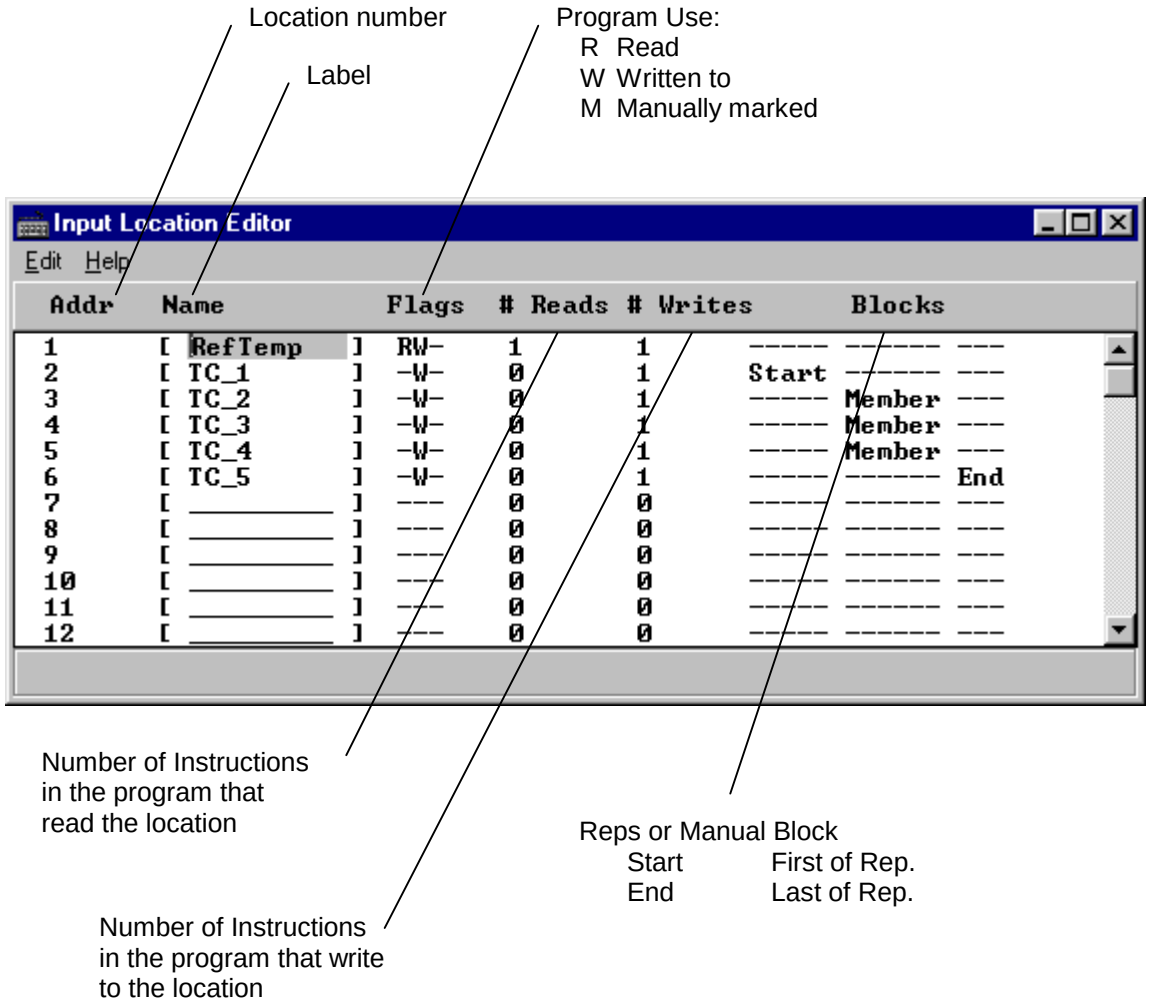


FIGURE 2.7-1. INPUT LOCATION EDITOR Screen

2.7.4 INPUT LOCATION ANOMALIES

In most instances, EDLOG will automatically assign Input Locations for locations which are generated by the datalogger program. An example of this is EDLOG's handling of Input Locations for the REPS parameter. Though only one Input Location is specified, if REPS is greater than 1, additional Input Locations are created by EDLOG (refer to 2.7.2 above).

However, there are certain instructions which generate multiple Input Locations for which EDLOG does not automatically allocate Input Locations. These are:

- Instruction 15, Serial I/O with Control Port
- Instruction 23, Burst Measurement
- Instruction 49, Spatial Maximum
- Instruction 50, Spatial Minimum
- Instruction 54, Block Move
- Instruction 75, Histogram
- Instruction 80, Set Active Storage Area
- Instruction 81, Rainflow Histogram
- Instruction 101, SDM-INT8
- Indexed input locations in a loop

Details on how EDLOG handles each of these instructions is given in Appendix A.

When these instructions are used in a program, the Toggle Manual feature (see 2.7.3 above) can be used to manually mark Input Locations for use by the program.

2.8 INDENTION

Indention is typically used with If Then/Else sequences and loops to provide a visual key to the program execution. Indention is to assist the person looking at the program; it has no meaning to the datalogger. If the programmer chooses to use indention, it can be done automatically or manually.

The settings for indention are found under OPTIONS | EDITOR. Turn on Automatic Indention by checking the box next to it. The distance for each indention (in spaces) is set on the same Dialog box.

To manually indent an instruction, place the cursor on one of the instruction's parameters and press either <ctrl> right arrow or <ctrl> left arrow; the instruction is indented the direction the arrow is pointing.

DISPLAY | REBUILD INDENTION resets all existing indentions and rebuilds automatic indentions. Automatic indentions may need to be rebuilt when editing instructions cause the indentions to get misaligned.

2.9 FILE TYPES

2.9.1 PROGRAM DOCUMENTATION, *.CSI

*.CSI files are the EDLOG program files that the user edits. When an EDLOG program is saved, EDLOG automatically adds a .CSI extension to the program's name. Existing .CSI files can be loaded into EDLOG by selecting FILE | OPEN. Although .CSI files are ASCII files, editing the files with a text editor can affect the EDLOG files so that they no longer compile.

2.9.2 DOWNLOAD FILES, *.DLD

.DLD files are files which can be downloaded to a datalogger. They are produced by compiling an EDLOG program or by retrieving a program from the datalogger. Whenever an EDLOG program is compiled, any existing .DLD files with the same name are overwritten. .DLD files that were created with an older version of EDLOG or created with the keyboard/display can be documented by selecting FILE | DOCUMENT DLD.

2.9.3 FINAL STORAGE LABEL FILES, *.FSL

.FSL files list the final storage labels. The .FSL files are used by SPLIT to help select output values. These files can be viewed with FILE | VIEW.

These files are produced by EDLOG when the program is compiled. Whenever an EDLOG program is compiled, any existing .FSL files with the same name are overwritten.

2.9.4 PROGRAM TRACE INFORMATION FILES, *.PTI

.PTI files show the execution times for each instruction, block (e.g., subroutine), and table as well as the number of final storage locations used per day. The execution times are estimates. .PTI files do not account for If commands, Else commands, or repetitions of loops. For some instructions, the execution times are listed as 0. This occurs when the execution time is difficult to calculate (e.g., Burst Mode).

.PTI files can be viewed with FILE | VIEW.

These files are produced by EDLOG when the program is compiled. Whenever an EDLOG program is compiled, any existing .PTI files with the same name are overwritten.

2.9.5 PRINTER OUTPUT FILES, *.TXT

.TXT files are EDLOG programs that can be printed or used with a text editor. They are produced by selecting FILE | PRINT TO FILE.

2.9.6 LIBRARY FILES, *.LBR

Library files allow a section of a program to be inserted into any EDLOG program. To avoid programming errors, the library file and the EDLOG file should be for the same type of datalogger.

SECTION 3. REPORT: SPLIT

3.1 OVERVIEW

SPLIT is a tool to aid the use and analysis of data collected with Campbell Scientific dataloggers. Its name comes from its function of splitting specific data from a larger data file. The most common application, as shown in Section 3.2, Getting Started, is to separate data collected on a particular interval from a data file containing data output at several different intervals.

In addition to simply splitting a data array collected at one interval from all the data collected by a datalogger, SPLIT applications include: data selection based on time or conditions, statistics, calculations, file reformatting, data quality checking (limit testing), table generation with report and column headings, and time synchronizing and merging of up to eight files.

Input Files (maximum of eight) are read by SPLIT, specific operations are performed on the data, and the results are output to a single Output File or print out. SPLIT creates a parameter file (filename.PAR) that defines which data files are read, what operations are performed on the data set, and where the final results will be saved. The parameter file may be saved and used again.

Input Files (Section 3.3.1) must be formatted in Printable ASCII, Comma Separated ASCII, Field Formatted ASCII, Final Storage (Binary) Format, or Raw A/D data.

Output files generated by SPLIT are Field Formatted (default), Comma Separated ASCII, or Printable ASCII.

SPLIT lends itself to experimentation. The processed data are displayed on the screen, giving immediate feedback as to the effect of changes or new entries to the parameter file. The Input File is not modified by SPLIT.

3.2 GETTING STARTED

The most common use of SPLIT is to separate data collected on a particular interval from a data file containing data output at several different intervals.

In the following example, hourly data are split from a data set that contains both hourly and daily data. The data were collected from Weather Station 1, a CR10X. The station was named WS1 when telecommunications were set up so the data are stored in the file WS1.dat. The CR10X was loaded with a program created by EDLOG named WS1.dld.

The hourly data, array 10, and the daily data, array 20, are intermixed in the data file (Figure 3.2-1)

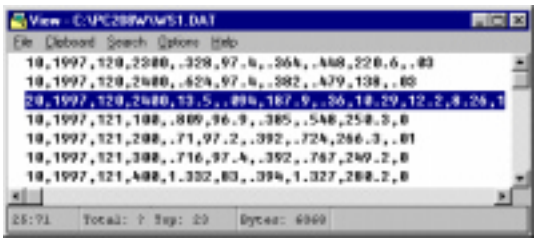


FIGURE 3.2-1. Data File

When EDLOG compiled WS1.dld, it also created the Final Storage Label file, WS1.FSL which lists the final storage locations for each data element (Figure 3.2-2).

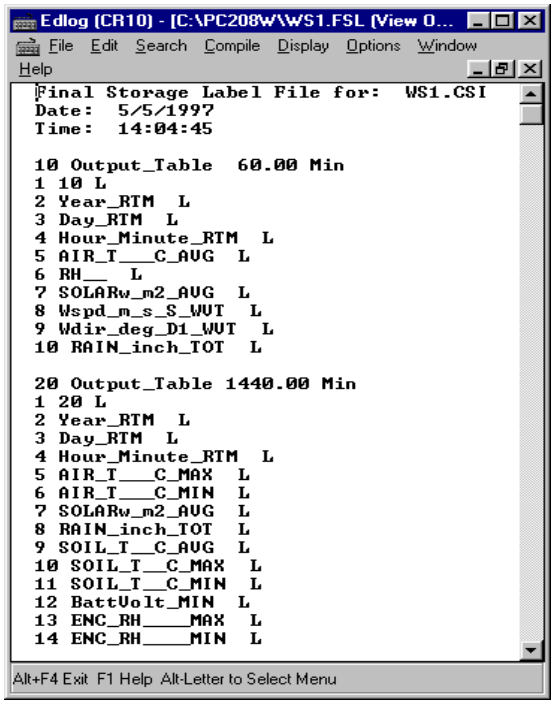


FIGURE 3.2-2. FSL file

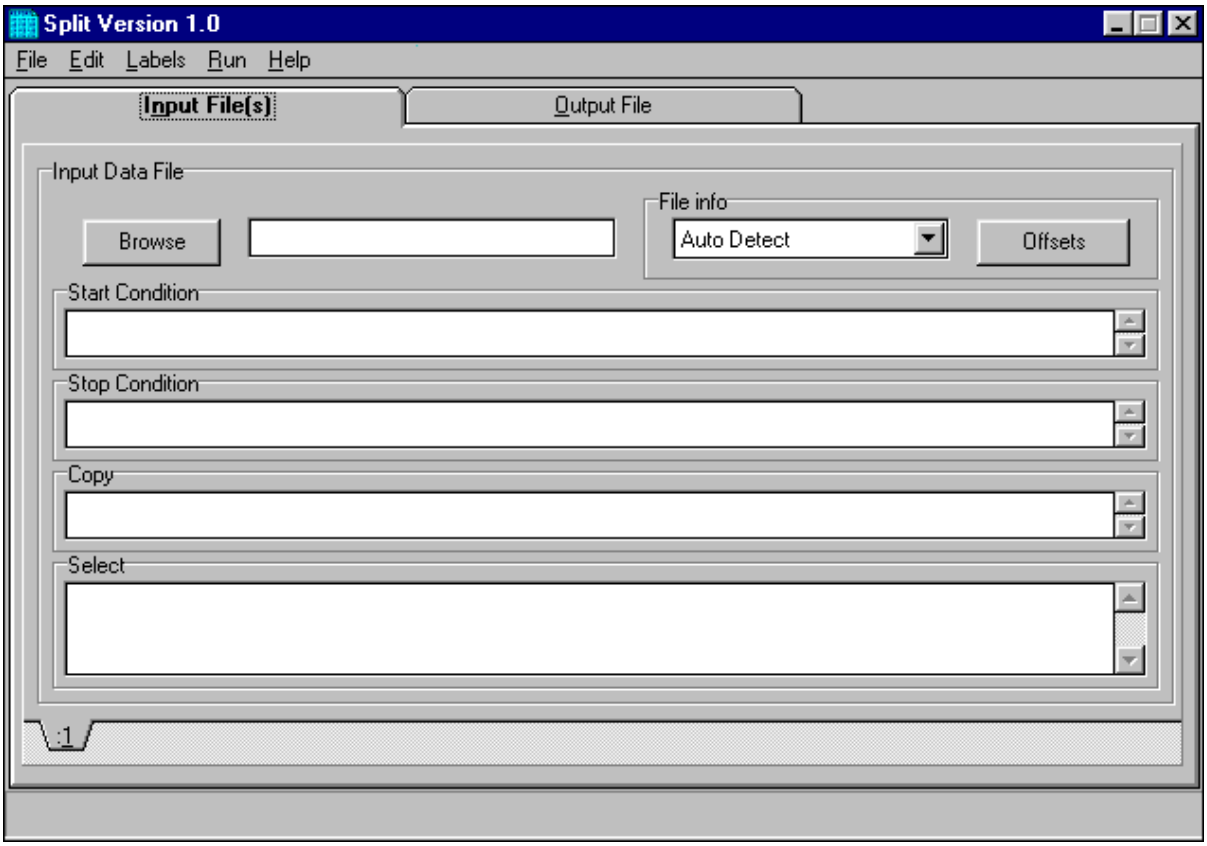


FIGURE 3.2-3. Input File Template

Start SPLIT from the PC208W toolbar by clicking on the Report icon. Figure 3.2-3 shows the blank SPLIT template. Information entered into this template can be saved as a parameter file (*.PAR) and reused for other data.

On the **INPUT FILE** tab we only need to specify the input filename, copy condition, and the data to select. SPLIT allows specific start and stop conditions to be specified but if they are left blank, the entire file will be read.

The name of the Input Data File can be typed in or the **Browse** button can be used to select from available files. In this example: C:\PC208W\WS1.dat is selected as the input data file.

Selecting the data to copy is simplified by the use of the WS1.fsl file. From the toolbar menu, click on LABELS | USE FINAL STORAGE LABELS. From the Data File Labels pop-up, Select File is used to find WS1.fsl. When one of the Output Arrays is highlighted, the Field Names of the data in that array are displayed (Figure 3.2-4).

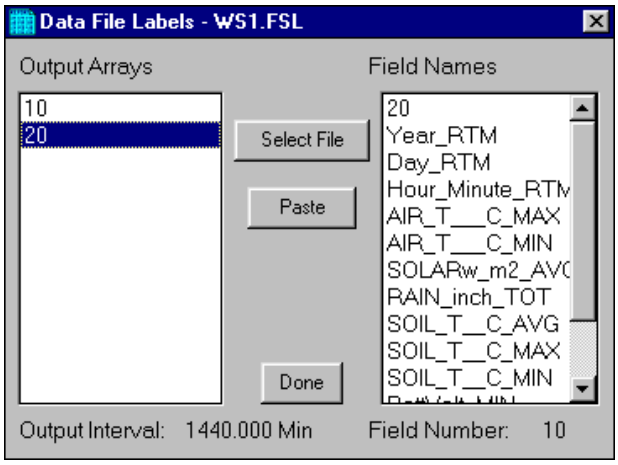


FIGURE 3.2-4. DATA FILE LABELS Screen

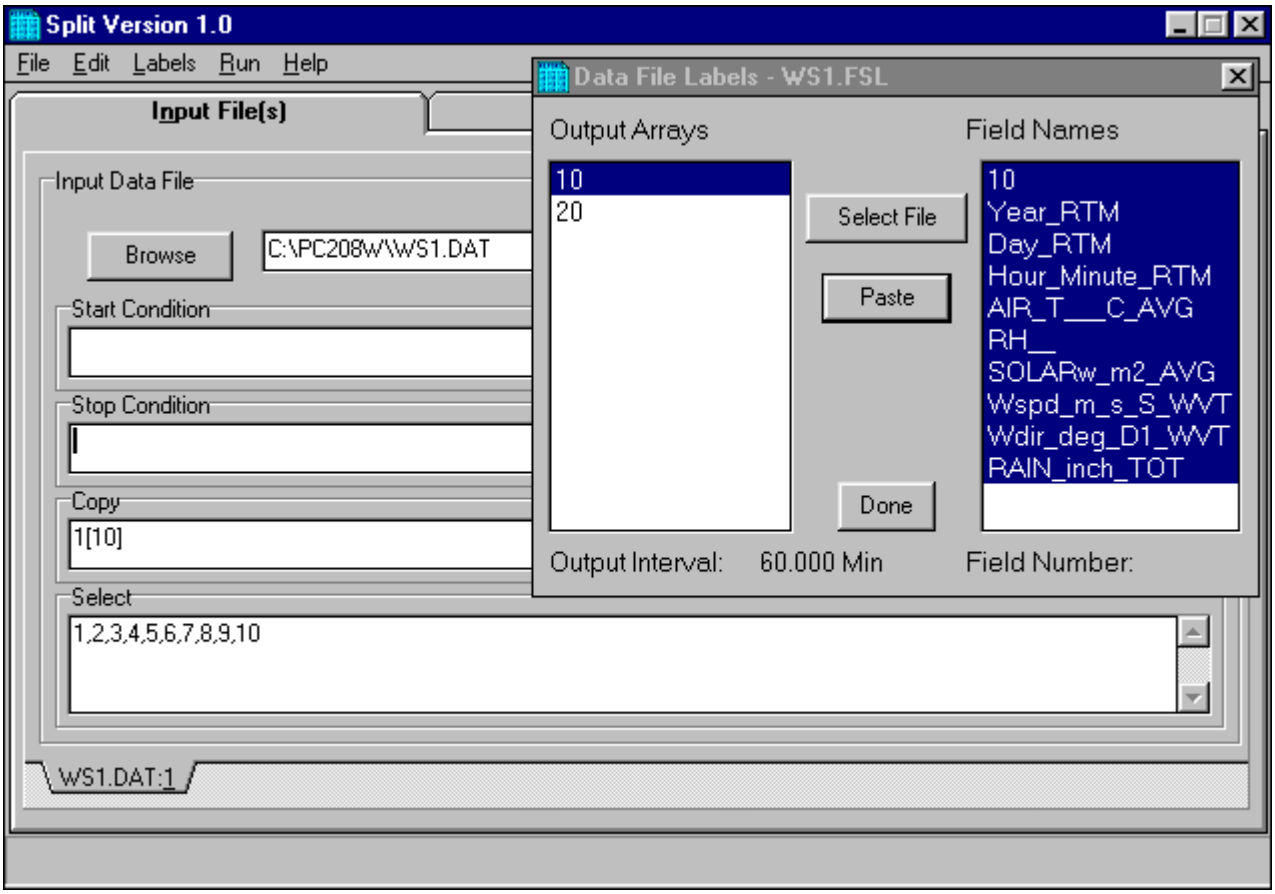


FIGURE 3.2-5. Pasting Values into SPLIT

In this example we want the hourly data (note the Output Interval at the bottom of the Data File Label window) so we click on array 10. To paste all the values in this array into the Select box, we put the SPLIT cursor in the Select box by clicking the mouse in that box, select all the field names (click on the first one and holding the mouse button down, drag to the end), then click on Paste (Figure 3.2.5).

Note that the cursor in the **INPUT FILE(S)** screen must be in valid paste area (Copy or Select). If the cursor is in the Filename box or in Start/Stop condition, you will get the error message “Cannot Paste There”.

The Paste operation put the numbers of each of the fields into the Select box. Notice also that it pasted the Array ID into the copy condition: 1[10] tells SPLIT that in order to copy a line of data, the first value in that line must be 10. SPLIT uses the Array ID to discriminate between the hourly and daily data.

Now we must specify the Output File. (Without one specified, SPLIT will run and display results but no output file will be created.) Click on the **OUTPUT FILE** tab. We type in “hourly” for the name of the output file. By default, SPLIT will use the file extension “PRN”, creating the output file: hourly.prn.

The Labels option from the toolbar can also assist in labeling the output values. Once again, choose LABELS | USE FINAL STORAGE LABELS and select array 10 and all the field names. This time move the cursor to Line 1 of the first column of labels on the **OUTPUT FILE** tab and press Paste (Figure 3.2-6).

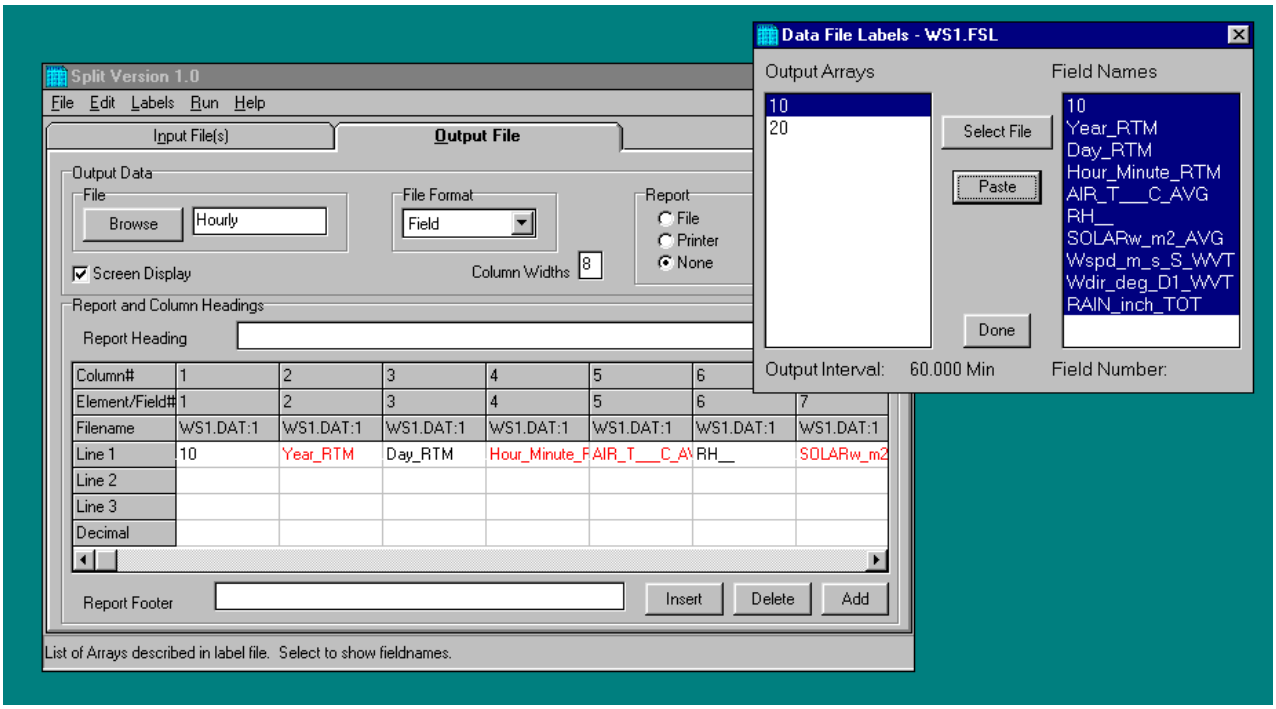


FIGURE 3.2-6. Pasting Labels Into SPLIT

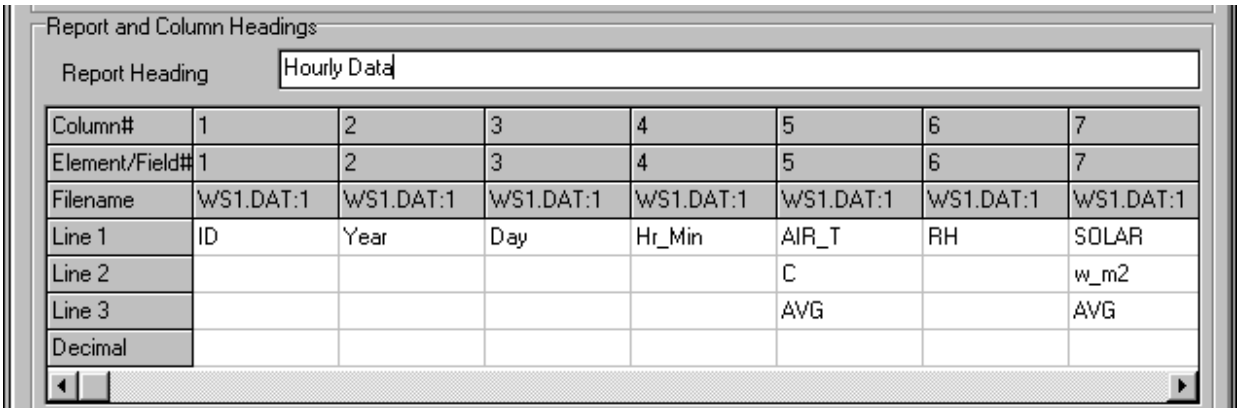


FIGURE 3.2-7. Edited Headings

Maximum column heading width is one less than the number entered in the column width field. Any FSL labels which are too long for SPLIT column headings will be shown in red. They should be edited before running SPLIT. To edit one of the labels, press the <Enter> key or use a mouse to copy, cut, and paste. A Report Heading can also be entered (Figure 3.3-7) using the same editing technique.

Now we are ready to run SPLIT. Select RUN | GO and the hourly data will be split out and stored in hourly.prn. The results are displayed on the screen as shown in Figure 3.2-8.

Close the Run window. If you wish to save this parameter file for future reports, choose FILE | SAVE. The file will be saved with a .PAR extension.

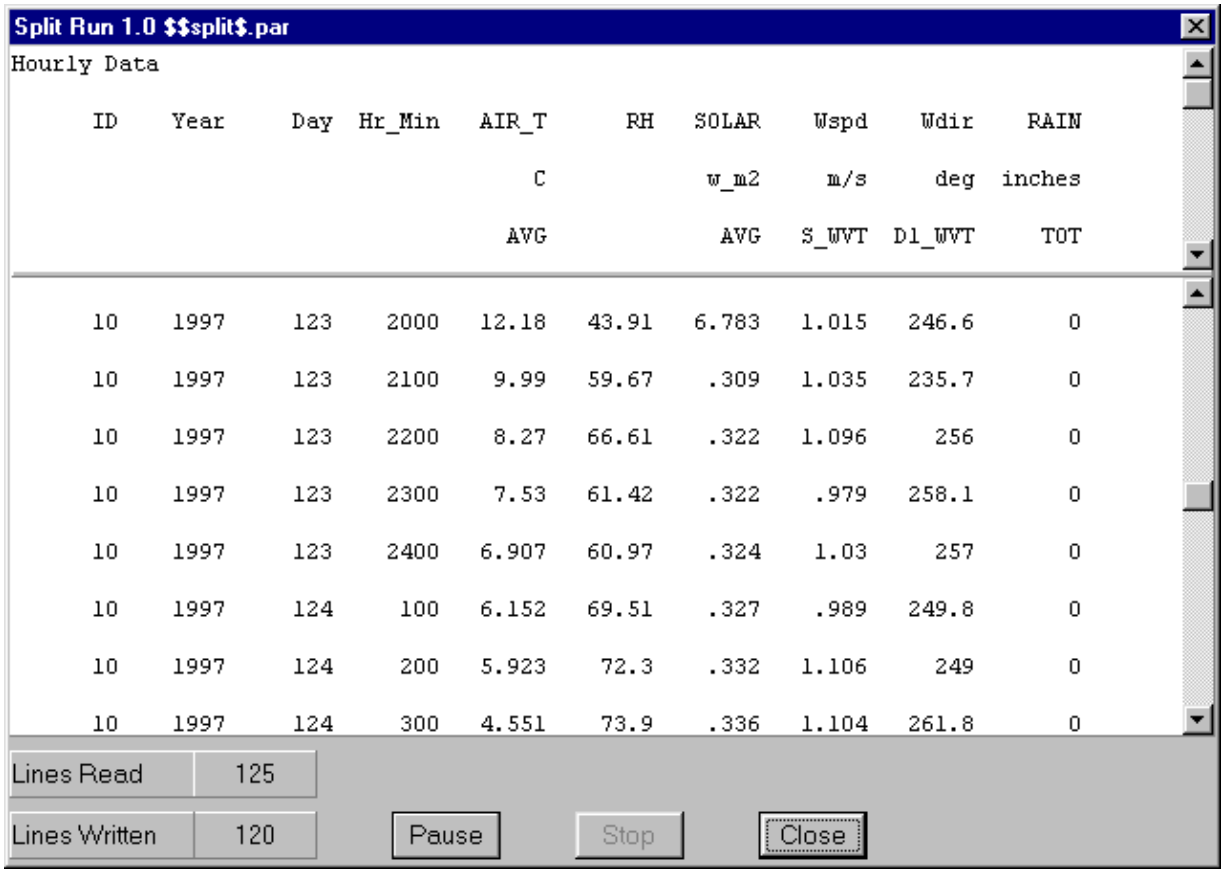


FIGURE 3.2-8. SPLIT Run Showing Hourly Data

3.3 SPLIT PARAMETER FILE ENTRIES

3.3.1 INPUT FILES

The name of the Input File is entered in the space to the right of the **Browse** button. The default directory is defined by the properties of the shortcut that started PC208W. If the input file is not in the default directory, use the **Browse** button to find the input file.

Input Files must be formatted in Comma Separated ASCII, Final Storage (Binary) Format, Field Formatted ASCII (SPLIT default output format), Printable ASCII, or Raw A/D data (refer to special Burst Mode Instruction in Campbell Scientific datalogger manual).

Table 3.3-1 provides an example of Printable ASCII, Comma Separated, and Field Formatted input file types. The data in the various formats are identical. Each line of data represents an "Output Array", starting with an Output Array ID (in this case 115). Each data point in the Output Array is referred to as an "element". The element number is given in the Printable ASCII format, and implied in the other formats. Data presented in Table 3.3-1 is used for example purposes in the following sections.

TABLE 3.3-1. Printable ASCII, Comma Separated and Field Formatted Input File Format Types

COMMA SEPARATED

115,189,1200,89.6,55.3,25.36,270
115,189,1300,91.3,61.5,27.25,255.4
115,189,1400,92.7,67.7,15.15,220.1
115,189,1500,94.1,69,20.35,260.6

FIELD FORMATTED

115	189	1200	89.6	55.3	25.36	270
115	189	1300	91.3	61.5	27.25	255.4
115	189	1400	92.7	67.7	15.15	220.1
115	189	1500	94.1	69	20.35	260.6

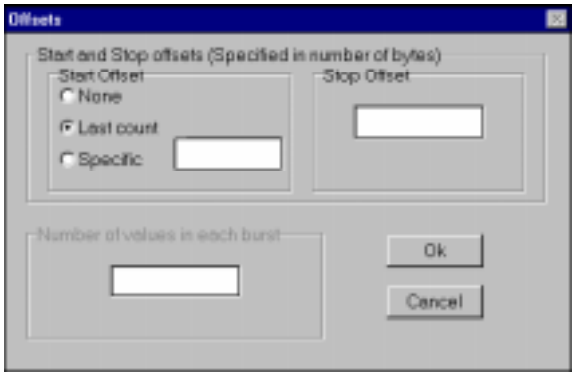
PRINTABLE ASCII

01+0115 02+0189 03+1200 04+089.6 05+055.3 06+25.36 07+270.0
01+0115 02+0189 03+1300 04+091.3 05+061.5 06+27.25 07+255.4
01+0115 02+0189 03+1400 04+092.7 05+067.7 06+15.15 07+220.1
01+0115 02+0189 03+1500 04+094.1 05+069.0 06+20.35 07+260.6

Element 1 = Output Array ID# (115)
Element 2 = Julian day (189)
Element 3 = hour, minute
Element 4 = average temperature in deg. F
Element 5 = average soil temperature in deg. F
Element 6 = average wind speed in mph
Element 7 = wind direction in degrees

A maximum of eight input files may be opened by SPLIT. Additional input files are opened with the EDIT | ADD DATA FILE menu option. SPLIT defaults to a file extension of .DAT if no extension is specified. If the Input File does not exist, an error message is displayed when RUN | GO is selected from the menu options.

For instance, to process two files named TEST.DAT and TEST_1.DAT the user would select TEST.DAT and TEST_1.DAT as Input Files. Two blank templates will be generated. To change from one template to the other, click on the appropriate tab on the bottom of the screen. Both templates must be completed before SPLIT will process the data. To merge different output arrays from the same input file into one array, open the data file once for each different array.

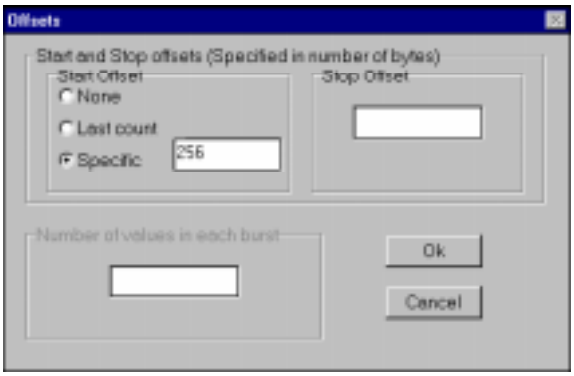


Example 1

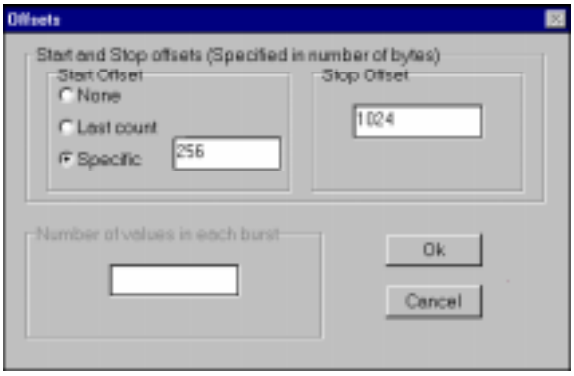
Each time SPLIT runs a parameter file, it keeps track of the number of bytes it read from the input file and saves this information in the parameter file. SPLIT can then start where it last left off. This is done by clicking on the **Offsets** button and selecting the Last Count option (see Example 1). This feature may be used to process only the new data from a file in which new data are being appended periodically to the data file using the scheduled data collection option of PC208W.

CAUTION: When using the Last Count option, if the Start and Stop Conditions (Sections 3.3.2, 3.3.3) are specified, they must exist in the newly appended data or SPLIT will never begin execution.

By selecting the Specific option and entering a number, SPLIT will "seek" that position in the file. This option saves time by starting part way through a large data file. The number specifies the number of characters into the file to seek before processing data. A positive or negative number can be entered. If the number is positive, SPLIT will start reading from the beginning of a file; if the number is negative, SPLIT will start reading from the end of a file. All characters, including spaces, carriage returns, and line feeds, are counted. Beginning and ending points may be specified.



Example 2



Example 3

In Example 2, SPLIT will skip the first 256 characters before it begins processing the data in Input File. In Example 3, SPLIT will skip the first 256 characters before beginning and stop execution on character 1024.

In most instances, SPLIT automatically recognizes the type of data file it is reading by using Auto Select in the File Info field. However, there are two exceptions:

- **Reading Raw A/D Data from Burst Measurements**

  To read this type of data and convert it to ASCII, select Burst in the File Info box. Once Burst is selected, the Number of Values in Each Burst window in the Offset Menu will become accessible. Enter the number of elements in each Burst. This number does not include the array ID number or calibration data.

- **Reading Data in Final Storage (Binary) Format**

  If Start and Stop Offsets are used in parsing the data, Final Storage (Binary) Format must be selected in the File Info field. This tells SPLIT that the file must be decoded as Final Storage before counting the bytes. If Offsets are not used, Auto Detect may be chosen and the file will be processed correctly.

This option works with all subsequent commands except with the Transpose option (Section 3.3.1) or the Select elements option (Section 3.3.5).

3.3.2 START CONDITION

A starting point may be specified to begin processing data. If the Start Condition field is left blank, SPLIT will start processing data at the beginning of the data file. The starting point can be any element within the array or a combination of elements within an array.

For example, the data in Table 3.3-1 contains seven elements per Output Array, representing hourly data. Assume that this data file contains one month of hourly data. To start processing data at 1500 hours on the first day, the Start Condition is expressed as 3[1500], where 3 means the third element within the array and 1500 is the value of that third element.

The syntax can be expressed as:

$$e_i[val_i]$$

where e_i = the position number of the element within the array

val_i = the value of that element.

Logical “and” and “or” statements can be used when specifying the Start Condition. A logical “and” statement means that all conditions must be true for the statement to be true. Up to three conditions can be connected with “and” statements. If too many “and” statements are used, an error message will be displayed when you run SPLIT.

The logical “or” statement means that if *any* of the conditions are true, then the statement is true. SPLIT allows up to six conditions to be connected with “or” statements. Additionally, each “or” statement can contain up to three “and” conditions. As with the “and” statements, if the maximum number of valid statements is exceeded, an error message will be displayed.

These rules for logical statements also apply to the Stop and Copy Conditions.

An example of a simple logical “and” statement follows:

2[189]and3[1200]

Element two (the Julian day) must equal 189, and element three (the time in hours/minutes) must equal 1200.

If the following “and” statement was used:

2[189]and3[1200]and4[92]and5[67]

an error would be returned because the maximum number of allowable “and” statements has been exceeded.

A range can be specified for val_i by putting ".." between the lower and upper limit. For example:

2[189]and7[200..275]

In this example two conditions must be satisfied to start processing data. First, the day of year must be 189, and second, the wind direction must be between 200 to 275 degrees, inclusive.

The time synchronize function is useful when data are missing from files or several files of data need to be merged together. The files are synchronized according to time. This function synchronizes according to day, hrmn (hour-minute), and/or seconds. The syntax used to identify the time elements is:

$$e_i[day]:e_i[hrmn]:e_i[seconds]$$

Referring to Table 3.3-1, to identify the day of year, type:

2[189]::

for hrmn type:

:3[1200]:

and seconds are expressed as:

::4[5]

A single colon is assumed to be between day and hrmn (e.g., 2[189]: means day, :3[1200] means hours, and 2[189]:3[1200] means day and hour-minute). When the time synchronize function is used, a time interval must be specified in the Copy line. If no time interval is specified in the Copy line then the time specified in the Start Condition becomes simply a starting time with no time synchronization. The starting time specified must actually be found in the input file before the Start Condition is satisfied (e.g., if the input file starts at 1100 hrs and 1000 hrs is entered for the starting time, with no day specified, SPLIT will skip over arrays until it reaches 1000 hrs the next day).

Time elements can be identified without specifying a starting time (e.g., 2:3). When no starting time is specified, SPLIT assumes the starting time to be 0 and inserts carriage return line feeds (CRLF) at the start of the Output File. The number of CRLFs equals the time between 0 and the time of the first array selected, divided by the interval specified in the Copy condition.

When time synchronizing, if multiple input files are given starting times, SPLIT starts the Output at the earliest specified starting time. Blanks or the comment entered in the “Replace bad data with” field are inserted for values from

other input files until their starting times are reached.

Included in the time synchronize function is the ability to start relative to the current PC TIME (computer time). This feature allows a .PAR file to be run on different data without changing the Start Conditions, provided the Input Data File is collected at a fixed frequency and SPLIT is run at a fixed frequency. For example, assume that PC208W is used in scheduled data collection mode which automatically appends data to an archive file. SplitR is executed using the “After Call do” option in PC208W. In this case the frequency of data collection and data reduction is the same. Time values in the data file (day, hrmn, sec.) are different each time the data are collected, but by telling SPLIT where to Start reading relative to the PC clock, the Start Conditions do not need to be changed. To accommodate variations in the data collection and reduction frequencies, an interval in minutes or seconds may be specified as shown in the examples below.

:3[-60,5] tells SPLIT to start at the closest 5 minute interval that is less than the PC time minus 60 minutes. If the PC time is 1404, SPLIT calculates 1304 and looks for hour 1300 to start reading.

2[-3]:3[-120,60] tells SPLIT to find the closest 60 minute interval that is less than the PC time minus 3 days and 2 hours. If the PC time is the day of year 159, hour 0017, SPLIT will start reading on data output at 2200 hours on day 155.

2[-3]:3[-120]:4[20,5] tells SPLIT to find the closest 5 second interval that is less than the PC time minus 3 days, 2 hours and 20 seconds. If the PC time is 27 seconds after noon on day 30, SPLIT will begin reading on data output at 1000 hours and 05 seconds on day 27.

> **CAUTION:** SPLIT will not start reading if the exact specified starting time cannot be found. The interval (5 minutes, 60 minutes, and 5 seconds in the examples above) must be evenly divisible into 60 minutes.

3.3.3 STOP CONDITION

The Stop Condition is expressed with the same syntax as the Start Condition, with the exception that there is no time synchronization. If the Stop Condition parameter is left blank, SPLIT will execute until the end of the file. Logical "and" and "or" statements can be used when specifying the Stop Condition (Section 3.3.2).

The Stop Condition specifies when to stop processing data. This feature allows segments of data to be removed from large data files. For instance, if a data file contains one month of data and just one day is desired, the start and stop values allow the user to get just that day’s data. The array containing the Stop Condition is not included in the output file. If the stop value is not found, SPLIT will display a dialog box that gives the option to select a new file and continue processing the data. This feature is useful when data are contained on several diskettes.

The C and F commands alter the meaning of the Stop Condition.

3.3.3.1 "C" Option: Formatting Event Tests Containing Conditional Output Arrays

The C option is used to combine data from two or more conditional arrays on to one Split output line. A conditional array is one that is only output when a defined event occurs.

Assume that two or more conditional Output Arrays with unique Output Array IDs compose a test period, followed by an unconditional Output Array that defines the end of a test. The unconditional "end of test" Output Array is at the end of each test, but the conditional Output Arrays may or may not be present. The data file is comprised of several of these tests.

As an example, let's look at a vehicle test application. The start of the test is when the vehicle is turned on, and the end of the test is when the vehicle is turned off. The conditional output arrays could be:

- monitoring the engine temperature and outputting data to a unique array when the temperature exceeds a limit
- outputting data to a unique array when the brakes are applied
- outputting data when engine RPM exceeds a limit

The unconditional array data (the stop condition) would be output to a unique array when the engine is turned off. By processing the data with Split using the C option, the data collected during each test could be merged on to one line, with blanks inserted if a set of data didn't exist (e.g., if the engine temperature never exceeded the defined limit).

To use the C option, the Parameter file should be set up as follows:

- An Input File must be set up for each array ID in the test. The first Input File is configured on the Input File tab that appears when you open Split. Additional Input Files are added by choosing Edit | Add Data File from the Split menu. The same data file will be used as the Input File for each array.

- Type in the array ID in the Copy field of the Input File tab for each array. The array ID is the first element of a data file, so the line should read 1[*123*], where *123* is the actual array ID you want to process.

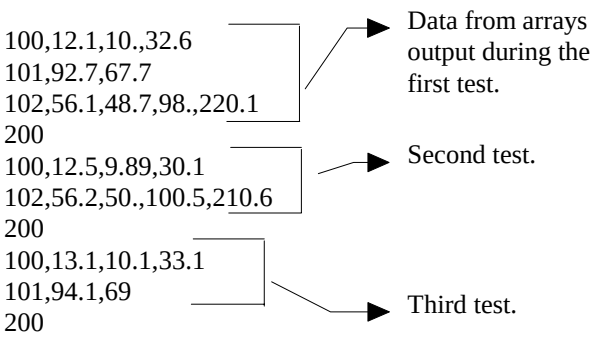
- In the Select field, type in the number for each element (data value) you want to be output in the report.

- In the Stop Condition field, type in a "C," followed by the ID of your stop condition array. If your "end of test" array was array ID 200, the Stop Condition field would read: C,1[200]. This should be typed into the Stop Condition fields of each array, including the "end of test" array.

Set up the Output File as you would for any Split process. If you are including column headings, the arrays and elements will appear in the order they are listed on the Input File tabs. That is, the first column will be Input File number 1, element number 1; the next column is Input File number 1, element number 2… Input File number 2, element number 1 follows in the column immediately after the last element of Input File number 1.

Consider Table 3.3-2 below:

TABLE 3.3-2. Example of Event Driven Test Data Set



This table contains four different output arrays: 100, 101, 102, and 200. During the first test, data was output from all three conditional arrays (100, 101, and 102), with 200 signaling the end of the test. During the second test, data was output from arrays 100 and 102. During the third test, data was output from arrays 100 and 101.

To process these files using the C option, the parameter file would be set up as follows (assuming the name of our data file is Data_1.DAT):

First Input File = Data_1.DAT:1
Stop condition = C,1[200]
Copy = 1[100]
Select = 1,2,3,4

Second Input File = Data_1.DAT:2
Stop condition = C,1[200]
Copy = 1[101]
Select = 1,2,3

Third Input File = Data_1.DAT:3
Stop condition = C,1[200]
Copy = 1[102]
Select = 1,2,3,4,5

Fourth ("end of test") Input File = Data_1.DAT:4
Stop condition = C,1[200]
Copy = 1[200]
Select = (leave blank)

NOTE: The *:(number)* after the data file name is inserted automatically by Split.

TABLE 3.3-3. Processed Data File Using Option C

100	12.1	10	32.6	101	92.7	67.7	102	56.1	48.7	98	220.1
100	12.5	9.89	30.1				102	56.2	50	100.5	210.6
100	13.1	10.1	33.1	101	94.1	69					

TABLE 3.3-4. Processed Data File

100	12.1	10	32.6	101	92.7	67.7	102	56.1	48.7	98	220.1
100	12.5	9.89	30.1				102	56.2	50	100.5	210.6
100	13.1	10.1	33.1	101	94.1	69					

When Split is run, the resulting data file will look similar to Table 3.3-3. Each line of data represents one test. Notice that blanks were inserted if the data set (conditional array) did not exist.

3.3.3.2 Trigger on Stop Condition (F Option) Output of Time Series

The Trigger on Stop Condition, or F option, changes the function of the Stop Condition when one or more Time Series functions (Section 3.3.5.2) are contained in the Select field. When a Stop Condition is met, the time series data is calculated and written to the output file. However, instead of stopping at this point, processing resumes and time series data is output the next time the Stop Condition is met. This continues until the end of file or until the user stops Split manually.

The Trigger on Stop Condition is enabled by clicking **Other…** on the Output Tab and checking the box next to the Trigger on Stop Condition field. When the Trigger on Stop Condition is enabled, the function affects all files being processed that have a Stop Condition specified. If multiple files are being processed but it is desired that the function affect one or more—but not all—of the files, the F option is used in the Stop Condition field of the files that you want processed using the function. The syntax for the F option is: $F,e_i[val_i]$.

A typical application for the Trigger on Stop Condition is to reduce days of hourly data into daily summaries. A logical element to use for the Stop Condition is time (hrmn). Assuming the third element of the hourly Output Array is hrmn, and midnight is output as 0, the Stop Condition is entered as 3[0] (or F,3[0] if the F option is used). The Time Series processing is performed over a day defined by midnight to midnight.

If only hourly Output Arrays were contained in the Input File, the Copy line could be left blank. If other Output Arrays are present which need not be included in the Time Series processing, a logical Copy condition would be the Output Array ID of the hourly output.

The Trigger on Stop Condition functions the same for multiple Input files as it does for a single Input File. If the option is enabled on several Input Files, and the Stop Conditions do not occur at the same point in each file, when a file's Stop Condition is met, its time series data are output and blanks are output for data selected from the other Input Files.

Say, for example, that you were interested in the average value of the first data point (element 2) for each test, in the data set listed in Table 3.3-4. The Input File template would look like that shown in Table 3.3-5.

TABLE 3.3-5. Input File Entries to Process the First Data Point for each Test

First Input File = DATA_1.DAT:1
Stop Condition = F,1[200]
Select = AVG(2)

3.3.4 COPY

After the Start Condition is satisfied, and before the Stop Condition is met, the Copy condition must be satisfied before any data will be processed according to Select line instructions (Section 3.3.1.4). If the Copy condition is left blank, all arrays are processed between the Start and Stop values. Syntax for the Copy condition is similar to the Start and Stop values mentioned above. Logical "and" and "or" statements (see Section 3.3.2) can be used when specifying the Copy condition.

For example, referring to Table 3.3-1, if only those hours during day 189 when the temperature was above 90 and the soil temperature was below 62 is desired, or, during day 189 when the average wind speed was below 21 while the wind direction was between 255 to 265 is desired, the Copy condition would be:

1[189]and4[90..150]and5[0..61.99]or1[189]and6[0..20.99]and[255..265]

Only Output Arrays with hours 1300 and 1500, Table 3.3-1, conform to the above Copy conditions.

To use the time synchronize function, time element(s) must be specified in the Start Condition (Section 3.3.1.1). The user specifies a time interval in the Copy condition. For instance, if the original data had 15 minute outputs and you only want hourly outputs, then an interval of 60 minutes must be specified following the element number. This is entered as (assuming hrmn is element number 3) “3[60]”. If time synchronization is specified in the Start Condition, SPLIT looks for the interval in a time element in the Copy condition. Only one time interval is specified. This interval is the unit of time to synchronize each file.

The interval can be given tolerance limits by following the interval with a comma and the tolerance. For example, if 3 is the hrmn element, and the time interval is 60 minutes +2 minutes, the syntax is 3[60,2].

If the time synchronize function is enabled, and data are missing at one or more of the time intervals specified, then a blank (or the comment entered in the “Replace bad data with” field) is output to the Output File.

3.3.5 SELECT

The Select line specifies which elements of an Output Array are selected for processing and/or output to the specified Output File. The Select line becomes operable only after the Start Condition and Copy condition are met, and before the Stop Condition is satisfied. If the Select line is left blank, all output arrays meeting the Start Condition and Copy conditions are output to the Output File. Up to 255 characters can be entered on one Select line. If this is a limitation, open the Input Data file twice (use the EDIT | ADD DATA FILE menu) and use the Select line in the second Input File template to define the additional operations.

TABLE 3.3-6. Effects of Out of Range Values for Given Output Options

Output Option	Screen Display*	PRN File	RPT File or Printer Output
Report = None; No other options defined (default)	bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	blanks inserted for bad values	N/A
Report = File or Printer; no other options defined	bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	blanks inserted for bad values	bad values preceded by asterisk
Report = None; comment in "Replace bad data with" field	bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	comment inserted in place of bad values	N/A
Report = File or Printer; comment in "Replace bad data with" field	bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	comment inserted in place of bad values	bad values preceded by asterisk
Report = None; "Display only bad data" option enabled	only lines with bad data are displayed; bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	only lines with bad data output; blanks inserted for bad values	N/A
Report = File or Printer; "Display only bad data" option enabled	only lines with bad data are displayed; bad values displayed in red and preceded by asterisk; the text "bad data" highlighted in a red box at bottom right of screen	only lines with bad data output; blanks inserted for bad values	only lines with bad data output; bad values preceded by asterisk

*The Screen Display box must be checked; if not, no data will be displayed on the Split Run screen.

Element numbers may be entered individually (e.g., 2,3,4,5,6,7), or, in groups (e.g., 2..7) if sequential. Range limits (lower to upper boundary conditions) may be placed on elements or groups of elements specified in the Select or Copy lines. For example, 3[3.7..5],4..7[5..10] implies that element 3 is selected only if it is between 3.7 and 5, inclusive, and elements 4,5,6, and 7 must be between 5 and 10, inclusive.

If range limits are used in the Select condition, when SPLIT is run, any data which are outside of the specified range will be highlighted according to the options chosen for the output file. Table 3.3-6 summarizes what each option produces on the screen and in the output file if out of range data are encountered. This type of range testing is a quick way to identify data problems.

> **NOTE:** In this instance, out of range data refers to data outside of the specified output range. It is not to be confused with out of range data generated by the logger.

Variables can be assigned names in the Select line. For example, x = 4-5∗(6∗3.0) means that x is equal to element 6, times the number 3, times

element 5, subtracted from element 4. A numeric value is distinguished from an array element by the inclusion of a decimal point. Variables must be declared before they can be used in the Select line. A variable name must start with an alpha character, can include numbers and must not exceed eight characters. Variable names can start with the same character but they must not start with another complete variable name (e.g., the variable XY is not valid if there is also the variable X). A comma must follow each variable statement, as with all parameters in the Select line. Once the variables have been declared they can be used later in the Select line (i.e., x=4-5∗(6∗3.0), y=6/3,2,3,6,7,7∗x,6+y). Variables can be defined in the first four Input File's Select lines only, but may be used in subsequent Input File's Select lines.

Processing is accomplished through arithmetic operators, math functions, spatial functions, and time series functions. The following is a list of operators, math functions, and Time Series functions supported by SPLIT.

TABLE 3.3-7. SPLIT Operators, Math Functions, and Time Series Functions

OPERATORS		OPERATOR PRECEDENCE ORDER (3 = high, 1 = low)
^	= raise to the power	3
x Mod y	= Modulo divide of x by y	2
∗ /	= multiplication, division	2
+ -	= addition, subtraction	1

EXAMPLES OF SYNTAX FOR MATHEMATICAL OPERATORS

3∗5	multiply element 3 by element 5
3/5	divide element 3 by element 5
(3..5)/(8..10)	same as 3/8, 4/9, 5/10
3+5	add element 3 to element 5
3-5	subtract element 5 from element 3
(3,9,5)-(8,7,10)	same as 3-8, 9-7, 5-10
3∗2.0	multiply element 3 by a fixed number 2
2^3.0	raise element 2 to the third power

MATH FUNCTIONS

Abs(x)	= Absolute value of x
Arctan(x)	= Arc tangent of x (in degrees)
Cos(x)	= Cosine of x (in degrees)
Exp(x)	= Natural Exponent function (e^x)
Frac(x)	= Fractional portion of x
Int(x)	= Integer portion of x
Ln(x)	= Natural logarithm of x
Sin(x)	= Sine of x (in degrees)
SpaAvg(x..y)	= Spatial average of elements x through y
SpaMax(x..y)	= Spatial maximum of elements x through y
SpaMin(x..y)	= Spatial minimum of elements x through y
SpaSd(x..y)	= Spatial standard deviation of elements x through y
Sqrt(x)	= Square root of x

TIME SERIES FUNCTIONS

Avg(x;n)	= Average
Blanks(x;n)	= Number of blanks in element
Count(x;n)	= Number of data points in element
Max(x;n)	= Maximum
Min(x;n)	= Minimum
RunTotal(x;n)	= Running total
Sd(x;n)	= Standard deviation
Smpl(x;n)	= Sample raw value

SmplMax(x;y;n)	= Sample (y) on a maximum (x)
SmplMin(x;y;n)	= Sample (y) on a minimum (x)
Total(x;n)	= Totalize
WAvg(x;n)	= Unit vector mean wind direction (in degrees)

NOTE: x can be an element or a valid expression. n is optional and is the number of arrays to include in the function. Date and Edate can be used for the "n" in the Time Series functions (see Table 3.3-8 Special Functions, this Section)

Illegal operations (e.g., logarithm of a negative number) will cause SPLIT to store blanks for the Output. It is possible to get a run time error (error 0/1) if the floating point math exceeds the limits of the PC.

The greatest number that can be output is determined by the field width (Table 3.3-3, "Column Widths" option). If the width is eight or greater, the maximum number is 99999; for widths less than eight the maximum is 9999. Larger numbers will be stored as 99999 or 9999.

TABLE 3.3-8. SPLIT SPECIAL FUNCTIONS

CrIf	=	Insert carriage return line feed in Output File.
Date(doy;y)	=	Convert day of year to calendar date, where doy=day of year and y=year. Format is MM DD. When used as the interval, n, with Time Series Functions (see example, this Section), Date triggers monthly time series summaries.
Edate(doy;y)	=	Convert day of year to European date, where doy=day of year and y=year. Format is DD MM. When used as the interval, n, with Time Series Functions (see example, this Section), Edate triggers monthly time series summaries.
"Label"	=	Insert Comment in Output file. (Label is anything within the quote marks.)
Line	=	Number of lines written to Output file.
.PA	=	Page break to printer or to the .RPT file.

3.3.5.1 Mathematical Functions, Details, and Examples

The following array of PRINTABLE ASCII data will be used for all Mathematical function examples.

0105 0176 1200 -07.89 55.10 12.45 270.5

Abs(x) returns the absolute, or positive value of element x.
Examples:
Abs(4) = 7.89
Abs(4∗5) = 434.74

Arctan(x) returns the arc tangent of element x in degrees.
Examples:
Arctan(7) = 89.788
Arctan(7/6) = 87.365

Cos(x) returns the cosine of element x in degrees.
Examples:
Cos(5) = .57215
Cos(5-6) = .73551

Exp(x) returns the exponential base e to the power of element x.
Example:
Exp(4) = .00037

Frac(x) returns the fractional value of the element x.
Examples:
Frac(4) = -.89
Frac(6+7) = .95

Int(x) returns the integer portion of the element x.
Examples:
Int(7) = 270
Int(5∗6) = 685

Ln(x) returns the natural log of element x.
Examples:
Ln(6) = 2.5217
Ln(7/6*5/1) = 2.4337

Sin(x) returns the sine of element x in degrees.
Examples:
Sin(7) = -.99996
Sin(7-2+5) = .50603

Spatial functions, included under Mathematical functions, operate on a per Output Array basis. The average, maximum, minimum, and standard deviation of a specified group of elements within an array are calculated.

SpaAvg(x..y) returns the spatial average of elements x through y.
Examples:
SpaAvg(1..7) = 258.74
SpaAvg(1,4,7) = 122.54

SpaMax(x..y) returns the maximum value of elements x through y.
Examples:
SpaMax(1..7) = 1200
SpaMax(1,2,5) = 176

SpaMin(x..y) returns the minimum value of elements x through y.
Examples:
SpaMin(1..7) = -7.89
SpaMin(1,2,5) = 55.1

SpaSd(x..y) returns the standard deviation of elements x through y.
Examples:
SpaSd(1..7) = 394.57
SpaSd(5,2,1) = 49.607

Sqrt(x) returns the square root of element x.
Examples:
Sqrt(3) = 34.641
Sqrt(3^ 2.0) = 1200

3.3.5.2 Time Series Functions, Details, and Examples

Time Series functions are used to perform vertical processing on selected elements, such as calculating the average of an element over a specified range of data. Time Series results are output in three instances:

1. when a Trigger on Stop Condition (F option) is met

2. at the end of a data file (or within a range specified by Start and Stop Conditions)

3. when an interval count is met

When the Trigger on Stop Condition (or F option) is used, any time series data defined in the Select line is output each time the Stop Condition is met. Refer to Section 3.3.3.2 for more information on the Trigger on Stop Condition.

Results which are output at the end of a file or a range of data are referred to as Final Summaries. A typical select line that would produce a Final Summary is:

1,2,3,4,Avg(4)

This line would output values for elements 1 through 4 each time an array was output. Additionally, an average value for element 4 would be calculated for the entire file and output as the last line of data in the output file.

Results which are output at the end of a specified Interval are Interval Summaries. A typical select line that would produce an Interval Summary is:

1,2,3,4,Avg(4;24)

This line would output values for elements 1 through 4 each time an array was output, and an average value for element 4 would be calculated every 24th array and output as an additional column in the file. An additional summary would occur for an Interval Count if the count was not evenly divisible into the number of output arrays present in the Input File. The summary, in this case, is calculated from an incomplete interval count.

The interval count in a Time Series Function is optional and does not require a decimal point. To determine the interval, SPLIT counts the number of arrays which meet the specified conditions (Stop, Start, and Copy). If the time synchronize function is enabled, the Time Series functions remain synchronized to the

starting time even if a complete array is missing from the input data. When elements are missing, the Time Series calculations are based on the actual number of elements found.

Semicolons are used in Time Series functions to separate the elements or expressions from the count which determines the interval. SmplMax and SmplMin require two elements separated by a semicolon. The first is checked for a maximum or minimum, while the second is sampled on the maximum or minimum.

The following set of weather data from Mt. Logan Peak in Cache Valley, Utah gives a total of seven elements each hour. This Field Delineated output, with title and column headers, was generated by SPLIT. These data are used in the following examples of Time Series functions.

Mt. Logan Weather Data

Day	Time	Air temp deg F	RH	Mean Wind Speed mph	Mean Wind Vec Dir	Std Dev of Dir
178	100	58.56	17.42	5.855	338.3	6.562
178	200	57.48	17.65	8.27	344.8	7.51
178	300	56.85	17.76	7.75	330.8	5.065
178	400	56.55	18.89	7.6	319.7	10.93
178	500	56.57	19.6	10.41	307.3	4.23
178	600	55.33	23.32	8.99	317.7	6.258
178	700	55.95	24.79	9.52	322.3	4.609
178	800	58.12	23.98	6.588	315.6	9.43
178	900	59.79	23.46	5.458	312	15.32
178	1000	61.09	24.12	4.622	299.3	18.3
178	1100	61.34	25.03	5.926	303	17.26
178	1200	60.61	27.46	6.815	309.7	18.71
178	1300	61.01	25.44	8.35	310.2	18.37
178	1400	60.93	25.48	10.92	317.5	12.68
178	1500	62.3	23.79	8.43	310.6	19.21
178	1600	63.75	24.31	8.88	321.4	15.22
178	1700	66.15	22.45	7.97	341	17.77
178	1800	67.33	23.06	6.758	344.1	20.74
178	1900	66.59	24.75	7.08	341.8	16.09
178	2000	64.52	26.03	8.76	337.2	14.91
178	2100	59.84	27.45	11.81	305.4	12.36
178	2200	56.19	35.46	15.62	316.7	19.01
178	2300	55.48	38.8	17.12	338.7	11.41
179	0	55.22	37.13	11.86	351.6	8.22

Avg(x;n) returns the average of element x over a full data set or every n^{th} value.
Examples:
Avg(3) = 59.898 (average daily temp)
Avg(3;4) = 57.36 (average 4 hour temp)
56.493 (average 4 hour temp)
60.708 (average 4 hour temp)
61.998 (average 4 hour temp)
66.148 (average 4 hour temp)
56.683 (average 4 hour temp)

Blanks(x;n) returns the number of blanks or bad data in element x over a full data set or every nth value. Refer to Table 3.3-9 for definition of blank or bad data. Example:
Blanks(3) = 0 (no holes in data set).

Count(x;n) returns the number of data points (non blanks) in element x over a full data set or every n^{th} value.
Example:
Count(1) = 24 (24 data points in data set).

NOTE: Blanks and Count are functions designed for checking the integrity of the data file. A common use for these two functions is "100.*BLANK(x;n)/BLANK(x;n)+ COUNT(x;n)" which gives the percentage of holes (bad data) in the file.

Max(x;n) returns the maximum value of element x over a full data set or every n^{th} value.
Examples:
Max(5) = 17.12 (max WS for day)
Max(5;12) = 10.41 (max WS for 12 hours)
17.12 (max WS for 12 hours)

Min(x;n) returns the minimum value of element x over a full data set or every n^{th} value.
Examples:
Min(7) = 4.23 (min std. dev. of WS for day)
Min(3;8) = 55.33 (min temp for 8 hours)
59.79 (min temp for 8 hours)
55.22 (min temp for 8 hours)

RunTotal(x;n) returns a running total of element x for every line in the data set. If an n^{th} value is specified, a running total will be output every n^{th} value.
Example: RunTotal(5) =
5.85
14.12
21.87
29.47
39.88
48.87
:
:
:
166.76
182.38
199.50
211.36
211.36

Running total of hourly average wind speed provides up-to-the-hour wind run for that day. Because an n^{th} value was not specified, the Final Summary output, which is daily wind, is the same as the "total" output.

Sd(x;n) returns the standard deviation of element x over a full data set or every n^{th} value.
Examples:
Sd(3) = 3.6593 (std. dev. temp for day)
Sd(3;8) = 1.011 (Sd temp for 8 hours)
1.1182 (Sd temp for 8 hours)
4.965 (Sd temp for 8 hours)

Smpl(x;n) returns a sample of element x every n^{th} value. The Sample function must always have an interval specified.
Examples:
Smpl(4;8) = 23.98 (RH every 8 hours)
24.31 (RH every 8 hours)
37.13 (RH every 8 hours)

SmplMax(x;(y);n) looks for a maximum value in element x and samples element y when the maximum is found. If an n^{th} value is specified then it outputs the sample on a maximum every n^{th} value, otherwise it outputs the sample on a maximum at the EOF.
Examples: SmplMax(5;(3)) = 55.48 (on max wind speed sample temperature)
SmplMax(5;(3,6);8) = 56.57 307.3
60.93 317.5
55.48 338.7
(on max wind speed sample temperature and wind direction every 8 hours)

SmplMin(x;(y);n) looks for a minimum value in element x and samples element y when the minimum is found. If an n^{th} value is specified then it outputs the sample on a minimum every n^{th} value, otherwise it outputs the sample on a minimum at the EOF. Examples:
SmplMin(3;(5)) = 11.86 (on min temp sample wind speed)
SmplMin(3; (5,6);8) = 8.99 317.7
5.458 312
11.86 351.6
(on min temperature sample wind speed and wind direction every 8 hours)

Total(x;n) returns the total of element x over a data set or every n^{th} value.
Examples: Total(5) = 211.36 (daily wind run)

WAvg(x;n) Returns the unit vector mean wind direction in degrees of element x (wind direction in degrees) over a full data set or every nth value.

Example:
WAvg(6) =
323.14 (mean wind direction for the day)
WAvg(6;4) =
333.41 (mean wind direction for 4 hours)
315.73 (mean wind direction for 4 hours)
306 (mean wind direction for 4 hours)
314.92 (mean wind direction for 4 hours)
341.03 (mean wind direction for 4 hours)
328.09 (mean wind direction for 4 hours)

3.3.5.3 Special Functions, Details, and Examples

The Mt. Logan Peak data set is used for the Special Function examples. These functions are helpful in converting day of year to months and days of month and formatting the output.

Crlf returns a carriage return and line feed where the Crlf is placed in the parameter file.
Examples:
Smpl("Max Temp";24),Max(3;24),
Smpl(Crlf;24),Smpl("Max RH";24),Max(4;24)
= Max Temp 67.33
Max RH 38.8

The Crlf is placed after the maximum temperature 67.33 so that the maximum RH is on the next line.

Date(doy;y) converts the day of year into a calendar date in the form MM DD; where doy = element with the day of year and y = containing year. If the year is not contained in the data set, a year can be specified in the “y” year location (i.e., 1987.). A decimal must be included with the year. Date can be used as the interval in a Time Series function to trigger a Time Series output on the first day of each month. For example, Avg(3;Date(1;1987.)) outputs the average temperature for each month. This option is useful when the data set contains several months of data and monthly information is needed.

Examples:

Date(1;1987.) =
6 27 (June 27 repeated
6 27 23 times)
6 27
6 28

Smpl(Date(1;1987.);8) =
6 27 (June 27)
6 27 (June 27)
6 28 (June 28)

Edate(doy;y) same as Date(doy;y), except form is DD MM.

"Label" returns a comment in the output file. This is a useful formatting function when labels are desired on the same line as the data. The label includes anything within the quote marks, the quote marks are not output but must be in the parameter file. The label cannot exceed the width of the output column (default is eight characters). A maximum of thirty (30) labels are allowed per Select line.

Examples:

"Max Temp" =
Max Temp (outputs Max Temp
Max Temp 24 times)
.
.
.
Max Temp

Smpl("8 hour ";8),Smpl("Max Temp";8), Max(3;8) = 8 hour Max Temp 58.56
8 hour Max Temp 63.75
8 hour Max Temp 67.33

This example samples the labels called "8 hour" and "Max Temp" and looks for a Maximum temp for every 8 hour interval.

Line returns the number of lines written to the output file created by SPLIT. This differs from the Count function in that Count looks at how many lines were read.

Examples:

Line, 4, 5 =

1	17.42	5.855
2	17.65	8.27
3	17.76	7.75
4	18.89	7.6
5	19.6	10.41
6	23.32	8.99
7	24.79	9.52
	.	.
	.	.
	.	.
19	24.75	7.08
20	26.03	8.76
21	27.45	11.81
22	35.46	15.62
23	38.8	17.12
24	37.13	11.86

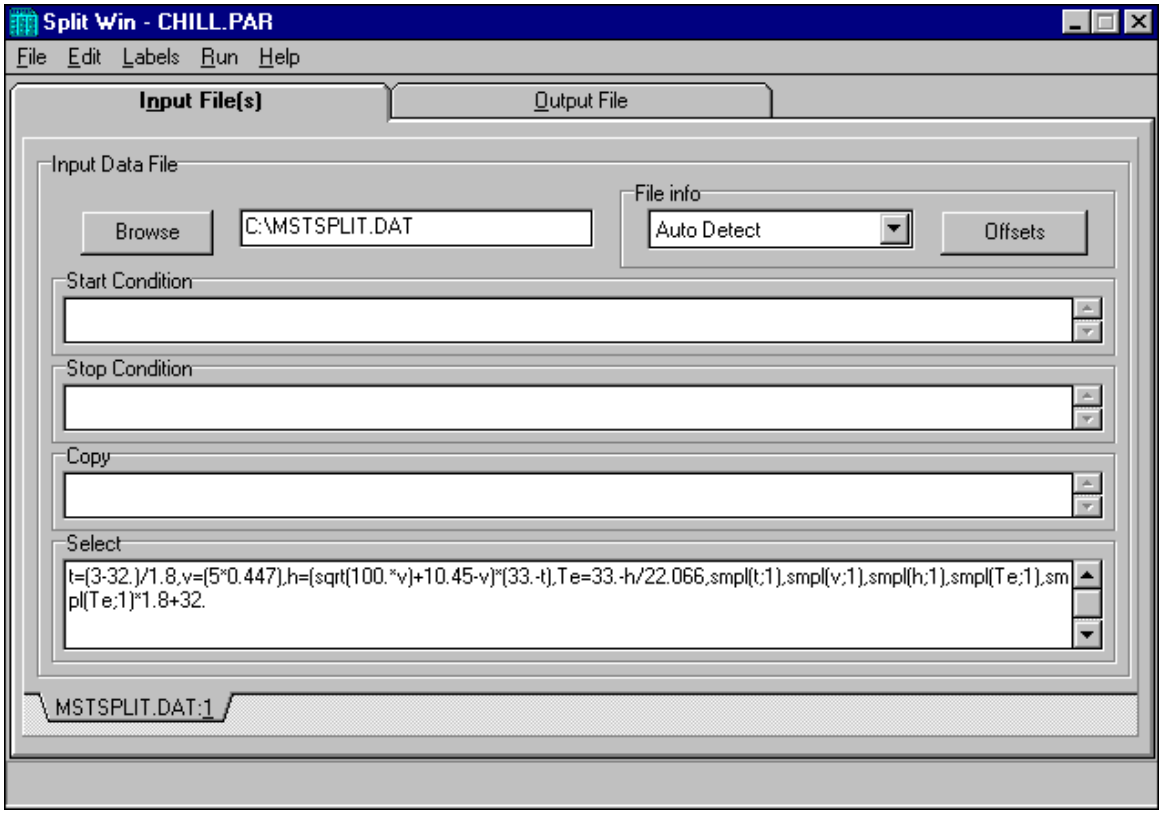
Smpl (Line;8), Smpl (4;8), Smpl (5;8)

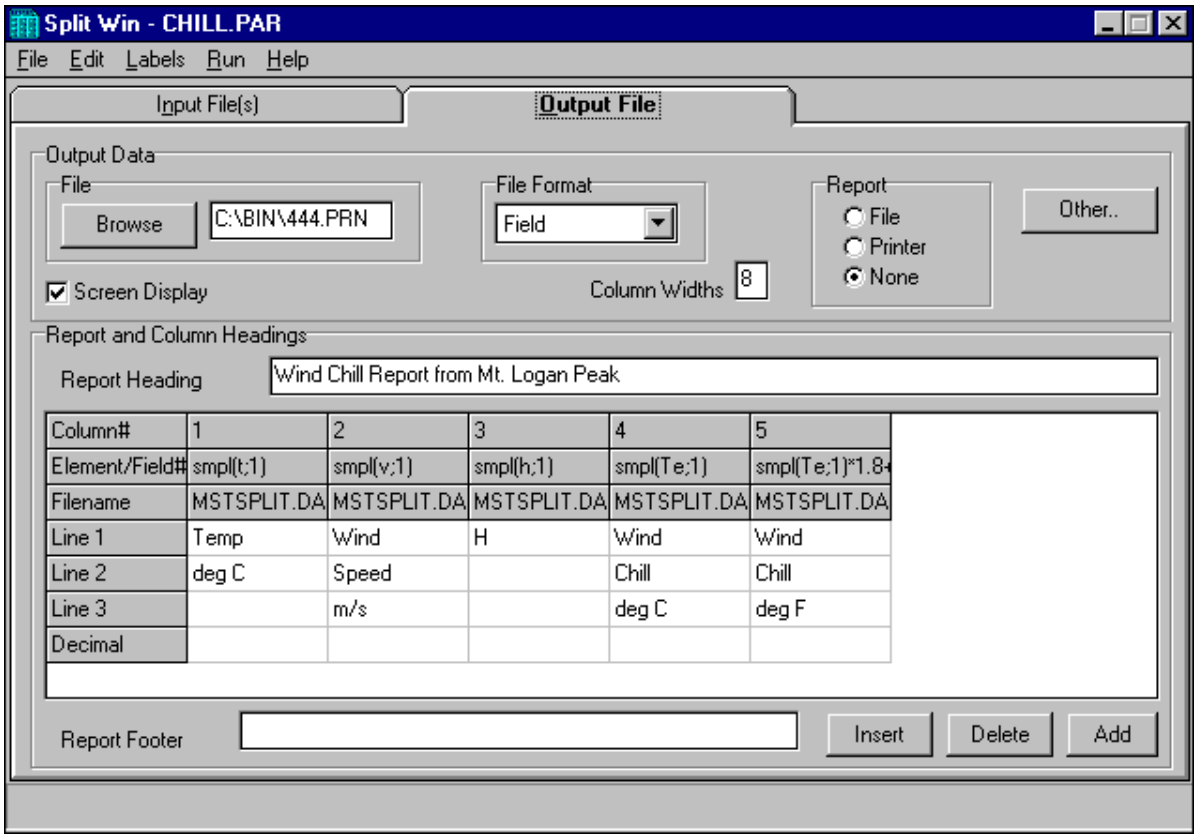
1	23.98	6.588
2	24.31	8.88
3	37.13	11.86

.PA Sends a page break to the printer or .RPT file.
Examples:
2, 3, Smpl (.PA;12) =

100	58.56
200	57.48
.	.
.	.
.	.
1100	61.34
1200	60.61
Page 2	----------
1300	61.01
1400	60.93
.	.
.	.
.	.
2300	55.48
0	55.22

The following is a parameter file that operates on the Mt. Logan data with several of the SPLIT features being utilized.





This .PAR file produces a wind chill summary of the Mt. Logan Peak data set. The formula for calculating wind chill is given as follows:

$$Te = 33-(h/22.066)$$

where

Te = Wind Chill equivalent temperature, degrees C

$$h = ((100V).5 + 10.45-V)(33-t)$$

where

h = Kcal m^{-2} hr^{-1} wind chill index

v = wind speed in meters/second

t = temperature in degrees C

Note that at low wind speeds this formula gives wind chill temperatures that are greater than the ambient temperature. The formula is for example purposes and is not endorsed by Campbell Scientific as a standard.

When this .PAR file is executed, the following output is displayed on the screen.

Wind Chill Report from Mt. Logan Peak

Temp deg C	Wind Speed m/s	H	Wind Chill deg C	Wind Chill deg F
14.756	2.6172	438.06	13.148	55.666
14.156	3.6967	489.58	10.813	51.463
13.806	3.4643	491.34	10.733	51.319
13.639	3.3972	493.4	10.64	51.151
13.65	4.6533	529.57	9.0005	48.201
12.961	4.0185	530.58	8.9547	48.118
13.306	4.2554	528.27	9.0596	48.307
14.511	2.9448	456.04	12.333	54.199
15.439	2.4397	414.97	14.194	57.55
16.161	2.066	383.21	15.633	60.14
16.3	2.6489	402.08	14.778	58.601
15.894	3.0463	425.2	13.731	56.715
16.117	3.7325	439.59	13.078	55.541
16.072	4.8812	468.26	11.779	53.202
16.833	3.7682	421.85	13.882	56.988
17.639	3.9694	405.59	14.619	58.314
18.972	3.5626	361.39	16.622	61.92
19.628	3.0208	331.76	17.965	64.337
19.217	3.1648	345.62	17.337	63.207
18.067	3.9157	393.08	15.186	59.335
15.467	5.2791	493.51	10.635	51.142
13.439	6.9821	584.71	6.5016	43.703
13.044	7.6526	607.86	5.4526	41.815
12.9	5.3014	566.29	7.3368	45.206

3.3.5.4 Summary of Select Line Syntax Rules

- A fixed numeric value must include a decimal point "." or be in scientific notation. There are some exceptions to this as noted below.
- Scientific notation has the format "mantissa E power of ten" (e.g., 3E5 = 3 x 10^5).
- Element numbers are entered without a decimal point.
- Commas separate Select line parameters (e.g., 2,3,(3+4)/3.2,6).
- Two decimal points are used to select consecutive elements between starting and ending elements (e.g., 3..6, refers to the elements 3,4,5, and 6).
- A set is a group of two or more elements and/or expressions separated by commas and enclosed by parentheses. No member of a set can include parentheses. Therefore, a set cannot include a set or a function as one of its members. For example:

VALID EXPRESSION	INVALID EXPRESSION
Arctan (2/3)	Arctan ((2/3))
Arctan (2/3, 3/4, 4/5)	Arctan ((2/3, 3/4), 4/5)
Arctan (COS(2))	Arctan (COS(2), COS(3))

- A single expression can operate on a set of elements. For example, the expression (3..6,8)/2.0 is the same as 3/2.0, 4/2.0, 5/2.0, 6/2.0, 8/2.0; (3..6)/(2..5) is the same as 3/2, 4/3, 5/4, 6/5.
- The element or expression which is the argument of a math or Time Series function, must be enclosed in parentheses. A range of elements can be specified, resulting in as many

outputs as elements (e.g., Avg(3..5,7) will output 4 averages).

- Square brackets are used to enclose an allowable range for a value (e.g., 3[3.6..12] indicates that the allowable range for element 3 is from 3.6 to 12). Whole numbers within brackets do not require a decimal point. Table 3.3-6 explains how values outside the specified range are treated.

- The count in a Time Series function is optional and does not require a decimal point.

- Semicolons are used in Time Series functions to separate the elements or expressions from the count which determines the interval. Sample on maximum and sample on minimum require two elements or expressions also separated by a semicolon.

3.3.6 OUTPUT FILES

To create an Output File, click on the **OUTPUT FILE** tab. The file is created on the default drive or directory unless the filename is preceded with an alternative drive or directory. Use the **Browse** button to change directories.

SPLIT will assign this file an extension of .PRN if an extension is not specified by the user. Whenever an Output Filename is entered, regardless of extension, an Output file is created only when the RUN | GO menu option is selected.

If this line is left blank, SPLIT does not write data to an Output File on disk; rather, it will display the processed values on the screen if the Screen Display box is checked. If Screen Display is not enabled, no data will be displayed on the **SPLIT RUN STATUS** screen.

> **CAUTION:** The Output Filename cannot be the same as the Input Filename. SPLIT will display an error message if this condition occurs.

Several output options may be specified to alter the default output to the file. Some are located on the main **OUTPUT FILE** screen and some are made available by pressing the **Other** button.

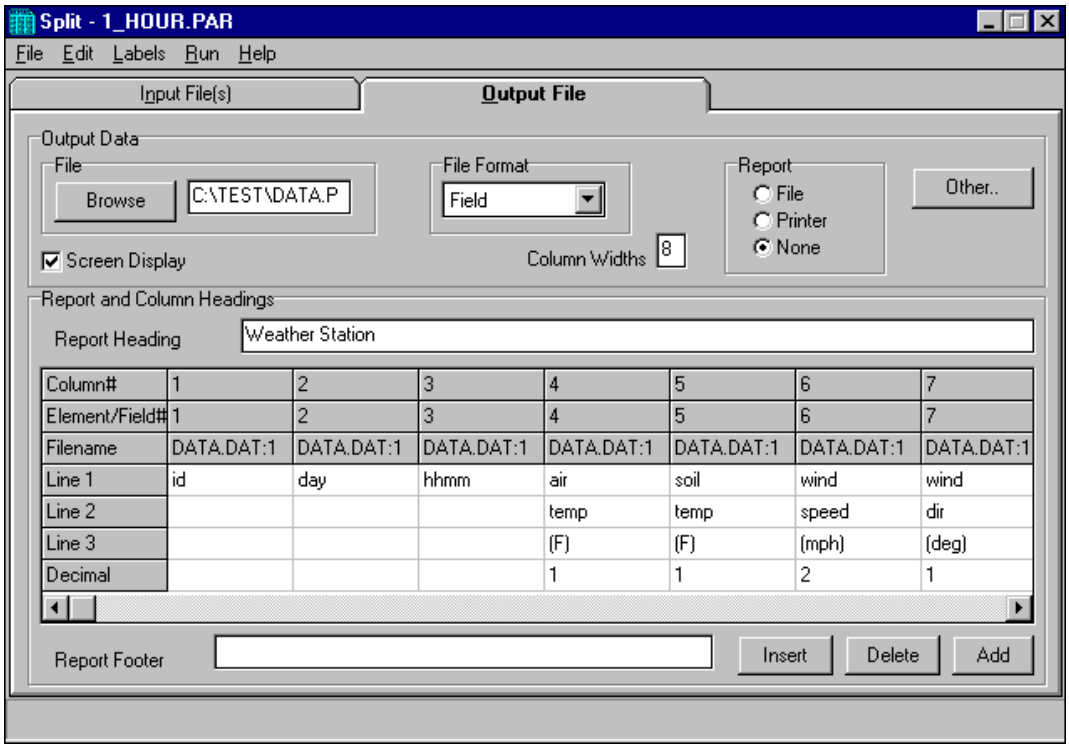


Figure 3.3-1. Output File Template

3.3.6.1 Description of Output Option Commands

File Format
There are four File Format options to choose from: No File, Field, Comma, and Printable. If No File is chosen, then no output file is saved to disk. The Field, Comma, and Printable options produce files formatted as Field Formatted, Comma Separated, and Printable ASCII, respectively. An example of each of these file types is given in Table 3.3-1 in the Input Files section.

Column Widths
The Column Widths field is used to set the width of the columns. Valid entries are 6,7,8, and 9. The default width is 8. High Resolution Final Storage data requires a minimum column width of 8.

Screen Display
The Screen Display field controls writing the processed data to the screen. To write to the screen, check the box. To omit writing to screen, for faster execution, uncheck the box. The data will then be written to the file only.

Report
A report, with page and column headings, can be sent to a file or printer. Select the appropriate option in the Report field. A report sent to a file has the extension of .RPT. If the report is sent to a printer, the printer must be on-line. The default Output File is also created.

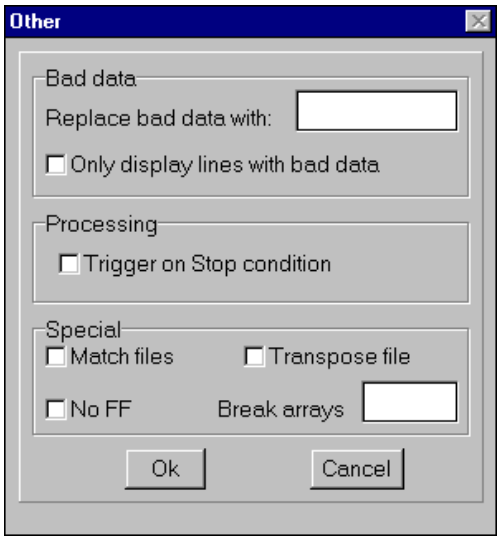


FIGURE 3.3-2. Other Output File Options

Other

The **Other** button allows the settings below to be modified:

Replace bad data with - The text in the field, to the right of this option, is entered into the data set if data are blank, bad, or out of range. See Table 3.3-9 for definition of blank or bad data. Whatever the user enters in the field will be entered if a blank or question mark is in the data or if data are out of range. This option is useful when the Output file is imported into a spreadsheet program, such as Excel.

TABLE 3.3-9. Definition of Blank or Bad Data for each Data File Format

File Format	Definition of Blank or Bad Data
Printable ASCII	????
Comma Separated ASCII	blank or any character except numeral or space
Field Formatted	blank or "" (double quotation marks)

Only display lines with bad data - Outputs only those arrays containing one or more Out of Range elements. If a report is generated, an asterisk precedes the Out of Range value in the .RPT file.

Trigger and Stop condition - Changes the meaning of Stop Condition to trigger Time Series processing output. The Stop Condition is included in the Time Series processing if it satisfies the Copy line.

If the Trigger and Stop Condition is selected, a Time Series output will occur each time the Stop Condition is met. See Select line elements (Section 3.3.3.2).

Match files - This option compares two files of the same data. If good data exists in one and not the other (question marks), then SPLIT will fill the OUTPUT file with the good data. This is used to get a more complete record from an error ridden file (e.g., one recorded at freezing

temperatures by reading the tape twice and running both files through SPLIT).

> **CAUTION:** For the Match files option to produce a correct Output File, the differences between the two Input Files can only be question marks. Both files must have the same Start Condition or the beginning of both files must be the same.

Transpose file - Transposes the rows and columns of the input file. Only one Input File can be transposed at a time and no Select options can be specified. A maximum of 26 arrays are transposed per pass of SPLIT.

To transpose a file containing more than 26 arrays, several passes are required. Change the Output Filename and Start Condition for each pass. SPLIT may then be used to merge the multiple files.

No FF - Suppresses form feed and page breaks to printer or .RPT file when a report is generated.

Break arrays - This option breaks up the Output Array into new arrays that are #+1 elements in each new array. SPLIT automatically assigns an array ID number equal to the first element in the first array. Only one Input File may be specified. Start, Stop, and Copy Conditions may be specified, but the Select line must be left blank.

3.3.6.2 Report Headings

A report is output to a printer or file with the extension .RPT. Headings are not included in the standard output to disk (.PRN or user named extension output file). However, a report can be labeled with a header by entering text into the Report Heading field. A report heading can have several lines, but it is limited to a total of 255 characters including backslashes and carriage returns. "\" denotes a new heading line for the report.

When Time Series functions are used in the Select field, they appear as a final summary at the end of the report. They can be labeled by entering a title into the Time Series Heading field at the bottom of the Output File page. Time Series interval summaries cannot be assigned individual titles.

"PCDATE" within the Report Heading inserts the computer's current date (Month-Day-Year). For the European format (Day-Month-Year), enter "PCEDATE".

3.3.6.3 Column Headings

Up to three lines per column can be entered as column headings. These headings are limited to a length of one less than the Output field width.

Column headings associated with Time Series outputs are repeated for Final Summaries if a title for the Final Summary is requested on the headings for report line.

The number of digits to report to the right of the decimal point is entered in the Decimal field and is unique for each column. The value output will be rounded to the specified number of digits. Leave this field blank if you do not want to round the data to a specific number of digits.

3.4 HELP OPTION

On-line Help is available from any location in SPLIT. Simply select the area of SPLIT in question and press <F1>. SPLIT also offers a brief on-the-fly Help. Place the cursor on the area of SPLIT in question; after a moment a brief description pops up.

3.5 EDITING COMMANDS

SPLIT supports the Windows Cut, Copy, and Paste commands. Text from any field in SPLIT or other Windows applications can be Cut, Copied, or Pasted.

3.6 COMMAND LINE ENTRIES

Existing parameter files can be executed from a Windows 95 or Windows NT command line prompt using the executable file called SPLITR.EXE. SPLITR.EXE is a Windows program and cannot be executed from a standard MS-DOS command line prompt (therefore, it will not run from the Windows 3.1 MS-DOS prompt). If a parameter file named "LOGAN" already exists, the command line "SPLITR LOGAN/R" executes the parameter file and returns to the command line prompt. The /R switch alerts SplitR that it is running

from the command line and closes the application when the specified files have been processed.

> **NOTE:** There should be no space before the /R switch. If there is a space, an error will be returned.

SPLITR allows the user to select different input and/or output files for an existing parameter file by entering them on the command line after the parameter filename. For example:

"SPLITR LOGAN/R TEST.DAT TEST.PRN"

Replaces the Input and Output filenames in LOGAN.PAR, with TEST.DAT and TEST.PRN, respectively.

A space must be used to separate parameters. SPLITR uses as many entries as exist on the command line. However, the command line is limited to 255 characters for Windows 95 and 127 characters for Windows NT. The parameters must be in the following sequence: Input Filename, Output Filename, Start Condition, Stop Condition, and Select.

If a parameter is to be left as it is in the parameter file, space comma space (,) may be entered in the command line. For instance, if the parameter file LOGAN.PAR contained TEST1.DAT as an input filename, the following command line would leave the input file TEST1.DAT and change the output file to TEST.PRN.

"SPLITR LOGAN/R , TEST.PRN"

More than one .PAR file can be executed with a single SPLITR command line. Each .PAR file and its associated parameters are separated from the next .PAR file by a semicolon with one space on each side (;). For example:

"SPLITR LOGAN/R TEST.DAT TEST.PRN ; SINKS/R TEST1.DAT TEST2.DAT 1[189]"

executes the LOGAN.PAR file on TEST.DAT and outputs the results to TEST.PRN, then executes the SINKS.PAR file on TEST1.DAT and outputs the results to TEST2.DAT. Execution of SINKS.PAR starts when the first element in TEST1.DAT is 189.

Batch files containing one or more SPLITR command lines can be useful for automating data processing. Batch files can be executed manually or by setting them up as a Task in PC208W's Setup window.

Batch files process each command in succession, without waiting for execution of a command to be completed before proceeding to the next unless they are configured to do so. If multiple parameter files are being processed using SPLITR in a batch file, there are no conflicts because only one copy of SPLITR can be active at any one time. However, if other commands are used along with SPLITR (such as opening the file in a spreadsheet, copying it to an archive directory, or appending it to an existing file) these commands might be executed before SPLITR finishes processing data.

The Windows 95/NT Start /w (wait) parameter is added to a batch file command line to delay execution of the next command until the first command has finished. The syntax for this type of command line is:

Start /w SPLITR LOGAN/R

By adding the Start /w parameter to the command line that initiates SPLITR, no further commands will be processed until the LOGAN.PAR file has finished execution.

Multiple copies of SplitR can be run at one time by using the /M switch. This switch must appear immediately after SplitR. For instance, a batch file or PC208W task containing the lines:

SplitR /M Logan/R
SplitR /M Sinks/R

will open two copies of SplitR and process the two files simultaneously.

SECTION 4. VIEW

4.1 INTRODUCTION

Pressing the View button on the main tool bar invokes PC208W's file viewing program. This program can be used to look at any data file collected by PC208W from dataloggers and storage devices, regardless of the format in which it was saved (comma separated, binary, or printable).

Once a file is opened, data can be printed, graphed, or displayed in ASCII, tabular, or hexadecimal format. However, since View is primarily a file viewing utility, a file cannot be edited or saved using this program.

4.2 MENU ITEMS

All of the tools available in the pull-down menus can be accessed using the graphical icons. Each of these icons is explained below.

4.3 ICON ITEMS

Following is a description of the icons that appear below the menu items. If you move and hold the mouse pointer over an icon for a few seconds, a brief text message will pop up that describes the function that will be performed if the button is pressed.

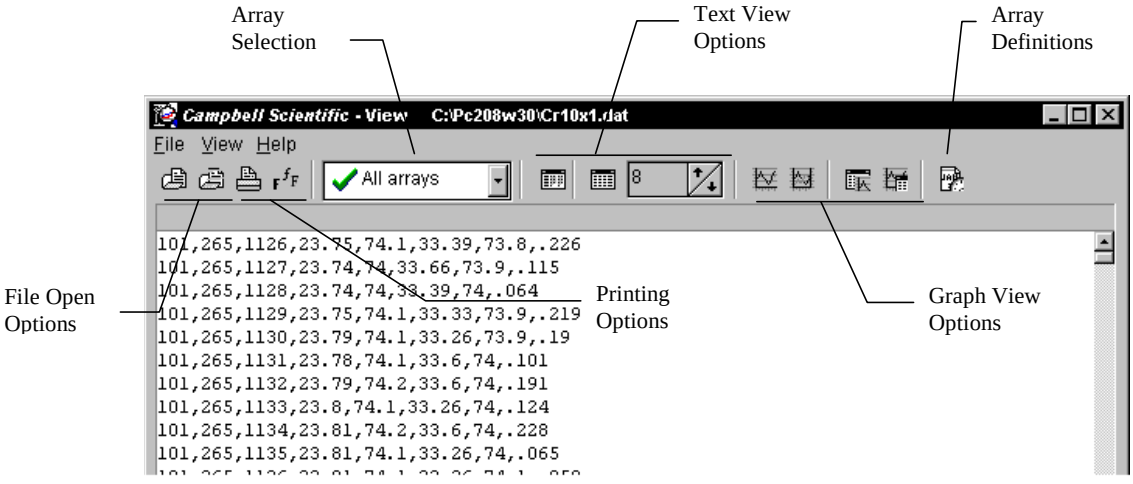
FILE OPEN OPTIONS

Select the icon (first icon) to open a data file. The default file extension is DAT, but PRN and CSV files can also be opened in this mode. (Opening files of types other than these may cause unexpected results in View.)

When a data file is opened in this mode View will automatically associate an FSL file with a matching name if one exists in the active directory (see Final Storage Label below). If a matching FSL file does not exist, a box will appear asking whether or not you want to select one. When an FSL file is associated with a data file, View will use column headers and time stamps from this file when viewing or graphing an array.

> **NOTE:** View will try to assign a time/date stamp automatically when opening a data file. Therefore, it may take a few minutes to completely open a large file.

Select the icon (second icon) to open a file in Quick Open mode. In this mode files are not processed for time stamping. A file opened in this mode can be viewed only in its original format or in hex mode. The Array Selection, Array Definitions, Tab options, and Graph View options do not work. This mode is most often used to open files other than data files (or to quickly open data files, but without any of View's graphing or time stamping capabilities). In Quick Open mode, data can be copied to the Windows clipboard and pasted into other applications.



PRINTING OPTIONS

Select the icon to immediately print the current data file. (If you prefer to set the margins of a page or perform a print preview before printing, this can be done through the File menu.) Press the icon to bring up a dialog box to change the font used for the printer and the display.

ARRAY SELECTION

When a data file is first opened, all arrays in that file will be displayed. Use the Array Selection list to pick a single array to be displayed, graphed, or printed.

TEXT VIEW OPTIONS

There are three icons that can be used to change the way data is displayed in View. When the icon is pressed, the data will be displayed in hexadecimal format. This is a toggle button, so if it is pressed again data will return to ASCII format.

Pressing the icon will change the data display from ASCII to tabular. This icon also acts as a toggle. The tab size can be changed by typing a number directly into the combo box or by using the up and down arrows to change the value. The tab size can be varied from 8 to 20 spaces.

GRAPH VIEW OPTIONS

To view data graphically, you must first select an array and then select up to two columns of data (elements) to be graphed. After the first column is highlighted, hold the control key and click on the second column to select it.

If the icon is pressed, both traces will be graphed on the same Y axis (on the left hand side of the graph). If the icon is pressed, one trace will be graphed on the left axis, and the other trace will be graphed on the right axis. By pressing one of the next two icons, you can choose whether the graph will always be on top, , or the data file will always be on top, . If one of these options is selected, you can press the icon again to return to normal view.

The graphical display includes the four graphing icons described above, a printer icon, , and a icon. When in graph view mode, you can click on a point to invoke a data label for that point. You can zoom in on a particular area of a graph by holding the left mouse button and dragging the mouse cursor from left to right over the area to be zoomed. To return to normal view, hold the mouse button and drag the cursor from right to left or press the icon. Right clicking on the graph will bring up a menu from which you can choose to preview the graph before printing, as well a change page margins, paper orientation, and printer setups for the graph.

If the data file you are viewing is larger than a few hundred lines, not all of the data will be graphed at once. You can scroll through the data on the graph by using the scroll bar. Scrolling through the data on the graph will also scroll through the data in the Data Panel and vise versa.

ARRAY DEFINITIONS

The icon on the toolbar is the Array Definitions icon. You can use this window to manually assign correct time and date stamps to a data file, or to change or add an FSL file association.

When you first click on the Array Definitions icon, a single column appears. One of the top lines of this window may read "No array selected". If you highlight a column in your data file for the array you want to assign values, View will automatically assign elements from the array to the date and time values if they exist uninterrupted by other elements in the order of day, time in the data file.

If these elements are not assigned automatically, you can assign values manually. Click anywhere in the Year, Day, Time, or Seconds box, and the window will expand to two columns. In this second column, you can choose to assign no element, an element from the data file, or a fixed value to any of the date and time values.

FINAL STORAGE LABEL

A final storage label (FSL) file is created by Edlog when a program is compiled. It provides information about the output tables in the program and the labels associated with each output instruction. The information in this file is used by View for column headers and to assign date and time values. To add or change an

FSL file association, press the file open icon to the right of the FSL File field and select the desired FSL file.

4.4 RUNNING VIEW AS A STAND-ALONE PROGRAM

Depending upon your computer's hardware specifications and operating system, you may experience low memory errors when trying to open large data files with View. It may be necessary to close other applications when running PC208W and View or to run View independently of PC208W.

View can be run as a stand-alone program by double-clicking the View.exe file in the PC208W\Bin directory. This should allow you to open larger data files without memory errors.

4.5 ASSIGNING DATA FILES TO VIEW

Windows will let you assign the program with which a particular file type will be opened based on that file's extension. When a file with an assigned extension is double-clicked, it will be opened with the associated program. You may want to associate *.DAT files with the View program for quick opening of data files.

In Windows 95/NT this association can be made by selecting **View | Options | File Types** from the Explorer menu. In Windows 3.1, this association can be made by **File | Associate** from the File Manager menu.

> **NOTE:** Only one association can be made for each file type. If the *.DAT file is assigned to another program, you must remove this association before the new association can be made.

SECTION 5. STG MODULE: SMS – STORAGE MODULE SOFTWARE

SMS is a Windows-based software package that provides a simple and efficient way of collecting and storing both data and programs in Campbell Scientific's Card Storage Modules (CSM1 and MCR1), Storage Modules (SM192 and SM716) or a PC (PCMCIA) Card. The software communicates with the Storage Modules through an asynchronous adaptor. The software is provided with an extensive on-line help system.

An SC532 Storage Module–RS232 hardware interface is required for direct connection of some of the storage modules to a computer.

5.1 INTRODUCTION

SMS is designed to run on a PC compatible computer running Windows. Data can be read directly from a PC card if the computer is fitted with a PC card reader. Using a special interface, SMS provides easy communication with Campbell Scientific's external Storage Modules, card readers and dataloggers for collection and storage of both programs and data.

> **NOTE:** The CSM1 Storage Module, the MCR1 Card Reader and a computer fitted with an internal PC Card reader all use SRAM PCMCIA cards which must formatted to Campbell Scientific's specifications.
>
> See Appendix D for more information about Card Services Driver Software.

All the SMS functions can be accessed from the **MAIN USER INTERFACE** screens. The package has an extensive on-line help system.

It is recommended, and assumed, that you will be using a mouse, but some functions can also be accessed using the keyboard "hot-keys" (using <Alt>_).

It is recommended that this manual be used in conjunction with the SMS on-line help system and the User Guide for the module you are using.

> **NOTE:** The CSM1, MCR1 and SM192/716 User Guides contain references to other Campbell Scientific software (CSMCOM and SMCOM). These programs are *not* required if you are using SMS.

5.2 GETTING STARTED

SMS runs under the standard Windows interface and is presented in graphical form. All functions can be accessed with a mouse by clicking on appropriate parts of the screen. An example screen for the SM192/716 module is shown below.

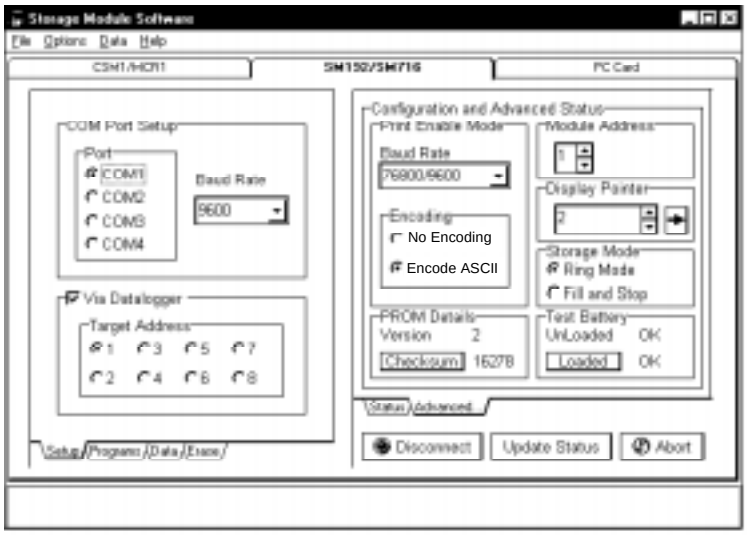


FIGURE 5-1. Typical SMS Screen

CSM1, SM192 or SM176 Storage Modules must be connected to the PC with an SC532 interface, plus a 7026 or SC25PS cable. A basic connection diagram is shown in Figure 5-2 below. The Storage Module manuals give full details of connecting the modules to the computer.

NOTE: The MCR1 Memory Card Reader does not require an SC532 interface.

If your computer is fitted with a PC card reader supporting Card Services protocol, SMS can be used to interrogate the memory card in a similar way to interrogating a card in the CSM1/MCR1 Storage Module. Please see Appendix D for specifications and requirements for PC Card support.

Start SMS by clicking on the **Stg Module** button. The first time you use the program it will show the screen for the CSM1/MCR1 Card Storage Module as in the screen in Figure 5-3.

5.2.1 THE SETUP SCREEN

The **SETUP** screen shown in Figure 5-3 is used to set the Communications (COM) Port and Baud Rate before communications between the module and computer are established.

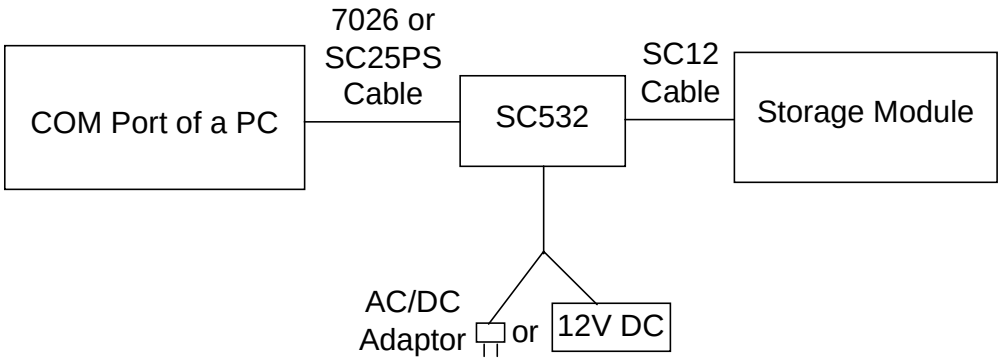


FIGURE 5-2. Connecting the Storage Module to a PC using an SC532 Interface

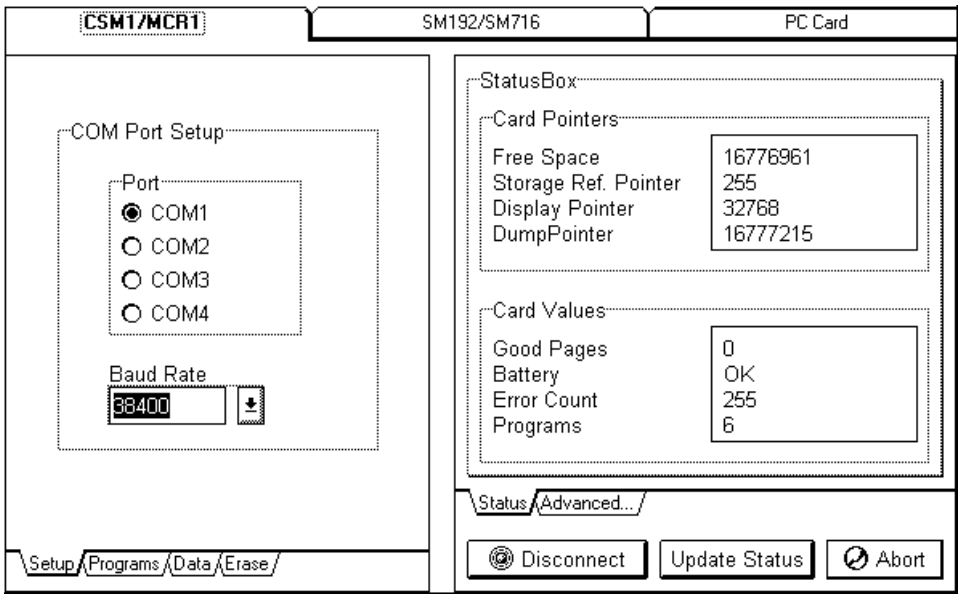


FIGURE 5-3. Typical SETUP Screen and StatusBox for the CSM1/MCR1 Module

5.2.2 ESTABLISHING COMMUNICATIONS

For storage modules, the first step is to establish communications between your storage device and the computer. This involves selecting the correct communication port and baud rate for the device being used from the **SETUP** screen.

If you have a computer fitted with a correctly configured card reader you can set it up for use with SMS. This is accomplished by going to the **PC CARD SETUP** screen, and clicking the **Setup PC Card** button. When you click this button you will be guided through the correct setup procedure and the optimum software drivers will be used to establish communications (usually via socket 0).

> **NOTE:** The PC Card drivers are NOT set up automatically, and should be set up as described above. Even if you are already using PC Cards successfully with your PC, it is recommended that you use the setup facility provided for use with SMS.

The following paragraphs in this section are applicable for Campbell Scientific's external Solid State Storage Modules.

Selecting a COM Port

For an externally connected storage device you can use one of your computer's Communication (COM) ports for connecting the storage module to the computer. The first time that you connect a module to your PC a default port of COM1 is automatically selected. If this is the port to which you have connected your module you can proceed to select the BAUD rate as explained below. If you are not using COM1 for connection, you will need to select the port being used by clicking on the appropriate checkbox in the Port Selection box. This will be COM1 or COM2 for most computers, but SMS can use COM3 and COM4 if your computer is equipped with these.

The physical implementation of the COM port is hidden from the software. The Windows environment provides a common interface via a COM port driver, with which SMS communicates. Sometimes, however, you might have a COM port which shares interrupts with other COM ports. This can be checked by examining the "Ports" (Settings/Advanced) setting in the Windows Control Panel. A COM port sharing an interrupt with another device can prevent Windows from performing high-speed communications using that port. This shared interrupt problem would, therefore, also affect operation of SMS.

> **NOTE:** If you appear to be experiencing communication problems when using external modules, always check the COM ports setting in the Windows Control Panel, using the "Ports" (Settings/Advanced) option, and check for any shared interrupts.

Baud Rate

The first time you use SMS with a storage module a default baud rate of 38,400bps and 19,200bps is automatically selected for the CSM1/MCR1 and SM192/716 modules, respectively.

The default (maximum) baud rate (38,400bps for CSM1/MCR1 and 19,200 for SM192/716) can be adjusted to your own requirements, depending on, among other things, the speed of your COM port and processor speed.

When SMS reads data from the storage device, it automatically performs a checksum test and requests that bad data be re-transmitted. To signify when bad data is detected, the computer will beep.

When communication errors occur, it may help if you reduce the baud rate. Possible reasons why good communications cannot be established using higher baud rates under Windows are:

- Using cheaper low-quality cards, or using cards but with generic drivers rather than the hardware specific drivers.
- Processor is loaded down by other applications (e.g., DOS programs set to exclusive mode or other operating Window communication programs).
- A computer without a buffered UART (Universal Asynchronous Receive/Transmit) serial port (e.g., a 16550 UART), or without the FIFO (First In First Out) option enabled. (Other communications problems can occur if the computer does not actually have a buffered UART, but is using the Windows 95 default configuration with the FIFO option enabled.)

- A high level of communications activity on another port.
- Lack of computer memory, causing the computer to do a considerable amount of hard-disk paging. If the SMS communication routines are paged to disk and take too long to load back into memory, the buffer can overflow.

If the communication errors are reported as CE_RXOVER errors, reducing the baud rate will help because it stops the buffer from filling up too quickly in situations where SMS is not able to respond to the incoming data.

Via Datalogger/Target Address (SM192/716 Modules Only)

When an SM192/716 Storage Module is connected with an SC12 cable to a datalogger which supports direct communications (CR10(X) or CR500), you can establish communications using the "Via Datalogger" mode as explained below.

Via Datalogger

As an example, when an SM192/716 module is connected to a CR10/10X or CR500 datalogger with an SC12 cable, direct communications can be established between the computer and the storage module (see Figure 5-4). Note, however, that dataloggers can only support a maximum baud rate of 9600.

SMS can automatically detect if a datalogger which supports direct communication is present and request that a connection be made to the Storage Module. In these circumstances the Via Datalogger checkbox can be used for connection. This does nothing more than set the baud rate to 9600 and enable the Target Address checkboxes. If you "uncheck" this box but keep the baud rate at 9600, you can still connect via the datalogger if one is there.

Target Address

The Target Address selector is disabled unless you check the Via Datalogger box. This is to prevent confusion, as the Target Address boxes only have an effect if a datalogger supporting direct communications is connected.

An appropriate datalogger can be connected to up to eight storage modules simultaneously, each having a different address. The address selector is used to specify the address of the module to which you wish to connect. Note, however, that any module having a high address may answer to a call of a lower address if no module of the lower address is present.

For example, if only one storage module with an address of 8 is present, and you set the target address to 1, you will still be able to connect to the storage module. However, if your module had an address of 1 and your target address was set to 5, you would *not* connect.

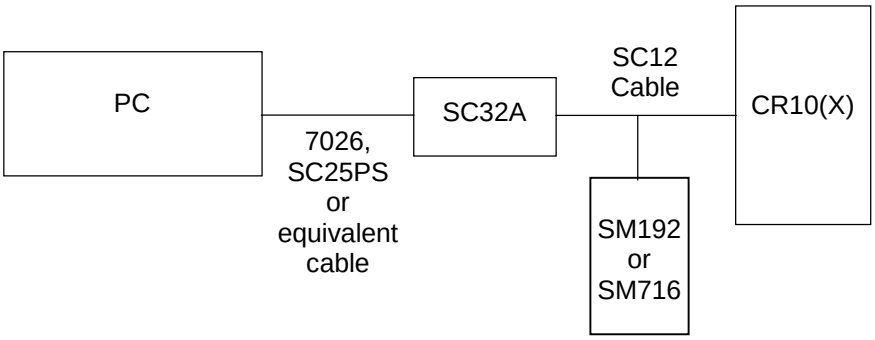


FIGURE 5-4. Connecting the SM192/716 to a PC and a CR10(X) Datalogger

The above shows that the target address need not represent the *actual* address of the Storage Module. The physical address of the Module is shown on the "Advanced" status page. Changing the physical module address will not change the selected target address, and, in fact, changing the physical address will not have any noticeable effect until you disconnect from the Storage Module.

Connect/Disconnect

After connecting your storage module and selecting the required COM port and baud rate, click on the **Connect** button to establish communication between the computer and the storage module. If everything is satisfactory, a successful communications link will be established and the wording on the **Connect** button will change to "Disconnect".

If your first attempt at connection fails and an error message "Cannot connect to CSM1" or "Communications Problems:..." appears, try to connect again (by clicking the **Connect** button). It may take some time (up to 30 seconds) to establish the connection. The most likely cause of a failed communication link is a loose physical connection between the storage module and the computer.

5.3 STATUS INFORMATION

5.3.1 STATUS BOX/UPDATE STATUS

When successful communication has been established, a set of values will appear in the StatusBox at the right-hand-side of the screen. This screen shows the current status (values) for the card or module. Some typical values are shown in Figures 5-1 and 5-3 above.

The information in the main StatusBox is relatively straightforward and is generally similar for all modules. The SM192/716 module has an additional Switch Settings information box which shows the settings made using the **ADVANCED SETTING** screen. Further information on each item in the **STATUS** screen can be obtained by clicking on that item to utilize the on-line help system.

The status information can be updated at any time by clicking the **Update Status** button. Additional information is also available in the appropriate Storage Module Manual.

5.3.2 ADVANCED STATUS INFORMATION

The **ADVANCED STATUS** screen is different for each type of module, and provides extra information and configuration facilities.

CSM1/MCR1

Display Pointer

The Display Location Pointer (DLP) indicates the location which holds the first value which will be read when extracting data from the card. The pointer can be positioned anywhere within the readable area of the card, either by typing the location directly into the window or by using the up/down arrows to increment the value. The green arrow moves the pointer to successive filemarks.

> **NOTE:** If the Display Pointer points to a filemark or the start of a program no data will be output in response to the "Get One" request. It must be moved past the filemark to get further data.

Further details on Filemarks and Pointers can be found in the relevant Storage Module Manual and the on-line help system.

Card Format Type

This is a format code for the card currently in use. It is an internal reference for use by Campbell Scientific and so has no significance for the end-user.

PROM Details

The PROM checksum calculates a unique signature for that PROM version. For all storage devices the PROM checksum is displayed on the **ADVANCED** screen. At start-up the value may be displayed as a series of ??????. The true value can be shown by clicking the **Re-Calculate Checksum** button. Once displayed, this value is latched and will normally remain until you disconnect or remove power from the storage module. You can recheck the value at any time by clicking the **Re-Calculate Checksum** button.

SM192/SM716

The SM192/716 **ADVANCED STATUS** screen is similar to the **CSM1** screen, but has additional configuration facilities. These are listed below.

Print Enable Mode

Data from 21X and CR7 dataloggers with older PROMs, and data from non-datalogger devices are stored in the Storage Module using the "Print Enable" mode. Full details of this storage method can be found in the SM192/716 Instruction Manual.

Baud Rate – Print Enable Mode

This is the printer baud rate when data are collected by the Printer Enable (PE) method. For this method of collection you must ensure that the baud rate of the Storage Module and the datalogger are matched. Most Campbell Scientific dataloggers have selectable baud rates. You can select the baud rate using the Baud Rate setting box in the Print Enable Mode section. The SM192/716 Instruction Manual gives further details.

> **NOTE:** The 76,800/9600 setting is the default after an SM reset. In this setting the SM can automatically change from capturing data at 76,800 baud to capturing data at 9600 baud. Baud rate errors (baud rate mismatches between SM and datalogger) cause the SM to increment its error counter and enter a low power standby state.

Encoding

These checkboxes allow you to select how data are encoded in the Print Enable method of data storage:

Encode ASCII is the normal default setting. The Storage Module will encode incoming Printer Enable Method printable ASCII data and store it as Campbell Scientific Final Storage Format (binary) data.

No Encoding enables incoming data to be stored "as is", regardless of its format. This permits storage of raw ASCII data.

> **NOTE:** Raw ASCII data will take up much more storage space than encoded data, and so it is generally recommended that data are always stored as binary.

Module Address

This shows the physical address of the Storage Module. Changing the physical address will *not* change the selected target address.

See Target Address for further details.

Display Pointer

This has exactly the same function as the Display Pointer for the CSM1/MCR1 (see above).

Storage Mode

Two methods of memory allocation for storage of data are provided – Ring Mode and Fill and Stop, as explained below.

Ring Mode: When the storage module is set to Ring Mode, data are stored sequentially. When all memory locations become full the ring memory "wraps around" and new data are then written over the oldest data stored in memory. To avoid data loss, retrieve all data from the storage module before this occurs.

Fill and Stop: When the storage module is set to Fill and Stop, data are stored sequentially, but, when all memory locations become full, no more data are stored.

When multiple storage modules are used, each should be assigned its own address and set to Fill and Stop. When one module is filled, data will then be stored in the next module in the chain.

Test Battery (Unloaded)

The value shown under "Test Battery" is for the state of the battery when compared to a 2.5V DC reference without any load on the battery.

Test Battery (Loaded)

The battery can be checked when a 100 ohm load is connected for 1 second. This test shortens the overall battery life by approximately 3 minutes, but it is a much better test of the battery's integrity. It is particularly important to carry out a Loaded Battery Test if the battery is more than four years old and you are intending to leave the Storage Module unattended to collect data for several months or

you are leaving the module unattended to collect data in cold weather.

> **CAUTION:** Failure to check the battery can lead to data loss in the above conditions.

5.4 PROGRAMS

Clicking the **PROGRAMS** tab brings up the **PROGRAM CONTROL OPTION** screen. The screen is very similar for both the storage devices and the PC Card. It is essentially an interface for extracting programs, saving programs or clearing programs from the storage device.

The screen in Figure 5-5 shows the program selection checkboxes and the program manipulation buttons. Datalogger programs can be stored in a card or module for subsequent downloading into a datalogger. The programs are stored in the same memory space available for data storage and are stored in up to eight memory locations. With the correct version of datalogger software, the program in location 8 can be automatically loaded, compiled and run in a datalogger on power-up of the datalogger. (See appropriate Storage Module Instruction Manual for more details.)

There are also three buttons – **Clear**, **Store** and **Read**.

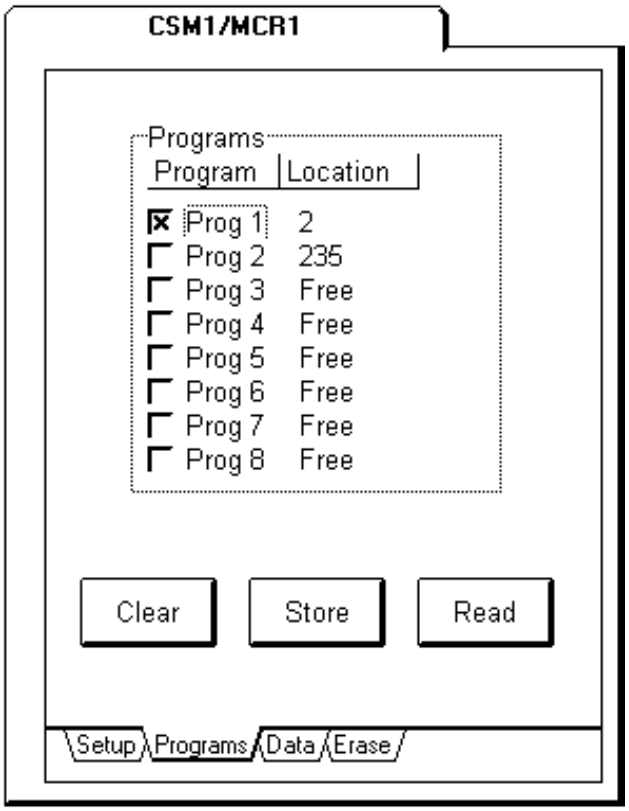


FIGURE 5-5. PROGRAMS Screen

5.4.1 PROGRAM LOCATION

There are eight memory locations available. The Program Status box will either:

- Display a number indicating the starting position in memory of the program (CSM1/MCR1 and PC Card),
- Say “Used” for a program in a specific location (SM192/716),
- Say “Free” for an unused location.

A program location is selected by clicking on it to highlight its checkbox. Once you have selected the program location(s) you can carry out the desired action by clicking on one of the three buttons.

5.4.2 CLEAR

Clicking on the **Clear** button will clear the selected programs from the card or module memory. You will be asked to confirm this request before the program(s) are actually cleared.

5.4.3 STORE

Clicking on the **Store** button will allow you to store a program in the selected program location. If a program already exists in this location, you will be asked to confirm that it can be overwritten. Once a valid location has been selected, a Program Selection box appears showing any valid program files (.dld extension) on the current drive and directory. You can select any of these, or choose any other valid program file by browsing the directory structure. Once you have selected the required program, it will be written to the storage card or module. A Progress Indicator at the bottom right of the screen will confirm that the program is being written to the card or module.

5.4.4 READ

This is the reverse operation to the Store request. Clicking the **Read** button will allow you to read a program from the selected program location to disk. Once a location containing a program has been selected, clicking on the **Read** button brings up a file selection menu for you to select the location in which to store the program on disk. A progress indicator will confirm that the program is being read from the card or module.

5.5 DATA

Clicking on the **DATA** tab will bring up the **DATA CONTROL OPTION** screen as shown in Figure 5-6, below.

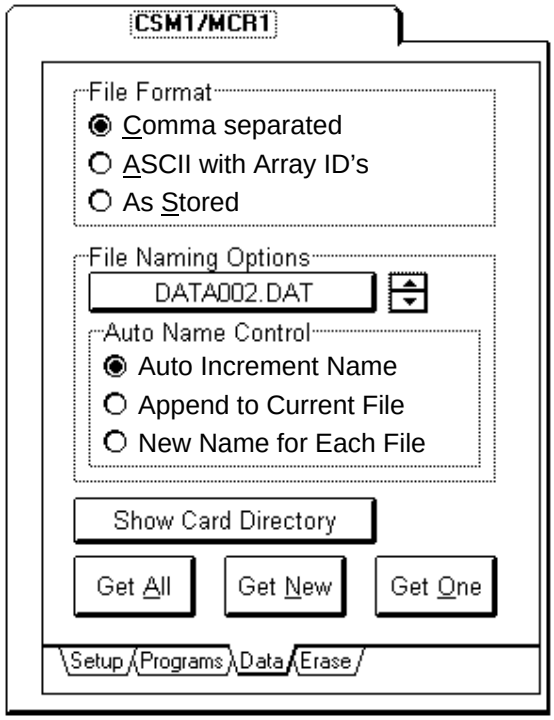


FIGURE 5-6. DATA CONTROL Screen

5.5.1 FILE FORMAT

Before you attempt to "get" (collect) any data you should ensure that you are using the file format required.

ASCII with Array IDs

This format is ASCII with Array Identifiers, and is the most "readable" when printed. Each data value is preceded by an identifier indicating its position in the array. This format requires the most disk space on the computer, and can be difficult to use in some data analysis packages.

This format is also ambiguous for arrays of 100 or more data values. It is therefore not recommended for purposes other than printing.

> **NOTE:** If the data was stored from the datalogger in ASCII format, you should use the "As Stored" format to collect data from the module (see below). Storing data directly in ASCII format is inefficient in terms of use of the module memory, as printable ASCII requires five storage locations in the module (10 bytes) as compared to one (2 bytes) for binary data.

Comma Separated

This format has array elements separated by commas. No element identifiers are included and all leading zeros and trailing zeros after the decimal point are removed. Each array is terminated with a carriage return and line feed. Comma separated format is comparatively compact, can be read when printed, and can be imported into most data analysis packages (e.g., if you use a .CSV file extension, Excel will auto-import the file).

As Stored

When this option is used the software does not perform any decoding on the data in the card or module. Data are read out of the module and written to disk "as it is" (8-bit data).

If data has been stored from a datalogger in a binary format, this option writes the data to disk in the same format. This is the most compact format for data storage on disk, but the data needs to be decoded before it can be read. SPLIT (see Section 3) can be used to decode the data. If you wish to decode data with your own software, refer to the appropriate datalogger Operator's Manual.

5.5.2 FILE NAMING OPTIONS

Data files can be collected either individually or collectively. The File Naming Options box gives you control over how the extracted files are collected and named.

Filename

Data files will normally be stored under a filename with the .DAT extension. When first using SMS, the button in the **FILE NAMING OPTION** screen will show the name DATA000.DAT. This name can be changed to any other name of your choice. By clicking on the button, a standard Windows Filename box appears that will allow you to type in the name you want.

Auto Name Control

You may choose to Auto Increment the filename, to Append the data to a current file or to manually choose a New Name for each file saved.

Auto Increment Name

When the software is first used, the program will always select a default filename and use automatic file incrementing. This means that the filename will initially be DATA000.DAT. Unless you change the default, further files will be saved in the sequence DATA001.DAT ... DATA999.DAT. This means that whenever you store new data files to disk, the name will automatically increment by adding one to the number at the end of the name.

If you are collecting multiple files (as in the Get All command), the filename will be incremented automatically as each file is collected. The program will scan the disk for files with the same base filename. If a file is found with a higher numeric suffix than the one you enter, the program will ask for confirmation to avoid the possibility of existing files being overwritten.

You may change the "base" filename by clicking on the **Filename** button, but the last three letters of the eight possible characters will automatically be converted to digits if you have chosen auto incrementing.

For example, if you chose a filename of ANDREW.DAT the program would convert it to ANDRE000.DAT so that auto incrementing could take place.

If you want completely unique names for each file choose the “New name for each file” option, and ensure that auto incrementing is off.

Append to Current File

In this option files to be collected are added to the end of the current file (as indicated by the name on the **Filename** button). The new file will retain the current filename.

New Name for Each File

By selecting this option you have the flexibility to choose a completely different name for each collected file. The names can be any name conforming to standard Windows convention – up to eight characters long. As the files are collected, you will be prompted to enter a valid filename for each individual file.

5.5.3 SHOW CARD/MODULE DIRECTORY

By clicking the Show Card Directory (CSM1/MCR1/PC Card) or the **Show Module Directory** (SM192/716) buttons, a secondary window showing the directory structure appears. Initially, the directory is in a concise, or folded, form showing just the locations of the file-marks. Click on the **Unfold** button to expand the directory to show program and data file locations and other filemarks and pointers. Data/Programs can be read into a file and/or viewed directly from the **DIRECTORY** screen using the **Read File** and **View File** buttons. For a card or module with a large number of programs or data files the Show Directory command may take several seconds to complete.

The Show Directory command is repeated on the menu bar. See the on-line help system for further details.

5.5.4 GET ALL

This option collects *all* valid data files in the card or module and stores them to the specified directory, starting with the filename of XXX.DAT shown in the Filename box. The way that the file is named is controlled from the **Auto Name Control** selection buttons (see above).

As the files are read to the disk, a Progress Indicator showing the filenames is displayed at the bottom right of the display.

5.5.5 GET NEW

This option collects new data files, starting from the position of the storage module dump pointer. This pointer marks where data was last collected, and is automatically moved after each successful collection of data. This option can be used to collect any new data files stored in the card or module since data was last read. As the file is read to the disk a Progress Indicator showing the filename is displayed at the bottom right of the display.

See the relevant Storage Module Manuals and on-line help system for further details about Filemarks and Pointers in memory.

5.5.6 GET ONE

This option collects one data file starting at the specified location of the display pointer. You can position the display pointer to any starting position required by using the **ADVANCED STATUS INFORMATION** screen (see section 5.2).

This option is used mainly for re-collecting old data that has already been collected, or to recover data from a corrupted card.

As the file is read to the hard (or floppy) disk a Progress Indicator showing the filename is displayed at the bottom right of the display.

5.6 ERASE

Clicking on the **ERASE** tab will bring up the **ERASE OPTIONS** screen. This gives control over how data is erased or cleared from the card or module.

5.6.1 ERASE DATA

This option “erases” data by removing the references to the data files and erasing all file marks. Only DATA from the card or module is affected. Any programs are temporarily off-loaded to disk, all pointers to data and programs on the card/module memory are erased and the pointers reset. The PROGRAMS are then copied back to the card or module. Before any data is “erased” you will have the chance to cancel the option.

5.6.2 ERASE DATA AND PROGRAMS

This option will completely "erase" (see above) all data and programs from the card or module. You will have the opportunity to cancel the option before the card/module is erased.

5.6.3 ERASE AND TEST CARD/MODULE

This option is similar to the Erase Data and Programs option except that a full integrity test is carried out on the card or module. This option takes considerably longer than the simple Erase Data option, and all memory location values are reset.

5.7 THE MENU BAR

The Menu Bar appears along the top of all **SMS** screens with four options which can be accessed either by clicking with the mouse, or from the keyboard by pressing <Alt> + Underlined Character (F, O, D or H). Each option provides a further pull-down menu. The underscored character in the name is the "hot-key" for keyboard access.

5.7.1 FILE

The **File** pull-down menu gives three additional options:

Terminal Emulator

Clicking on **Terminal** on the pull-down menu produces a **TERMINAL EMULATOR** screen. It allows you to communicate directly with the Storage Module and is provided so that you can issue storage module telecommunication commands manually. When the **TERMINAL EMULATOR** screen is opened, it automatically establishes a connection to the storage module. The storage module prompt (%) should be displayed while the connection is active. This option is seldom used because all normal operations can be accomplished easily by using other sections of the software.

Keyboard entries are transmitted from the COM port to the storage module, and storage module responses are displayed on the computer screen. Pressing <Esc> exits this mode and returns you to the main **SMS** screen. You are unlikely to need Terminal Mode unless you are trying to diagnose problems with a corrupted module.

Full details of direct module communication commands are given in Campbell Scientific's Storage Module Instruction Manual.

Show Directory

Clicking Show Directory produces exactly the same result as clicking the **Show Card/Module Directory** button (see Section 5.5.3).

Exit

Clicking Exit on the Menu Bar will close all communications completely, exit the program, and return you to Windows. Alternatively, you can exit the system by double-clicking the small box at the top left-hand corner of the **SMS** screen or by clicking this box and choosing **Close** from the pull-down menu. Windows 95 users can also exit using the small "Exit" icon at the top right-hand corner of the screen. Do not use this option merely to close the pull-down menu and return to the **SMS** screen. To return to SMS, click the mouse anywhere on the **SMS** screen outside the pull-down menu box.

5.7.2 OPTIONS

The options menu provides a pull-down menu with two options:

Enable Pop-Up Hints

Clicking this option acts as a toggle. If Pop-Up Hints are already enabled, a check appears at the left hand side of the name. By clicking on the name, the Pop-Up Hints in the **MAIN PROGRAM** screen can be disabled. An additional click will re-enable them.

Save Settings on Exit

If you want to save all your set-up information when you leave SMS, ensure that Save Settings on Exit is checked. Again, this option may be enabled/disabled by clicking the mouse on the name.

5.7.3 DATA

Selecting **Data** from the Menu Bar provides a pull-down menu with five items. The first three—**Get All**, **Get New** and **Get One** are identical to the similarly worded buttons on the **SMS DATA** Screen (see Section 5.5).

The two remaining items on the pull-down menu are:

Insert File Mark

File Marks are used to separate data in the storage device. A File Mark is automatically placed in the device's memory when it is connected to a power source (a datalogger or retrieval interface) or when a datalogger compiles a program containing Instruction 96. For example, if you have retrieved data from one datalogger and disconnected the storage module and then connected it to another datalogger, a File Mark is placed in the data when the second datalogger is connected. This mark follows the data from the first datalogger and precedes the data from the second.

The pointer can be moved to the next file mark in the data by clicking on the green arrow. Repeated clicks move the pointer to subsequent file marks until the last one is read.

Next File Mark

The Next File Mark option moves the Display Pointer to the next file mark in the data in a similar way to clicking on the green arrow in the **ADVANCED STATUS** screen (see Section 5.3.2).

5.7.4 HELP

The final option on the Menu Bar is Help. This provides a pull-down menu with five items connected with the SMS on-line help system and Windows help in general. The options are self-explanatory.

Note that the option "About..." will display the current version number of the software. This may be required if you need to contact Campbell Scientific for further information or support for SMS.

"About..." also indicates if any Card Services drivers have been found on your system.

5.8 ABORT

Most SMS operations can be aborted by clicking the **Abort** button. For example, if you click Abort while the software is attempting to establish communications between a storage module and computer, that operation will be aborted. Note that for some rapid operations, such as reading or storing small program or data files from a module, the operation may be completed before the abort option can take effect.

SECTION 6. PROGRAM CONVERSION

6.1 DOC2CSI

DOC2CSI is an easy-to-use DOS utility that converts datalogger programs created with Edlog Version 5.4 or earlier (*.DOC files) to the *.CSI format files used by more recent versions of Edlog (DOS Version 6.2 or greater and Windows PC208W Version 2.0 or greater). Conversion of these older files allows you to use them with PC208W without performing the time-consuming task of recreating each program.

6.1.1 COPYING PROGRAM FILES

Though the DOC2CSI utility is not a menu option that can be invoked from PC208W, it has been shipped as part of the PC208W software suite. When the PC208W software is installed, DOC2CSI is installed in the PC208W\BIN directory.

DOC2CSI could be run from the PC208W\BIN directory. However, to avoid inadvertently overwriting critical program data, you should copy the DOC2CSI program files (and, for convenience, your old program files) to the PC208W working directory.

> **NOTE:** If you choose a folder name other than the default for the PC208W program when installing the software, your PC208W working directory will be different than that used as the example in this section.

Invoke a DOS command line and change to the PC208W working directory (i.e., C:\PC208W). Both Windows 3.X and Windows 95/NT will allow you to "shell out" to DOS by clicking on an MS-DOS icon, or, with Windows 95, choosing **Command Prompt** from the **Start** menu.

To copy the DOC2CSI program to the PC208W directory, at the C:\PC208W prompt type:

copy .\bin\doc2csi.exe

You should receive a message that the file was copied successfully.

To copy all of your existing program files to the PC208W directory, at the C:\PC208W prompt type:

copy c:*oldfiles**.dld

and

copy c:*oldfiles**.doc

where *oldfiles* is the full directory path where your existing files are located.

6.1.2 RUNNING DOC2CSI

To run DOC2CSI, at the C:\PC208W command line type the following command string:

bin\doc2csi *oldname newname* loggertype

where,

oldname is the *.DOC filename
newname is the filename for the new *.CSI file that will be created
loggertype is the type of datalogger for which the program was written. Valid entries are 21X, CR7, CR10, CR10X, and CR10T.

It is not necessary to type the *.DOC and *.CSI extensions for the files in the command string. If you did not copy your existing program files to the PC208W working directory as suggested above, you will need to include the *oldname* file's full directory path.

If you forget the format of the command string, you can type DOC2CSI at the DOS prompt and the program will provide syntax information.

6.1.3 EXAMPLE USAGE

As an example, if you want to convert a program named *weather.doc* to a CSI format file named *weather1* for use on a CR10X, you would type the following at the C:\PC208W prompt:

bin\doc2csi *weather weather1* CR10X

DOC2CSI will create the file *weather1.CSI*. This new file will be placed in the PC208W working directory. (Again, the above example assumes that the file *weather.doc* resides in the working directory.)

DOC2CSI does not create a new program *.DLD file. However, when the file is retrieved into Edlog and recompiled (Edlog is invoked by pressing the Program button in PC208W), a *.DLD file will be created.

6.1.4 NOTES

There are some instances where complete program information is not converted from *.DOC files to the *.CSI files. Notes on these instances follow:

- Comments to the right of the P97 instruction for the 21X and CR7 are lost.
- Comments created with CTRL+N (whole line comments) in the middle of an instruction are lost.
- Comments to the right of the table interval are lost.
- Duplicate labels are set to no label (i.e., _____), and a warning is printed.
- Special characters (@#$%^, etc.) in labels are replaced with the underline character (_).
- DOC2CSI will use Input Location labels from the old program's *.DLD file. If a *.DLD file is not found, labels from the *.DOC file will be used. Use of these labels from the *.DOC file may result in an incomplete set of labels, since not all labels are found in the *.DOC file.

6.2 CR10 PROGRAMS TO CR10X

Though programming instructions for the CR10 and CR10X are similar, CR10 programs should be converted to ensure proper operation on a CR10X. Conversion of the CR10 programs will allow you to use the CR10X-specific options and instructions within the Edlog program.

The following steps explain how to convert a CR10 program to a CR10X program. The sequence in which these steps are performed is critical. To ensure the program is converted successfully, follow the steps as listed.

1. Open Edlog by choosing **Program** from the main PC208W toolbar. Open the existing CR10 program. The datalogger caption in the top left portion of the screen should read "CR10".

2. The first line of text in the program file should read " ;{CR10} ". Edit this text so it reads " ;{CR10X} ".

3. Choose **Save** or **Save As** from the **File** menu. When asked whether or not to compile the program, select No.

4. Close the file and then reopen it. The datalogger caption in the top left corner of the screen should now read "CR10X".

NOTE: CR10 programs created with Edlog version 5.4 or earlier must first be converted using DOC2CSI.

6.3 XLATE

Xlate is a utility which is used to convert 21X(L) programs into CR23X programs. Like the DOC2CSI utility discussed above, Xlate is not a PC208W menu option, but it has been shipped as part of the PC208W software suite. When the PC208W software is installed, Xlate is installed in the PC208W\BIN directory.

To start Xlate, double-click on the Xlate.exe file in the PC208W\BIN directory. The Xlate main window has a toolbar with three items: File, Options, and About (About provides information on the version of Xlate you are using). Use of the Xlate utility is explained below.

6.3.1 FILE MENU

To process a file using Xlate, choose **Translate** from the **File** menu. A standard Windows dialog box appears on the screen displaying all of the Edlog programming files (*.CSI) resident in the current directory. Select the file or files to be translated and choose OK. Depending upon your setup in the **Options** menu (see below), you will be prompted to enter a filename or the file(s) will be processed immediately. The translated file(s) will be saved to your current directory.

To view an Edlog programming file, choose **View File** from the **File** menu. To clear the display, choose **Close File**. The program is terminated by choosing **Exit**.

6.3.2 OPTIONS MENU

When the **Options** menu item is chosen, the following dialog box appears on the screen:

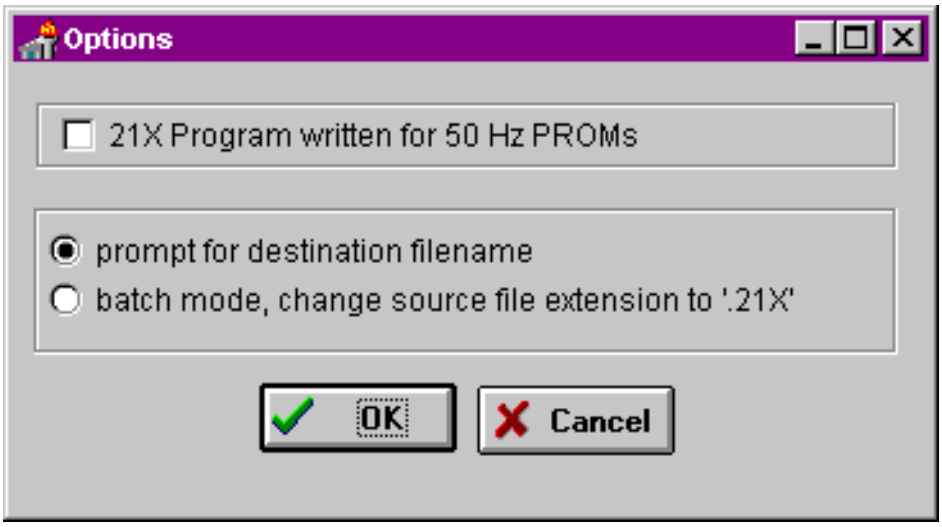


FIGURE 6.3-1. Xlate Options Dialog Box

The 21X dataloggers could be ordered with special PROMS which set the slow integration rate to 50 Hz (20 ms). Since the CR23X supports both 50 Hz and 60 Hz integration times, you must select the "21X Program written for 50 Hz PROMs" checkbox if the 50 Hz integration method is to be used.

You can toggle between the remaining two options to choose the output file type. When "prompt for destination filename" is chosen, only one file can be processed at a time, and you must enter a name for the translated file. The processed file is given a .CSI extension.

Choosing "batch mode, change source file extension to 21X", allows you to process multiple files at once. To select the files, highlight the first file and then press and hold the **Ctrl** key on your computer keyboard while you select the remaining files. When all files have been selected, choose OK. You will not be prompted for a filename. The translated files will be given the name of the original CR21X files, and the original files will be renamed with a *.21X extension (i.e., a 21X file named *site1.CSI* will be renamed *site1.21X* when the translation process is complete).

Note that your CR21X files which were given a *.21X extension must be renamed before they can be used again in Edlog. In Windows 3.1 or Windows 95/NT, locate the file in File Manager or Explorer, respectively. With the file highlighted, choose rename from the file menu (Windows 3.1) or press F2 (Windows 95/NT). This will allow you to edit the filename. Rename the file with a *.CSI extension. Make sure to choose a unique name that does not conflict with a newly created CR23X *.CSI file (Windows will not allow you to overwrite an existing file when renaming a file).

Caution! If the Xlate process will cause an existing file to be overwritten when using the "prompt for destination filename" option, you will be prompted to choose whether or not Xlate should continue. However, when using the "batch mode" option you will not be prompted and any existing files will be overwritten!

6.3.3 USING TRANSLATED FILES

Files that have been processed using Xlate can be opened and used in Edlog without modification. However, before a program can be downloaded to a CR23X datalogger, *it must be compiled*. The compilation process produces the *.DLD program that contains the actual programming code for the datalogger.

APPENDIX A. FREQUENTLY ASKED QUESTIONS

A.1 SETUP, CONNECT, AND STATUS

The default modem will not work, and my modem is not in the list of modems. How do I add it?

Select a modem similar to yours. Edit the modem name. When this field is changed, a new entry is created in the WMODEM3.INI file in the working directory. Edit the other strings as needed. The changes are saved as the fields are edited. (See Appendix B.)

How do I delete a modem entry I no longer need?

Delete the Modem Name parameter to delete the entry.

How do I select (or deselect) a non-contiguous set of Input Locations to monitor?

Select the first location or group of contiguous locations (click or drag). Then, while holding down the <Ctrl> key, select the rest. The same technique can be used to deselect locations (including the last highlighted location).

My radio network uses repeaters. How do I enter them on the DEVICE MAP screen?

Repeater only sites are not added on the **DEVICE MAP** screen. Datalogger sites that also function as a repeater are only entered once as a datalogger site (attached to the RF base). To use a repeater, enter its ID as part of the **Dialed using RF95 Path** parameter on the **HARDWARE** tab for the remote datalogger. For example, "10 32F" would call address 32 using 10 as the relay. It doesn't matter if 10 has a datalogger at the site or not.

Why do I have to change my base MD9 address to 255?

With the address at 255, PC208W can provide better information about the success or failure of an attempt to dial the MD9 network. This leads to better messages and more reliable operation. Field MD9 addresses do not have to be changed. See Appendix B.3.

I simply must have more than one network of dataloggers. How can I do it?

Normally all dataloggers can be considered as part of a single network, even if you use the same COM port for different devices. For example, you can call a remote datalogger using a telephone modem attached to COM1 and then disconnect the phone modem and connect the same COM1 directly to a local datalogger. This is still considered a single network. You just attach two devices to COM1 and realize that both cannot be used at the same time. You should remember to set the schedule for both dataloggers so they do not conflict.

What are all these log files?

Log files are included for monitoring and recording system performance and for debugging. These log files are designed to keep reasonable logs while safeguarding against an unattended log consuming all disk space.

Each active log file is written to until it reaches its maximum size. The active log files are designated by a $ character in the filename. Messages are added to these files whenever a connection is attempted or active. When an active log file reaches its maximum size, the file is renamed by replacing the $ in the name with a single digit number. If the maximum number of files already exists, the oldest one is overwritten.

SWF

This contains a log of all Warning, Fault, and Status messages. Status messages are optional and are enabled/disabled on the **STATUS WARNING FAULT MONITOR** screen. The maximum size for the active file (SFW$.LOG) is 1,000,000 bytes. In addition to the active log file, the two most recent files are kept (SWF0.LOG and SWF1.LOG).

CQR

This contains the radiotelemetry quality numbers. (See the RF manual.) The maximum size for the active file (CQR$.LOG) is 100,000 bytes and one old version is kept (CQRO.LOG).

IO$COMx

In this filename, x represents a COM port number. These files use a hexadecimal display of the actual characters sent and received by the computer. This is normally used by those familiar with the telecommunications commands or dialing of modems. This log is similar to the IO display in the Low Level I/O Monitor window. It is enabled/disabled in the **STATUS OF COMMUNICATIONS** screen. The maximum size for the active file (IO$COMx.LOG) is 1,200,000 bytes. Ten older versions are kept (IO0COMx.LOG through IO9COMx.LOG). There are logs for each COM port in use. If these log files are enabled, they should be monitored—even with restrictions on size, they can consume 13 to 14 MB per COM port.

Why can't I manually toggle the control ports on my CR10(X)?

By default, all CR10(X) control ports are configured as input unless configured as output in the datalogger program using Instruction 20. Because an input port cannot be turned off or on, if you try to toggle the port, you will receive no response and the checkbox will not accept an X.

Every time there is a telecommunications error, a message pops up even though I have minimized PC208W. How can I prevent this?

You can't. PC208W may actually try again and make the connection, but datalogger communications and data collection are considered important enough that other applications should be interrupted.

How can I get PC208W to automatically start when the computer is started?

Add the PC208W Icon to the startup folder on the Windows desktop. Windows 3.1 users need to make sure that Windows is started in the AUTOEXEC.BAT file, too. You may want to customize the icon's properties to run minimized.

A.2 EDLOG.EXE

I closed a program file and there was another file behind it.

With EDLOG it is possible for the user to open multiple files at the same time. This allows one program to be viewed while editing another or for program instructions to be cut and pasted between programs. Selecting FILE | OPEN, FILE | NEW, or FILE | DOCUMENT DLD FILE does not close any files that are already open. Select WINDOW | TILE to have EDLOG arrange all open files so they can be viewed at once.

How do I document a .dld?

Select FILE | DOCUMENT DLD FILE. Then select the .DLD file using the File Selection window.

Documenting a .DLD file automatically creates a new .CSI file. You do not need to use the FILE | NEW sequence. The .CSI file is named "Noname" until it is saved.

How do I access the Input Location number?

While the cursor is on an Input Location name (i.e., LOC name field is highlighted), type <Ctrl> and the left arrow cursor key to access the Input Location number field. Usually this is not necessary as EDLOG will automatically allocate Input Locations. Pressing <F5> brings up the Input Location Editor.

How do I add comments to the end of an instruction line?

The END key will move the cursor and place a semicolon at the end of the line. Comments can then be added.

How do I index or change the sign of a parameter?

To index a parameter, press the <F4> key. Two digit (00) and four digit (0000) parameters can also be indexed by pressing the "C" key. Negative floating point numbers are entered by typing the "-" key followed by the number.

What does redundant write mean?

EDLOG has detected that two instructions are writing to the same Input Location. Writing to the same location twice is valid but is often

done unintentionally. If writing twice intentionally, ignore the warning.

What is an "Improper use of P80" warning?

Normally P80 follows the instruction that sets the Output Flag and is used to set the Array ID or specify were data are stored. If P80 does not immediately follow the instruction setting the Output Flag, EDLOG generates this warning. Using P80 in this manner has three consequences:

- The Final Storage file that EDLOG generates may not correctly identify how the array is divided.
- EDLOG's estimates of the amount of storage used may be off.
- Some dataloggers (21X and CR7) do not support dividing arrays with different destinations.

Why can't I use CSI_R, CSI_1, ... as Input Location labels?

The expression compiler uses these locations to store temporary values. These labels are reserved. Do not use these labels in your program.

Where did my expression labels go?

When you run Optimize in the Input Location Editor, EDLOG removes any Input Location labels that are not referenced in the program or that do not have the Manual Flag set. EDLOG does not evaluate expressions as part of the optimization so locations that are only referenced by an expression will be removed. This is no cause for concern as the labels are reallocated with the next compile. The temporary labels CSI_R, CSI_1, etc. are good examples of this.

When should I optimize?

Optimization removes all labels that are not referenced in the program or do not have the manual flag set. It will move locations to remove any unused locations. It will also check and reapply the Read - Write flags. It will update all direct references (where the instructions specify an Input Location) in the program. Indirect references are not updated. You do not need to optimize, but it can be helpful if the Input Locations get cluttered after numerous program changes. If in doubt, save your work before optimizing.

What's with these "INI" files!

PC208W uses INI files, or information files, to keep track of various program configurations. There are INI files for each datalogger in your device map, INI files for your modem setup, and INI files for the PC208W program itself. Additionally, EDLOG uses an EDLOG.INI file to remember the datalogger type you used last. It also keeps track of the program that was open when you exited from EDLOG. It will reload these automatically when EDLOG is restarted.

I know the number of the instruction I want to add. How can I avoid the <Shift><Insert> or EDIT menu steps to add it?

Type the number starting in column one on a blank line in EDLOG.

How can I find an instruction when I don't remember the number?

Pressing <Shift><Ins> will display a pick list of instructions. There are two ways to find and insert an instruction:

- Use the up and down arrow keys to move through the Instruction list. When the instruction you wish to use is highlighted, press <Enter>.
- Use the <Tab> key to highlight the Instruction Name field. Type in the first few letters of the instruction name; that instruction name will be found and highlighted in the Instruction list. Press <Tab>, and press <Enter>.

Once an instruction has been inserted into the program, you can press <F1> for help on the instruction and its parameters.

Why did the dld file not change when I changed my program?

You did not recompile. Changes to the .CSI file are reflected in the .DLD file following a compile operation.

EDLOG does not allocate Input Locations for some instructions.

There are a some instructions that are exceptions to the way EDLOG allocates Input Locations and tracks Final Storage. These instructions may require the user to manually allocate Input Locations.

Instruction 49 Spatial Maximum

This instruction finds the maximum value from a "swath" of adjacent Input Locations. It is stored in the location specified in the third parameter. This instruction can also return the address (number) of the location that contained the maximum. To do this, add 1000 to the location number for parameter three. For example, the following instruction stores the maximum value, found in location 1 (Temp#1) through location 6 (Temp#6), in location 7 (Max_value).

```
10:                 Spatial Maximum (P49)
  1:      6         Swath
  2:      1         First Loc [ Temp#1    ]
  3:      7         Max Option [ Max_value ]
```

Alter the location number for parameter three as follows, store the maximum value in location 7 (Max_value) and the location address (number) where the maximum was found in location 8.

```
10:                 Spatial Maximum (P49)
  1:      6         Swath
  2:      1         First Loc [ Temp#1    ]
  3:   1007         Max Option [ Max_value ]
```

Location 8 must be named in the Location Editor (press <F5>). Use the mouse or <Ctrl>+left arrow to move the cursor to the location number while on the third parameter.

Instruction 50 Spatial Minimum

See Instruction 49 Spatial Maximum

Instruction 54 Block Move

```
1:                  Block Move (P54)
  1:      1         No. of Values
  2:   0000         First Source Loc [ _________ ]
  3:     00         Source Step
  4:   0000         First Destination Loc
                    [ _________ ]
  5:     00         Destination Step
```

This instruction copies the values in a set of locations to another set of locations. Using the STEP parameters (parameters 3 and 5) it is possible to copy from and/or store to a set of locations that are not adjacent. For example, a SOURCE STEP of 2 would copy the values from every other location. When displaying the READ and WRITE flags and when allocating and naming locations written to, EDLOG always assumes the STEP parameters are set to 1. This means the user is responsible for naming and allocating locations if either or both STEP values are not 1.

Instruction 75 Histogram

```
2:                  Histogram (P75)
  1:      1         Reps
  2:   0000         No. of Bins
  3:     00         Form Code Option
  4:   0000         Bin Select Value Loc
                    [ _________ ]
  5:   0000         WV Loc Option [ _________ ]
  6:    0.0         Low Limit
  7:    0.0         High Limit
```

Leave the fifth parameter set to zero to select a standard histogram. This will result in a compile message "Warning: zero is an invalid inloc address, Line: xx". In this case the address of zero is acceptable and the warning can be ignored. If a weighted value (WV) histogram is done, an Input Location is specified as usual.

Instruction 23 Burst measurement

This instruction makes a series of high speed measurements. These measurements are sent out the serial port or stored in Input Locations.

```
1:                  Burst Measurement (P23)
  1:     00         Input Channels per Scan
  2:     00         Range Option
  3:     00         In Chan
  4:   0000         Trig/Trig/Dest/Meas Options
  5:    0.0         Time per Scan (msec)
  6:    0.0         Scans (in thousands)
  7:   0000         Samples before Trigger
  8:    0.0         mV Limit
  9:   0000         mV Excitation
 10:   0000         Loc [ _________ ]
 11:    1.0         Mult
 12:    0.0         Offset
```

EDLOG does not detect when results are stored in Input Locations. It does not allocate or name

Input Locations. This should be done with the Block command and the Manual flag in the Input Location Editor. If results are set to be sent to the serial I/O port and no Location is specified for parameter 10, a " Warning: zero is an invalid inloc address, Line: xx" warning will occur. This warning can be ignored.

Instruction 80 Set Active Storage Area

The second parameter of Instruction 80 specifies an Array ID or an Input Location. If parameter one is set to 1 or 2, then an Array ID is expected and the cursor will move by default to the numeric field. If parameter one is set to 3, then an Input Location is expected and the cursor will move to the label field by default and the numeric field will display the address of the Input Location.

EDLOG does detect when results are stored back in Input Locations. It does not allocate or name Input Locations based on the instructions that follow Instruction 80. This should be done with the Block command and the Manual flag in the Input Location Editor.

1:		Set Active Storage Area (P80)
1:	00	Area Option
2:	0000	Array ID or Loc [_____]

Instruction 81 Rainflow Histogram

This instruction will store results in Input Locations or Final Storage. Thus, this instruction can be an Output Processing or a Processing instruction. Parameter 10 is used to specify which mode will be used. If an Input Location is specified, results will be stored starting at the specified location. In either mode, output occurs when the Output Flag is set. Set the Input Location address to 0 to store results directly in Final Storage. This will result in a "Warning: zero is an invalid inloc address, Line: xx" warning which can be ignored.

1:		Rainflow Histogram (P81)
1:	1	Reps
2:	0000	Input Location of Input Data [_____]
3:	0000	Swath (per rep) of Data at Inloc (1=online)
4:	0000	No. of Mean Bins
5:	0000	No. of Amplitude Bins
6:	0.0	Low Limit of Input Data
7:	0.0	High Limit of Input Data
8:	0.0	Min Distance Between Peak & Valley
9:	00	Option (AB)
10:	0000	Loc [_____]

EDLOG does not detect when results are stored in Input Locations. It does not allocate or name Input Locations other than the location specified in parameter 10. This should be done with the Block command and the Manual flag in the Input Location Editor.

EDLOG does not calculate Final Storage usage or Specify Final Storage Labels for this instruction.

Instruction 101 SDM-INT8

The INT8 can be used to make interval or pulse measurements and store the results in Input Locations or directly in Final Storage. EDLOG does not detect when results are stored in Input Locations. It does not allocate or name Input Locations other than the location specified in parameter 7. This should be done with the Block command and the Manual flag in the Input Location Editor.

EDLOG does not calculate Final Storage usage or Specify Final Storage Labels for this instruction.

1:		SDM-INT8 (P101)
1:	00	Address
2:	0000	Input Config: Channels 8,7,6,5
3:	0000	Input Config: Channels 4,3,2,1
4:	0000	Function: Channels 8,7,6,5
5:	0000	Function: Channels 4,3,2,1
6:	0000	Output Options
7:	0000	Loc [_____]
8:	1.0	Offset
9:	0.0	Mult

When results are stored in Final Storage, parameter 7 may be left at zero. This will result in a "Warning: zero is an invalid inloc address, Line: xx" warning which can be ignored.

Instruction 15 Serial I/O with a control port

This instruction allows the datalogger to communicate with an RS-232 type sensor. Characters can be received and/or transmitted.

EDLOG does not name or allocate Input Locations other than the location specified in parameter 10.

1:		Port Serial I/O (Special) (P15)
1:	1	Reps
2:	00	Configuration Code
3:	0000	CTS/Delay
4:	00	First Control Port
5:	0000	Output Loc [_____]
6:	0000	No. of Locs to Send
7:	0000	Termination Character
8:	0000	Maximum Characters
9:	0000	CTS/Input Wait
10:	0000	Loc [_____]
11:	1.0	Mult
12:	0.0	Offset

When characters are only transmitted, parameter 10 may be left at zero. This will result in a "Warning: zero is an invalid inloc address, Line: xx" warning which can be ignored.

How do I convert files created by the old EDLOG into new EDLOG files?

If you don't care about transferring the comments to the new file, you can simply open the new EDLOG, use FILE | DOCUMENT DLD FILE, and select the old *.DLD from the list presented. This creates a new *.CSI file based on the original *.DLD, preserving Input Location labels and memory requirements (*A settings).

If you want to preserve the comments in your old files, you can run DOC2CSI.EXE from a DOS prompt. DOC2CSI is found in the BIN subdirectory in the directory where you installed PC208W. Section 6 of this manual describes the proper syntax, or you can enter "DOC2CSI" at a DOS prompt for a simple **HELP** screen.

Keep in mind that DOC2CSI does not create a new *.DLD file. If you change the program in the new EDLOG, you must compile it to create a new *.DLD.

(See Section 2.9 of this manual for an explanation of the various files created by EDLOG.)

APPENDIX B. MODEMS

B.1 PHONE MODEMS

After adding a new Phone Modem to the Device Map, you can click on the **Hardware Tab | Edit Modem Strings** to set up specific parameters related to the modem. Information for each modem is stored in the WMODEM3.INI file in the PC208W directory. This file is updated each time a field in the Hardware page is changed.

The available fields on this set up page are: modem pick list, modem name, modem reset string, and modem initialization string. Each of these is explained below.

Modem Pick List

Common modems are available in the Modem Pick List. When a modem is chosen from the Pick List, the remaining parameter fields are automatically filled in with default values for that particular modem. These default values can be changed by editing the desired field.

If the modem you are using is not listed, you can choose <default modem> (or create a new modem—see Modem Name below) and edit the remaining parameters according to your modem's specifications.

Modem Name

A new modem can be created by editing an existing modem name or entering a new name. You can also remove an entry from the Modem Pick List by deleting its name from this field and then selecting a new modem as its replacement.

Modem Reset String

This string is sent to return the modem to a known state. This state is such that the modem is off-line and ready to receive dialing commands. Usually the string will set the modem back to factory defaults, but a string to use settings stored in non-volatile memory can be used once those settings are stored. Your modem user manual should provide the command to reset the modem to its factory defaults (often this is &F0, &F1, etc.)

Modem Initialization String

This string is used to prepare the modem for use with the dataloggers and Campbell Scientific field or voice modems. Typically, these commands enable flow control, compression, verbal result codes, and error checking.

B.2 GENERIC MODEM DIALING STRINGS

Generic modem is specified for modems requiring special dialing. Generic modem dialing has the following characteristics:

- No automatic reset string (like "ATZ") is sent.
- Actual commands sent are specified by the user.
- Dialing can be paused for specified delays or responses.

The command string or "phone number" is formulated as follows:

"xx"—The actual text to be sent is enclosed in quotes. Control characters can be sent to the modem using the ^ character. For example, ^m would produce a <Ctrl>M, which is a carriage return character. Specify ^^ to send the ^ character, and ^~ or ^> to specify a <Ctrl>^. PC208W will wait for the echo of each character sent. If no echo is received within the duration specified in the Extra Response Time (msec) parameter, PC208W continues with the next character.

T"xx"—Similar to above, except the text enclosed in quotes is sent without waiting for an echo.

Delays and responses are implemented with the following characters (not enclosed in quotes):

Dn—Delay command. This forces the preceding portion of the command string to be sent, then delays for n milliseconds. Characters received during a delay are accepted and discarded.

R“xx”n—Wait for response. This forces the preceding portion of text to be sent, then waits up to n millisecond for the quoted string xx to be returned. When the last character of the quoted string is received, the command is complete and the command string continues with the next command. If the quoted string is not received in the allowed time, the PC208W dial attempt fails and the command string is aborted.

> **NOTE:** Only the order and existence of the quoted string is checked. Other characters may be mixed in the response and will be accepted and ignored as long as the specified characters come in the correct order.

Spaces are allowed between commands and quoted strings and will be ignored.

For example: “^m” D250 “^m” D250 “ATZ^m” R“OK” 250 will send a carriage return, pause 250 milliseconds, send a carriage return, pause 250 milliseconds, send ATZ followed by a carriage return, and then will wait 250 milliseconds for OK to be returned from the modem.

B.3 MD9 MULTIDROP INTERFACE

PC208W uses MD9 dialing with the base MD9 ID (address) set to 255. Note that phone to MD9 networks already have the base MD9 address set to 255 and do not need to be changed. Addressing the base MD9 at 255 provides PC208W with a prompt from the base MD9. This provides more reliable dialing and better information about any problems that may occur while dialing.

NOTES ON USING PC208 AND PC208W

Both PC208W and PC208 can be used to communicate with the same datalogger. Data collection will be independent. If an MD9 network is used, any station file used with PC208E, GraphTerm or TELCOM will need to be changed once the base MD9 address is changed. To modify the .STN file for current DOS versions of PC208, simply add a “#” to the end of the MD9 dialing path.

For example, the device section of a station with an MD9 addressed at 1 would change from:

```
Interface Device:
#1: MD9          Addr: 1
```

to:

```
Interface Device:
#1: MD9          Addr: 1 #
```

B.4 SHORT HAUL MODEMS

Newer short haul modems (SRM5) are configured as if the datalogger were operating under a direct connection. The datalogger is added directly to the COM port in the device map, with no other devices between the two.

Older short haul modems (SC95) require a different setup. A generic modem should be added to the COM port, with the datalogger attached to the generic modem. The generic modem should have half-duplex enabled (selected on the Setup screen's Hardware tab), with all other settings at the default. The Generic Modem's dialing string field should be left empty.

> **NOTE:** If the SC95 modem setup still does not work, ensure that the modem is set up to be rung by the calling device and not manually. Refer to the SC95 users manual for more information.

B.5 DATALOGGER INITIATED COMMUNICATIONS (CALL-BACK)

The PC must be turned on, PC208W must be running and configured for Call-back, and the datalogger must be programmed to initiate communications in order for the Call-back feature to work. PC208W automatically configures the modem for auto answer when the call-back box is selected.

APPENDIX C. PC208W ERROR MESSAGES

C.1 SPLIT ERROR MESSAGES:

Parameter following ";" must be an integer.
Check the Time Series functions where an output interval has been assigned.

Invalid range specified inside [], or expression buffer overflow.
On the Select condition line you should check that the range specified has two periods (..) or that only digits are used as delimiters. Also, check for a missing right or left bracket.

Input file does not exist or is not a data file, or has the same name as the output file.
Check the data file to see that it is valid.

Parameter file is too big.
You have exceeded the maximum parameter file size of 4700 bytes.

Illegal value for time.
In the Start parameter with the Time Sync option SPLIT did not recognize the time option. Check that the time is specified properly (day:hrmn:sec). Refer to the SPLIT section for proper syntax.

Illegal array element # specified.
You have specified an element number that does not exist. Most commonly, you forgot to put a "." in a fixed value so SPLIT thinks it is an element number.

Value inside [] missing or illegal.
Check the Start, Stop, and Copy syntax to verify that you have not missed a value or entered a value that is not valid.

Too many "and" conditions in expression.
There is a maximum of three elements that can be connected with “and” statements.

Too many "or" conditions in expression.
There is a maximum of six elements that can be connected with “or” statements.

Line length too long.
253 characters maximum.

More than 8 files specified.
Check that no more than 8 Input files have been specified. Check your syntax for a filename. Remember, SPLIT will add .DAT to a filename if a "." is not included.

"." without a number.
Check the Select Line for a "." that is not associated with a fixed value.

Illegal character in expression.
Check the last line that you were editing. SPLIT would not take some character. Experiment to see what it does not like.

"(" without ")".
Check your parentheses. You must have the same number of left parentheses as you do right parentheses.

Function without "(".
Check all Math, Time Series, and Special functions for a missing left parenthesis.

Function without ")".
Check all Math, Time Series, and Special functions for a missing right parenthesis.

Illegal value inside [].
In the SELECT line check that left and right brackets have been included. Also, verify that two periods (..) have been included. Only digits can be used for the beginning and ending numbers.

Element value not in range 1..254.
SPLIT cannot work on more than 254 elements at a time. Also, one element in the Select option cannot exceed the buffer size for an element. Try shortening the length of the expression between commas.

Length of heading too large for field width.
Column heading is too wide.

Illegal Output Filename.
Output file cannot have same name as input file.

There was a problem opening an input file. File could not be found or may be in use.
Ensure a valid input file was chosen, and that the file is not opened in another program.

If multiple input files have been specified, the input file name length may have been exceeded. The input file length is 253 characters. If file names have paths defined, this limit may be exceeded. Try deleting the path from each file's name by editing the Input Data File field.

Date functions must include year.
Check the Select line for a Date function without a year.

Invalid decimal place entry. Valid values are 0 to 9.
Check the Report and Columns heading section. Up to 9 decimal places may be specified.

Selected array not found in "Copy Condition", Cannot paste.
When you're pasting an element in the Select line using Final Storage Labels, the array you are selecting from must be found on the Copy line. If the Copy line is blank, SPLIT will add it for you but you must add it otherwise.

Cannot paste there.
You can only paste on Select and Copy lines. Ensure the cursor is within one of these fields.

C.2 EDLOG EXPRESSION ERRORS

The following error codes and messages may occur when using expressions.

Code	Error Message
100	Missing left parenthesis
101	Missing right parenthesis
102	Variable name expected
103	Number expected
104	Floating point numbers limited to 5 digits
107	Function expected
110	New line expected
111	Equal sign expected

C.2.1 CAUSES

C.2.1.1 Variable Name Expected

This message occurs when the expression is not set equal to an Input Location label. The label must be to the left of the expression and not enclosed in parentheses. An expression that contains no equal sign causes compiler error 202, "unrecognized text".

For Example:

"Variable name expected" is displayed when a program contains any of these expressions:

5=el*(Vee+en)

(lambda) = COS(theta)

10-(zee/2)=bee

These are correct ways of entering the above expressions:

five=el*(Vee+en)

lambda = COS(theta)

bee=10-(zee/2)

C.2.1.2 Number Expected

Indicates one of the following situations:

(1) An expression with a /, *, or ^ operator is missing a number or label before and/or after the operator.

(2) An expression with a + or - operator does not have a number or label after the operator.

(3) An expression with an @ operator does not have a number after the @; only a fixed number is allowed immediately after the @ operator.

(4) An expression with an @ operator does not have either a number or label before the @.

(5) There is nothing between a pair of parentheses (e.g., the expression contains this "()").

(6) A number is immediately followed by a label or function without an operator (e.g., an expression containing "8label" gets this error message).

C.2.1.3 Floating Point Numbers Limited to 5 Digits

All fixed numbers are limited to five digits not including negative signs and decimal points.

C.2.1.4 Function Expected

Letters that are immediately followed by parentheses are assumed to be a function. If the letters are not on the function list (see section 2.3.3.2), this error message occurs.

C.2.1.5 New Line Expected

Indicates one of the following situations:

(1) An expression contains more than one equal sign.

(2) There is no operator between two sets of parentheses.

For Example:

This error message is displayed when a program contains any of these expressions:

zee=(label1)(label2)

ex=(5)(ARCTAN(data))

eee=(em)(see^2)

These are correct ways of entering the above expressions:

zee=(label1)*(label2)

ex=(5)*(ARCTAN(data))

eee=(em)*(see^2)

(3) There is no operator between a set of parentheses and a number.

For Example:

This error message is displayed when a program contains any of these expressions:

tee=5(2)

mu=(nu)103

bee=10.52(ef/2)

sigma=-17(RCP(alpha))

These are correct ways of entering the above expressions:

tee=5*(2)

mu=(nu)*103

bee=10.52*(ef/2)

sigma=-17*(RCP(alpha))

(4) A label or function is immediately after a set of parentheses without an operator.

For Example:

This error message is displayed when a program contains any of these expressions:

result=(ex^2)data

gamma=(10-omega)SIN(psi)

dee=(17)number

These are correct ways of entering the above expressions:

result=(ex^2)*data

gamma=(10-omega)*SIN(psi)

dee=(17)*number

C.2.1.6 Equal Sign Expected

An equal sign MUST immediately follow the label of the Input Location that stores the results (e.g., label = expression). An expression that contains no equal sign causes compiler error 202, “unrecognized text”.

For Example:

“Equal sign expected” is displayed when a program contains any of these expressions:

zee/2=bee

data+number=volt1+volt2

These are correct ways of entering the above expressions:

bee=zee/2

data=volt1+volt2-number

APPENDIX D. CARD SERVICES DRIVER SOFTWARE NOTES

SMS comes with its own setup program for PC Cards. When using SMS for the first time with PC Cards, you should run this setup program by clicking the ***PC Card Setup*** *button on the* ***PC CARD USER*** *screen within SMS—see Section 5 this manual (Establishing Communications).*

This Appendix gives additional specific information about using Card Services drivers and software with SMS, and is intended to be supplementary to the documentation shipped with your Card Services drivers.

D.1 GENERAL

To use the PC Card option of SMS you *must* have a PC Card (previously called PCMCIA) slot and an implementation of Card Services drivers compliant with PCMCIA Card Services Standard 2.1 or PC CARD Standard 3.0 or greater.

Card Services is a generic term for a set of drivers written to an agreed standard, allowing your computer to access a variety of physical PC Card slots. You will need at least two sets of drivers: Socket Services and Card Services. They are almost always supplied together. Card Services drivers will not work without Socket Services drivers being present.

D.2 DRIVER VERSION NUMBERS

Card Services drivers version 2.1 or greater are required. Some implementations of Card Services are known to have at least *three* version numbers, one relating to the version of the standard to which they claim to be compliant, one relating to the software release and a further version number relating to something unknown. This can lead to confusion when trying to establish the correct number to quote. The important number is the one for the standard with which the driver claims to comply. This number must be 2.1 or greater.

> **NOTE:** If the version number of your drivers is earlier than 2.1 you *must* upgrade them. Card Services version 2.1 is incompatible with Card Services version 2.0 or previous versions. Any programs written for version 2.0 will *not work* with version 2.1.

SMS has been tested with the following drivers:

- SystemSoft's CardSoft CardWizard under Windows 3.1
- Award's CardWare under Windows 3.1 and Windows 95
- Phoenix PCM+3.0 under Windows 3.1

Please contact the original supplier of your drivers for any upgrades, not Campbell Scientific.

D.2.1 MICROSOFT WINDOWS

Windows 3.1

Windows 3.1 does not come with any built-in support for PC Cards and some older implementations of Card Services do not perform properly under Windows 3.1. However, it is possible to run the program under Windows using these older implementations by using the VMMAPPER.EXE file. This file is automatically installed by the "Setup PC Card" facility built into SMS; therefore, PC Setup should *always* be used to set up your computer before attempting to use PC cards with SMS. (See "Establishing Communications" in Section 5 of this manual.)

Windows 95

Windows 95 does provide some support for PC Cards, and provides links to real mode Card Services via the CSMAPPER driver. Note that VMMAPPER is also required. It is therefore possible to use PC Cards with Windows 95 without the need to purchase extra Card Services software. You *must* ensure that the PC Card support is enabled under Windows 95—see the Windows Help System for details.

Phoenix and SystemSoft have released Card Services drivers for Windows 95; these can be used if required. It may also be possible to

configure Award's CardWare 2.1 (for Windows 3.1) to work satisfactorily with Windows 95.

Always use the "PC Card Setup" facility included in SMS before using PC cards with SMS. (See "Establishing Communications" in Section 5 of this manual.)

D.2.2 CSL-CS.DLL

CSL-CS.DLL provides a common interface for SMS to the different implementations of card services.

CSL-CS.DLL creates an initialization file, CSL-CS.INI, in the Windows directory. You should not need to manually modify this INI file, since PC Card Setup automatically modifies this so that the DLL works correctly.

D.3 SETTING UP CARD SERVICES

> **NOTE:** SMS is provided with its own Card Services setup program. Before using SMS the first time with PC Cards, you should run this program. It is accessed from the **PC CARD** main screen by clicking on the **Setup PC Card** button. (See Section 5 of this manual—"Establishing Communications".)

Although the SMS PC Card Setup program may be all that is required for successfully setting up and running PC Cards with SMS, the following notes are provided for a more detailed understanding of the way PC Cards are used. These notes are intended as a supplement to the documentation shipped with any PCMCIA drivers you may have.

For some Card Services without MTD support, SMS needs a memory range of 32K allocated to Card Services. It uses the 32K memory to read and write to the card, while newer versions of Card Services configure this for you automatically, some older versions require the user to set this up. Notes are given below on how to set up some older versions of Card Services.

Card Services has the ability to map a region of the PCMCIA card to PC address space in the high-memory area. This is the same address space which is mapped to extended memory by EMM386 for programs loaded high with DEVICEHIGH or LOADHIGH. If all of this address space is allocated, then Card Services will not be able to map regions to SRAM cards, and SMS will fail with an error. The following information explains what to do if you have this problem.

Card Services does not use the actual memory in the high memory area, but it does use that address space. There is no memory in that address space. If you have EMM386 loaded (which you *will* have if you load other drivers, etc., into high memory) you must exclude an area of memory for Card Services use.

The line in your CONFIG.SYS which sets up EMM.386 will look something like:

```
DEVICE=C:\DOS\HIMEM.SYS
DEVICE=C:\DOS\EMM386.EXE NOEMS
```

If, after studying other memory requirements and memory use such as the address space used by your video card (the program MSD, supplied with MSDOS 6 can be used for this purpose), you decide that the memory area D0000 to DFFFF may be used by Card Services, then change your lines as shown below.

```
DEVICE=C:\DOS\HIMEM.SYS
DEVICE=C:\DOS\EMM386.EXE NOEMS /X=D000-DFFF
```

(Note that there is one 0 and one F missing from the memory address you enter in CONFIG.SYS. This is because EMM386 expects a segment address. Each segment contains 16 bytes, 0-15 (0-F), and so the beginning address in the segment D000 is D0000 and the ending address of the segment DFFF is DFFFF.)

Some graphics cards require memory to be excluded in the same way. More than one region of memory may be excluded by repeated use of the /X= option.

The EMMExclude line in SYSTEM.INI should reflect the EMM386 exclude range. For example:

```
[386enh]
EMMExclude=D000-DFFF
```

After setting aside this memory for use by Card Services, you may need to configure Card Services itself to use this region. Please consult the documentation provided with your specific Card Services drivers.

D.3.1 HOW TO CHOOSE WHICH AREA OF MEMORY TO ALLOCATE TO CARD SERVICES

The address range used by Card Services can be from A0000 to FFFFF. However not all machines will allow Card Services to use these addresses. Many machines cannot use addresses below C8000 or above E8000 for Card Services. If you are just extending the region of memory allocated to Card Services, it may be best to leave the start of the region where it is, as you know your machine is capable of using memory from that address.

Excluding a region of memory for use by other programs does not mean that Card Services will be aware of it and be able to use it. The various implementations of Card Services come with their own .INI files, that specify the regions of memory Card Services will use.

Many implementations of Card Services also have setup programs which automatically detect the free memory and create the .INI file accordingly. Often such programs allow you to specify and modify the memory range to be used by card services, to prevent Card Services from using a region which was excluded for use by some other device such as a graphics card or network card.

A problem which sometimes occurs when configuring memory ranges is that the setup program forgets which EMM386 exclude ranges were set for Card Services, and so each time the setup program is run, greater regions are excluded from EMM386. Because of this, it is advisable to keep a backup copy of your CONFIG.SYS file until after you have finished making changes.

If your system is already set up in accordance with the above (and as described in the Card Services supplier's instructions) and SMS still fails due to lack of resources, you may need to increase the size of your memory range. A range of 64K (example: D0000-DFFFF) should be plenty. Unless you have many other specialized PCMCIA drivers you will not need to increase the size of this range.

Please refer to the instructions which came with your Card Services driver software and your Microsoft Windows manual for help in optimizing and setting up your particular system.

APPENDIX E. GLOSSARY OF COMMON WINDOWS TERMINOLOGY

Cancel Choosing Cancel from a dialog box will ignore any changes made and close the box.

Click To Click on an object, press and release the left mouse button while the mouse pointer is positioned on the object.

Clipboard The Windows Clipboard is a buffer which holds information from a Cut, Copy, or Print Screen operation. Only the last operation is held in the Clipboard's memory. A new Cut or Copy operation will replace the contents of the Clipboard.

Close To close a window, most often you can choose Exit from the File menu, or, in Windows 95 and NT, click on the **X** from the objects located at the uppermost right of the window.

Copy When text is "Copied" a duplicate of the highlighted text is moved to the Windows Clipboard. The copied item can then be Pasted to another location. To Copy an item, first highlight it with the mouse. Then, the Copy operation can usually be chosen from an Edit menu or by pressing the **Ctrl** key and **C** key simultaneously.

Cut When text is "Cut", the highlighted text is deleted from the screen and moved to the Windows Clipboard. It can then be Pasted to another location if desired. To Cut an item, first highlight it with the mouse. Then, the Cut operation can usually be chosen from an Edit menu or by pressing the **Ctrl** key and **X** key simultaneously.

Double Click To Double Click on an object, press the left mouse button twice rapidly while the mouse pointer is positioned on the object. Double Clicking is often a shortcut for a common operation. For example, Double Clicking on a file name in a file dialog box is a shortcut for selecting the file name and then selecting the OK button.

Drag and Drop Drag and Drop is used to move an object from one place to another. First, highlight the text or object to be moved. Press and hold the left mouse button while the mouse cursor is over the highlighted object. Move the mouse; the object will be "Dragged" along with the movement. When at the desired location, release the left mouse button.

ESC Pressing your computer keyboard's **ESC** key will usually exit from the current menu or window and return you to the previous level.

F1 In most instances, pressing the **F1** key will invoke your program's help system.

Folder Windows uses graphical folders to represent DOS directories on your computer. Double Clicking a folder icon will "open" it to display the files contained within.

Highlight Text can be highlighted by positioning the cursor where you want the highlight to begin, holding the left mouse button, and dragging it across the words or group of objects to be highlighted. A single object can be highlighted by clicking it once with the left mouse button. Highlighted items can then be Cut, Copied, moved, opened, etc.

APPENDIX E. GLOSSARY OF COMMON WINDOWS TERMINOLOGY

Maximize You can restore a Minimized program to its original size by Maximizing it. If you are running Windows 3.1, double click on the icon. If you are running Windows 95 or NT, click once on the item's name at the bottom of your screen.

Minimize To Minimize a program press the — box from the objects located at the uppermost right of the window. When you Minimize a program, it is still running but appears on your screen as a small icon (Windows 3.1) or as an entry at the bottom of your screen (Windows 95 or NT).

OK Choosing OK from a dialog box will incorporate any changes made or perform the desired action and close the box.

Paste After text is Cut or Copied, it can be Pasted to another location. The Paste command can be executed by choosing it from an Edit menu or by pressing the **Ctrl** key and **V** key simultaneously.

Print Scrn Pressing your computer keyboard's **Print Scrn** key will produce somewhat different effects, depending upon your operating system. However, in most instances pressing the **Shift** key and **Print Scrn** key simultaneously will send a copy of the entire monitor display to the Windows Clipboard, and pressing the **Ctrl** key and **Print Scrn** key simultaneously will send a copy of the entire monitor display to the printer.

Tab Key Using the **Tab** key will move the cursor to the next parameter, button, or object on the screen. Pressing the **Shift** key and **Tab** key simultaneously will move the cursor to the previous item.

Tab Windows Many Windows applications represent a series of related windows in a graphical file folder tab format. When you click at the top of a file folder tab, the information on the tab you chose will be displayed.

GLOSSARY

CALLBACK - Telecommunication with a datalogger where communication is initiated by the datalogger—usually in response to an alarm or event. Callback is also used where it is impractical to call the datalogger but the datalogger can call out.

Final Storage - A more permanent portion of datalogger memory where processed Input Location values are stored. The values are typically the result of processing measured values over time, like averages, maxima, etc., over the last minute, hour, day, etc. These values are retrieved for archival and analysis. The values remain in the datalogger until it is full at which time the oldest values are written over as new values are stored.

Final Storage Area 1 or Area 2 - The CR10(X), CR10, and CR23X have two independent Final Storage Areas. Use of the second area is determined by the datalogger program. Data can be collected from either or both areas.

Input Locations - The addresses in the datalogger memory where the direct results of measurements are temporarily held. Typically these are the values viewed in Monitor or Graph Mode to determine current status. These values are not necessarily the same values retrieved when data is collected.

Network - A network refers to one or more dataloggers and the communication device used to communicate with them. It should not be confused with a computer local area network.

Node - Another name for a datalogger or modem in a network.

Task - An entry on the DEVICE MAP representing a program scheduled for execution. Scheduling of execution can be done on a regular interval or following a call to a datalogger.