

**CSM1 CARD STORAGE MODULE  
AND  
CSMCOM AND SMCREAD SOFTWARE**

**REVISION: 3/01**

**COPYRIGHT © 1993-2001 CAMPBELL SCIENTIFIC, INC.**



# Warranty and Assistance

---

The **CSM1 Card Storage Module** is warranted by CAMPBELL SCIENTIFIC, INC. to be free from defects in materials and workmanship under normal use and service for thirty-six (36) months from date of shipment unless specified otherwise. Batteries have no warranty. CAMPBELL SCIENTIFIC, INC.'s obligation under this warranty is limited to repairing or replacing (at CAMPBELL SCIENTIFIC, INC.'s option) defective products. The customer shall assume all costs of removing, reinstalling, and shipping defective products to CAMPBELL SCIENTIFIC, INC. CAMPBELL SCIENTIFIC, INC. will return such products by surface carrier prepaid. This warranty shall not apply to any CAMPBELL SCIENTIFIC, INC. products which have been subjected to modification, misuse, neglect, accidents of nature, or shipping damage. This warranty is in lieu of all other warranties, expressed or implied, including warranties of merchantability or fitness for a particular purpose. CAMPBELL SCIENTIFIC, INC. is not liable for special, indirect, incidental, or consequential damages.

Products may not be returned without prior authorization. To obtain a Returned Materials Authorization (RMA), contact CAMPBELL SCIENTIFIC, INC., phone (435) 753-2342. After an applications engineer determines the nature of the problem, an RMA number will be issued. Please write this number clearly on the outside of the shipping container. CAMPBELL SCIENTIFIC's shipping address is:

**CAMPBELL SCIENTIFIC, INC.**

RMA# \_\_\_\_\_

815 West 1800 North

Logan, Utah 84321-1784

CAMPBELL SCIENTIFIC, INC. does not accept collect calls.

Non-warranty products returned for repair should be accompanied by a purchase order to cover the repair.



## **CAMPBELL SCIENTIFIC, INC.**

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

Campbell Scientific Canada Corp.  
11564 -149th Street  
Edmonton, Alberta T5M 1W7  
CANADA  
Phone (780) 454-2505  
FAX (780) 454-2655

Campbell Scientific Ltd.  
Campbell Park  
80 Hathern Road  
Shepshed, Loughborough  
LE12 9GX, U.K.  
Phone +44 (0) 1509 601141  
FAX +44 (0) 1509 601091



## **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 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 a 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 a 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 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 no 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](http://www.campbellsci.com)

Campbell Scientific Canada Corp.  
11564 -149th Street  
Edmonton, Alberta T5M 1W7  
CANADA  
Phone (780) 454-2505  
FAX (780) 454-2655

Campbell Scientific Ltd.  
Campbell Park  
80 Hathern Road  
Shepshed, Loughborough  
LE12 9GX, U.K.  
Phone +44 (0) 1509 601141  
FAX +44 (0) 1509 601091

# ***CSM1 Card Storage Module***

## ***Table of Contents***

---

<b>1. Introduction .....</b>	<b>1</b>
<b>2. Specifications .....</b>	<b>2</b>
2.1 Datalogger Compatibility .....	2
2.2 Card Types Supported.....	3
2.3 Compatibility with SM192 and SM716 Storage Modules .....	3
2.4 Operating Specifications .....	3
<b>3. Getting Started.....</b>	<b>4</b>
<b>4. Storing Data .....</b>	<b>6</b>
4.1 CR10 .....	6
4.2 21X/CR7 .....	8
4.3 DSP4 .....	9
4.4 Data Retrieval.....	9
<b>5. Storing Datalogger Programs.....</b>	<b>9</b>
5.1 Transferring Programs to/from Datalogger.....	10
5.2 Transferring Programs to/from DSP4 .....	10
<b>6. Selected Operating Details .....</b>	<b>11</b>
6.1 Inserting the Card in the CSM1.....	11
6.2 Response of CSM1 on Detection of Full Card Memory .....	11
<b>7. Telecommunications .....</b>	<b>12</b>
7.1 Connecting the CSM1 to non-IBM computers.....	12
7.2 Establishing Communications.....	13
7.3 Breaking Communications.....	13

## **Appendices**

<b>A. 9-Pin Storage Module Interface.....</b>	<b>A-1</b>
<b>B. Internal Data Formats and Pointers in the Card     Memory .....</b>	<b>B-1</b>
B.1 Byte Pairs.....	B-1

B.2 File Structure ..... B-1

B.3 Program Files..... B-1

B.4 Data Pointers..... B-1

**C. Card Batteries .....C-1**

    C.1 General..... C-1

    C.2 Battery Status Detection ..... C-2

**D. Telecommunications Commands .....D-1**

**E. SC532A .....E-1**

***Figures***

1. CSM1 Case..... 2

2. Connection of the CSM1 to PC ..... 5

***Tables***

1. Status Indicator Flashing Sequence..... 5

2. Sample CR10 Program Using P96 to Send Data to CSM1 ..... 7



# CSM1 Card Storage Module

---

*The Card Storage Module system consists of a microprocessor-controlled read/write module (the CSM1) and removable credit card sized memory cards that hold data and/or datalogger programs in battery-backed memory. The memory cards are easily exchanged and transported to a computer for data retrieval.*

*The module is quickly prepared for use, and features built-in status indicators for module operation and data storage. The CSM1 can be left with the datalogger and the memory cards exchanged at regular intervals, or it can be moved from datalogger to datalogger to retrieve data from each one in turn. The low quiescent power consumption and wide operating temperature range allow the module to be used in remote battery-powered applications.*

## 1. Introduction

Data stored in a card is separated into files. The files are stored sequentially in the card and are segregated by filemarks. A filemark is written into the card when it is plugged into a CSM1, or can be written under software control by the datalogger or computer. This allows data from different dataloggers or different experimental runs to be separated.

Datalogger programs can also be stored from and loaded into a datalogger using the CSM1. (21X and CR7 dataloggers need recent versions of their operating software; see section 2.)

Two software programs are available for transferring data into an IBM-compatible PC and storing datalogger programs created on the PC in the card. SMCREAD is used when the computer has a built-in PC card slot. CSMCOM is used to transfer data from the CSM1 or MCR1 via the computer's RS232 port. The CSM1 can be connected to the RS232 port via a SC532(A) (Figure 2). The MCR1 Memory Card Reader has an RS232 port for direct cable connection to the Computer.

Alternatively you can develop your own programs to communicate with the CSM1. Simple ASCII telecommunications commands can be sent to the CSM1 to extract data and load programs.

The CSM1 unit is housed in a compact aluminum case (see figure 1). The card slot and connector are designed so that the card cannot be inserted the wrong way round. The card pushes in and locates with a positive click, and a push-button eject mechanism releases the card from the slot.

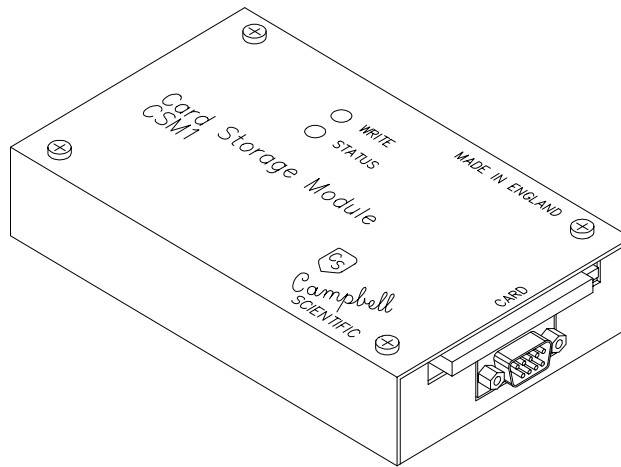


FIGURE 1. CSM1 Case

Below the card slot is a male 9-pin D connector. This is a Campbell Scientific serial port. An SC12 cable (supplied) is used to connect this port to the datalogger (for data storage) or SC532(A) interface (for data retrieval). The card slot and 9-pin connector are positioned together to allow the CSM1 to be mounted against the side wall of small enclosures, yet still allow the card to be removed, or the unit to be unplugged. By removing the electronic circuit boards inside the CSM1 the unit can be fixed to mounting plates, using the two screw holes in the base of the case.

Visible on the top of the CSM1 are two light-emitting diodes (LEDs). The red 'Status' LED indicates the status of the module and card when the CSM1 is first powered up. The green 'Write' LED indicates when the CSM1 is writing to the card.

## 2. Specifications

### 2.1 Datalogger Compatibility

The CSM1 can be used with the CR10, 21X, CR7 and DSP4. (The BDR320 is not currently supported.) The CSM1 emulates the SM192/716 Storage Modules for data storage (9600 baud, binary). Data can also be stored at 9600 baud in ASCII formats, but this uses more card memory. The CSM1 also supports data storage at 76800 baud with the CR10 Burst Mode instruction.

Program storage and retrieval are supported for the CR10, 21X, CR7 and DSP4. The 21X and CR7 must be fitted with OSX- or OS7- software respectively.

Where the datalogger software supports it, a program stored in program area 8 in the memory card is automatically loaded into the datalogger on power-up.

## 2.2 Card Types Supported

The CSM1 supports JEIDA 4, PCMCIA standard memory cards. Sizes of 256K to 2M are supported (256 bytes reserved for system use). Low resolution data format requires two bytes per data value with high resolution data occupying four bytes. Datalogger programs require the space as stored on disk, plus an overhead of approximately five bytes.

Standard card sizes are 256K, 1M and 2M. These equate to low resolution data capacities of 130944, 524160, and 1048448 locations respectively. (Please contact Campbell Scientific to check on the preferred card size and availability.)

### NOTE

CSM1 operation cannot be guaranteed with cards not tested and supplied by Campbell Scientific.

## 2.3 Compatibility with SM192 and SM716 Storage Modules

Unlike the SM192/716 Storage Modules the CSM1 does not support \*9 Mode commands for the CR10 datalogger. Also it does not support remote communications via a CR10 datalogger.

The module address is fixed at 1. Therefore only one CSM1 can be connected to a CR10. However, additional SM192/716 modules can be attached at the same time as a CSM1, providing their addresses are not 1.

## 2.4 Operating Specifications

Operating temperature:	-40°C to +50°C
Battery type: <sup>1</sup>	Lithium coin cell.
Typical battery life: <sup>2</sup>	7 years @ -20°C 8 years @ +20°C 1 year @ +60°C
Typical current (@ 25°C):	
Quiescent: <sup>3</sup>	< 0.5 mA
Active:	< 18 mA
Interface type:	9-pin D Campbell Scientific. Connection to datalogger using SC12 cable (supplied).
Baud Rate:	
Accepting data:	9600 baud.
Telecommunications: <sup>4</sup>	Supports all standard rates in the range of 300 to 38400 baud.
Memory configuration:	Fill and stop only. Data and programs delimited into files using filemarks.
Dimensions:	
Card:	85 x 54 x 3 mm
CSM1:	155 x 90 x 32 mm

Weight:

Card:	30 g
CSM1:	350 g

Construction: Anodized aluminum case. Two LEDs in case top indicate power-up status and data write operations.

<sup>1</sup>Type and capacity varies with card manufacturer and card size. Capacity 120 to 170mAh.

<sup>2</sup>These figures apply when the card is disconnected from an external power supply, i.e. the card is not plugged into the module and/or the module is not being powered by a datalogger. Low battery detection indicates approximately 3% battery life remaining. The CSM1 blocks attempted data storage when the battery is exhausted. Please see Appendix C for further details.

<sup>3</sup>Less than 200  $\mu$ A for the module plus the standby current of the card (110  $\mu$ A typical for a typical 1 M card). The standby current of the card increases with card size, although not always in direct proportion.

<sup>4</sup>Serial data format for telecommunications is one start bit, eight data bits, no parity and one stop bit. Potential data read speed using CSMCOM on a 25MHz 386 PC, at 38400 baud, (to hard disk in comma delineated format) is 1500 data values per second.

## 3. Getting Started

1. When you first receive a Card Storage Module please check you have been supplied with an SC12 cable and the memory card that you ordered.
2. New memory cards are supplied without the battery installed to prevent premature discharge in transit or storage. Install the battery, following the small instruction leaflet supplied with the card.

Be sure to insert the battery with the correct polarity. Also check that the write protect switch (if fitted), is not set in the 'protect' position.

3. Because the battery was not fitted, the card memory will be corrupted and the card will need to be erased and reformatted before you can use it. First install CSMCOM or SMCREAD from the disk supplied, following the instructions in the manual. (If you do not intend to use CSMCOM or SMCREAD, see section 7.)

Steps 4-6 are for computers without a PC cardslot (CSMCOM).

4. Connect the CSM1 to your PC. To do this you need either an SC532(A) interface or a PC201 card with an SC209 cable (see PC201 manual for installation details). If using an SC532(A), connect it to one of the free serial (COM) ports of your PC using either an SC25PS cable or 7026 9 to 25 pin adapter, which plugs into the 25-pin connector on the SC532(A). Make sure the SC532(A) is plugged into a power source and the PC is running. Plug the SC12 cable into the 9-pin port of the SC532(A).

5. Plug the memory card into the CSM1. The card should be inserted with the end with the small connector holes placed into the connector. It should not be possible to insert the card the wrong way round. Push the card in until you feel a positive click and the small button next to the card pops out.
6. Plug the SC12 cable from the SC532(A) (or SC209 cable if using a PC201 card) into the 9-pin connector on the end of the CSM1. Observe the status LED on the top of the case. After a short delay (up to 3.5s) this should flash a number of times to indicate the status of the module and card. The LED flashes for 0.5s, with a wait period of 0.5s between flashes. Refer to table 1 for a full description of the status indication. For a corrupt/unformatted card it should flash four times. If the LED does not flash, check all power and cable connections.

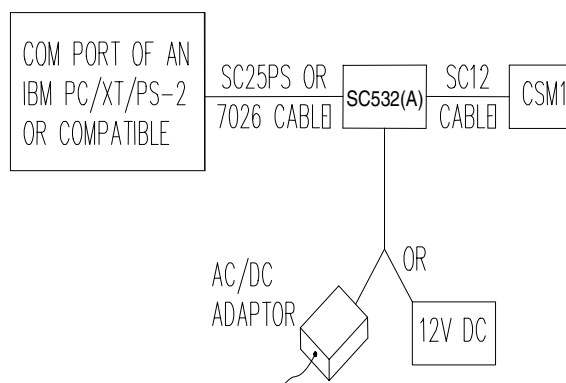


FIGURE 2. Connection of CSM1 to PC

TABLE 1. Status Indicator Flashing Sequence	
No. of Flashes	Indication
1	Card Storage Module and card OK
2	EPROM failed; contact Campbell Scientific
3	Card not plugged into module
4	Card corrupted or unformatted
5	Card battery DEAD — CSM1 will not store data
6	Card write-protected
7	Card full warning (see section 6.2 for detailed explanation)

7. Run CSMCOM or SMCREAD. When the menu appears use option E to erase and test the card. This tests and reformats the card.
8. Once the erase process is finished the card is ready to be used. Quit the program and unplug the CSM1.

**NOTES**

- 1) While performing the power up tests, the 'Write' LED will normally flash for a short period, before the status indicator flashes. This is caused by the CSM1 checking that it is able to write to the card.
  - 2) It is important that the SC12 connector is plugged into the CSM1 and also into the SC532(A) or datalogger, with the connector held perpendicular to the end of the case. If you force the connector in at any other angle, the CSM1 may not perform its power-up status display, although normally it will still be able to store data. However, if the CSM1 does not flash the status LED on power-up, it is advisable to unplug, wait five seconds and plug it back into the datalogger to ensure there has been a good connection.
  - 3) If you experience problems in communicating with a CSM1 when using a PC201 card, please refer to Appendix B of the CSMCOM manual for possible address problems and Appendix A of this manual for details of possible problems with the PC201 power supply.
- 

## 4. Storing Data

The CSM1 emulates Campbell Scientific's SM192/716 Storage Modules for most aspects of data storage. Therefore the same basic programs and procedures are used to store data in the CSM1. Details of how to store data for each datalogger type are given below.

### 4.1 CR10

Write your program as normal, including instructions to store data to Final Storage. Include Instruction 96 to store data to a Storage Module, after the output instructions. The CSM1 has a fixed address of 1 so the first and only parameter for Instruction 96 is 71 (see Table 2 for an example program).

Under the control of the program shown in Table 2, the CR10 makes a Module temperature measurement (in °C) and a thermocouple measurement (converted to °F) every 5 seconds. Every hour, the output flag is set and the Day, Hour and Minute as well as the hourly averages for the two temperatures are sent to Final Storage. Following this, the same data is sent to the CSM1, if present.

If the CSM1 is not present, the data is backlogged for later transfer.

If you are using older versions of EDLOG to develop your program, entering the code 71 for Instruction 96 will only display SM192/SM716 as the output option. Using parameter 71 will still work with the CSM1.

**TABLE 2. Sample CR10 Program Using P96 To Send Data To CSM1****\*Table 1 Programs**

01: 5	Sec. Execution Interval
01: Module Temperature (P17)	
01: 1	Loc [:MOD TEMP ]
02: Thermocouple Temp (DIFF) (P14)	
1: 1	Rep
2: 1	2.5 mV slow Range
3: 1	IN Chan
4: 1	Type T (Copper-Constantan)
5: 1	Ref Temp Loc MOD TEMP
6: 2	Loc [:TCTEMP ]
7: 1	Mult
8: 0	Offset
03: If time is (P92)	
1: 0	minutes into a
2: 60	minute interval
3: 10	Set high Flag 0 (output)
04: Real Time (P77)	
1: 110	Day,Hour-Minute
05: Average (P71)	
1: 2	Reps
2: 1	Loc MOD TEMP
06: Serial Output (P96)	
1: 71	SM192/SM716/CSM1
07: End Table 1	

If the CR10 executes Instruction 96 when a CSM1 is not plugged into the datalogger, the data is not lost. As long as the CR10's memory is not overwritten, it remembers which data it last stored to the CSM1. Each time the CR10 executes Instruction 96 it checks for the presence of the CSM1 and outputs data if the CSM1 is connected and able to store data.

Using this feature you can use the CSM1 to collect data from a CR10 programmed with Instruction 96, without the need for a keyboard or any knowledge of the operation of the datalogger. To collect data simply plug in the CSM1. The 'Write' LED flashes briefly and the 'Status' LED flashes once to indicate that the CSM1 is ready. You then wait until Instruction 96 is executed. When this happens the CR10 detects the CSM1 and sends data to it; the 'Write' LED illuminates as data is written to the card. The 'Write' LED only goes out when the CR10 has finished writing all the data destined for the Storage Module. When it does go out you can disconnect the module, having collected all available data.

You can also initiate data transfers to the CSM1 manually using the \*8 Mode commands, again specifying the destination code as 71 (Storage Module with

an address of 1). For Instruction 96 it is also possible to write a filemark into the memory card, to effectively close an open file, by pressing 'C' after the address (displayed as '71--').

The CSM1 also supports data storage at 76800 baud, when output from the CR10 with the Burst Mode instruction. To do this specify the output code for 'Serial port, 76800 baud to SM192/716 Storage Module' (see CR10 manual for further details).

---

**NOTE**

The CSM1 does not support the \*9 commands with the CR10.

---

## 4.2 21X/CR7

If your 21X/CR7 has a version of OSX/OS7 software respectively you should store data by including Instruction 96 after the data storage instructions, with the device code 30 as the first and only parameter of Instruction 96.

With OSX/OS7 software the datalogger checks for the presence of the module before outputting data. If the module is not connected the datalogger does not output data but remembers which data has dumped to the module. Thus you can leave a datalogger without a module connected and collect the data at a later date, without touching the keyboard, as outlined above for the CR10.

Manual dumps of data to the module are done using the \*9 Mode, entering 30 for the device code. Both \*9 and Instruction 96 can also be used to write a filemark into the card by specifying a device code of 31.

If you do not have OSX/OS7 software we recommend you consider upgrading your datalogger software. However, it is still possible to store data to the card as follows:

### 4.2.1 21X With ESX Software and CR7 With '700X' Control Module

Use Instruction 96 to store data in binary format at 9600 baud, i.e. device code 22. Do not use the \*4 Mode to enable printer output.

---

**CAUTION**

The operating system treats the module as a printer and does not check to see whether it is connected. Therefore if the CSM1 is not connected data can be lost.

---

Manual dumps can only be done at 9600 baud, in ASCII format, using the \*9 Mode (see below).

### 4.2.2 21X Without Extended Software and CR7 With '700' Control Module

Output must be enabled using the \*4 Mode to enable output at 9600 baud to the printer. Data is transferred and stored in the module in printable ASCII format. The CSM1 makes no attempt at encoding the data, so each data value occupies approximately 10 bytes (five locations) of memory in the card, as opposed to two bytes (one location) for low resolution data in binary format.



**CAUTION**

---

As with ESX software, the datalogger does not check for the presence of the CSM1, so if the module is not connected data could be lost.

---

Manual dumps to the module are done by setting the baud rate parameter of the \*4 Mode to 9600 baud. Data is then dumped using the \*9 Mode.

When reading data that has been stored in the card in ASCII format using CSMCOM, you should select the format 'As stored (8-bit data)'. This stores the data, as it is in the card, in printable ASCII format onto disk. You cannot convert such data to comma-delineated data using CSMCOM.

## 4.3 DSP4

DSP4s that support storage of data to a Storage Module can be used by enabling output at 9600 baud in binary (9600B). The CSM1 is used in the same way as an SM192/716 Storage Module.

**CAUTION**

---

The DSP4 does not check if a Storage Module is plugged in when it outputs data, so if the module is not connected data will be lost.

---

**NOTE**

---

When using pre-OSX 21X dataloggers, pre-OS7 CR7 dataloggers and DSP4s you should allow the CSM1 to finish its power-up test routines, which take about five seconds, before allowing output of data to the CSM1, as data may be lost during the power-up test.

---

## 4.4 Data Retrieval

Once data is stored in the card, the card or whole module can be unplugged and the data read off the card. This is normally done using the CSMCOM program, on an IBM-compatible PC (see CSMCOM manual for full details). As an initial trial of reading data, connect the CSM1 to the PC, run CSMCOM and use the N option to collect the newest data file.

If data has been stored in the card in binary format, as is normal when using Instruction 96, you can specify whether to write the data to disk in comma delineated or printable ASCII format. If data was stored in the card in printable ASCII format (on old 21X/CR7 dataloggers) or in Burst Mode binary format, you should store the data to disk using the 'As stored (8-bit)' option.

## 5. Storing Datalogger Programs

Datalogger programs can be stored in the cards for subsequent loading into a datalogger. The programs are stored in the card using the same memory available for data storage, i.e. there is no reserved area in which programs are stored. When loaded into the card the programs are put into special files in the

next available memory in the card. Every two bytes of a .DLD file loaded into the card reduces the amount of space for data storage by one location.

## 5.1 Transferring Programs to/from Datalogger

CR10 dataloggers, 21X dataloggers with OSX software and CR7 dataloggers with OS7 software can be instructed to transfer a program between the datalogger's memory and one of the eight program areas of the CSM1 memory card. This is done using the \*D Mode:

1. Connect the CSM1 to the datalogger and using the keyboard (CR10KD for the CR10), enter \*D. Enter 71A in response to the '13:' prompt, to address the CSM1.
2. To store a program in the card enter 1nA, when n is the number of the card program area (1..8) in which you wish to store the datalogger program.
3. To load a program from the card into the datalogger, enter 2nA, where n is the number of the card program area (1..8) in which the program is stored.
4. To clear a program stored in a card, enter 3nA, where n is the card program area.

---

**NOTE**

This does not immediately free the memory used by the program for use for data storage. The card must be erased to allow that memory to be used again.

---

CR10s with software OS10-0.1 (and later), 21Xs with OSX and CR7s with OS7 software support automatic loading of a program from card program area 8 when the datalogger first powers up. The program is loaded into program memory, compiled, and executed.

---

**CAUTION**

The datalogger clock is set to the default value on power-up and needs setting manually to the correct date and time if this is critical to the application.

---

## 5.2 Transferring Programs to/from DSP4

Programs can be loaded and unloaded to and from a DSP4 in the same way as is done with an SM192/SM716 Storage Module (see DSP4 manual for further details).

## 6. Selected Operating Details

### 6.1 Inserting the Card in the CSM1

One of the features of the CSM1 is that you can either leave the card in the module and move the module from datalogger to datalogger (or from datalogger to office), or you can install the module with the datalogger and transport the memory cards between sites. Whichever is your chosen method of use, it is critical that the card is inserted into the module correctly.

When plugging the card in, please ensure that you push the card in firmly until it reaches the back of the card connector, i.e. it will go no further. The eject button on the card slot comes out as the card is pushed in. When the card is fully inserted, the end of the eject button should be level with the end of the card.

If the card is plugged into a module which is then connected to a datalogger the CSM1 validates the status of the card on power-up and reports any errors by flashing the status LED (see Table 1). If a card is plugged into a CSM1 which is already connected to a datalogger, similar tests are done on the card and the status LED flashes with the same meaning. If the LED flashes three times indicating the card is not plugged in it is possible that you have not inserted it correctly. To correct this you should remove the card completely and reinsert it.

### 6.2 Response of CSM1 on Detection of Full Card Memory

The card memory structure is predefined as 'fill and stop'. This means that data can be stored in the card until it is determined that it is full. The card is then marked by the CSM1 as being full and can no longer be used to store new data. The card must be erased before it can store new data.

The CSM1 recognizes that a card is full if it reaches the end of available memory while storing data. At this point the CSM1 changes the indicated number of available storage locations to zero and marks the card as full. Further attempts at data storage are not accepted; this is evident when a datalogger attempts to write data as the 'Write' LED will not illuminate.

This method of marking the card as full can lead to confusion when writing data to the card with the CR10 or when storing programs in the card using a PC. This is because both processes involve transferring blocks of data to the CSM1, with a validation procedure at the end of the transfer process.

In the case of the CR10 data, the CSM1 accepts data from the datalogger as a complete block and writes it to the card as it receives the block. However, it only updates its internal data pointers to indicate it has new data at the end of a transmission from the CR10. Also, the CR10 only updates its own Storage Module Pointer (SPTR) after a signal from the CSM1 that it has received the data correctly.

If the CSM1 detects that a card is full while accepting data from a CR10, it marks the card as full and immediately aborts the receipt of the whole block. It

does not move its internal pointers and consequently the CR10 does not get confirmation of successful data transfer at the end of the transmission; thus it does not move its SPTR. This means that no data is lost as the CR10 will send the whole block again. However, for a large block transmission from the datalogger, e.g. a manual dump of the whole memory, it can appear that the card suddenly becomes full without storing any data from the datalogger. Subsequent attempts at storing small blocks of data to the card are unsuccessful, as the card is marked as full and the module does not wake up. This method of operation has been implemented to minimize power consumption in remote applications.

A similar phenomenon will be apparent when writing large programs or text files to a card that is almost full.

Once a card has been marked as full it cannot store any more data until the 'full' flag is reset. There are two mechanisms you can use to do this:

- 1) By erasing and resetting the card, after extracting your data.
- 2) By disconnecting and reconnecting the CSM1 to the datalogger. If the card is plugged in when the CSM1 powers up, the CSM1 will warn you that the card has previously been marked as being 'full' by flashing the status LED seven times. The CSM1 will then reset the card's internal 'full' flag to enable further attempts at data storage to be made. If there is any free space in the card, it can then be used to collect smaller blocks of data. However, if it is reconnected to the same CR10, as in the example above, which still has too large a block of uncollected data, collection will not be successful. If too large a block is sent again, the card will be marked as full again.

## 7. Telecommunications

Campbell Scientific's CSMCOM program is recommended for communicating with the CSM1 when using IBM-compatible PCs under the DOS operating system. However, any computer can communicate with the CSM1 via an RS232 interface.

This section describes how to connect and establish communication with the CSM1 when not using CSMCOM. It also describes the telecommunications commands to which the CSM1 responds. These can be used when writing your own software, or when using the terminal emulator mode of CSMCOM.

### 7.1 Connecting the CSM1 to Non-IBM Computers

The SC532(A) is normally used to connect the CSM1 to the serial port of a computer. The SC532(A) is wired as a DCE, RS232 device. Please refer to the SC532(A) manual and your computer manual to ensure correct connection of the two.

To wake up the CSM1 into telecommunications mode, two handshaking lines (lines 4 and 20) on the SC532(A) must be taken from the MARK (negative) to SPACE (positive) state and held high during communications.

**NOTE**

It is critical that line 4 does not go positive before line 20. This can either be done under careful software control, or by tying the two lines together at the SC532(A) and using just one line on the computer to pull the two lines high on the SC532(A).

If it is not possible to control these lines by software, then the older SM232A is available that uses a mechanical switch to emulate setting the lines high or low. Please contact Campbell Scientific for further details.

---

## 7.2 Establishing Communications

After raising the lines as described above, a series of carriage return characters (ASCII decimal 13) with a 100ms delay between them must be sent to the CSM1 to enable it to determine the baud rate for communication.

The CSM1 works with a serial format of eight bits, one start bit, one stop bit, and no parity. Valid baud rates are 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400. Please contact Campbell Scientific for further advice before attempting to use speeds of 38400.

Normally, the CSM1 will respond to the second carriage return sent to it with the standard telecommunication prompt, a '?' preceded by carriage return, linefeed.

## 7.3 Breaking Communications

Communications are broken either by taking lines 4 and 20 low, or by removing power from the CSM1, e.g. unplugging the SC532(A) power supply.



# Appendix A. 9-Pin Storage Module Interface

---

<b>Pin 1 (input)</b>	5V supply to CSM1. The minimum voltage for operation is 4.85V. Below approximately 4.85V the CSM1 will neither accept data nor communicate; it is disabled. The only time that this may cause a problem is when using Campbell Scientific equipment with the PC201 card. This uses the PC 5V supply to power the module. The 5V supply of some PCs may not be set up accurately and be below the minimum voltage for operation of the CSM1. If this is the case you should use another PC or have the PC power supply adjusted. For powering the CSM1 this power supply must be capable of supplying up to 18mA continuously, with peaks of up to 100mA.
<b>Pin 2</b>	Power and signal ground
<b>Pin 3 (input)</b>	Ring indicate (not used by module)
<b>Pin 4 (output)</b>	CSM1 Transmit Data line (TxD). In telecommunications mode data is transmitted to the computer on this line. Data is transmitted asynchronously at levels of 0-5V DC. The idle state is 0V. This line is held high to indicate that the CSM1 can accept data, when receiving data from 21X and CR7 dataloggers.
<b>Pin 5 (input)</b>	Modem Enable line. The module uses this line to detect other communications activity on the datalogger serial port. When this line is high, the CSM1 does not attempt to communicate.
<b>Pin 6 (input)</b>	Printer Enable/Serial Device Enable line. In combination with line 7 this is used by the datalogger to wake up the CSM1 to receive data. The mode of operation depends on the datalogger type (see datalogger manual for further details). To force the CSM1 into telecommunications mode this line must be taken high with line 7.
<b>Pin 7 (input/output)</b>	Clock/Handshake line. For the CR10, this line is used in conjunction with line 6 to control transfer of data to the CSM1. For telecommunications mode this must be taken high before, or at exactly the same time as pin 6.
<b>Pin 8 (input)</b>	Not used
<b>Pin 9 (input)</b>	CSM1 Receive Data line (RxD). Data transmitted from a datalogger or computer is received on this line.





# ***Appendix B. Internal Data Formats and Pointers in the Card Memory***

---

*Knowledge of the format in which data is stored in the card memory is not normally necessary to use the CSM1. However, if you intend to save text files in the card or need to try to extract corrupted data, the following information may be useful.*

## **B.1 Byte Pairs**

All data stored in the card is stored in byte pairs. If you transmit a file to the module that has an uneven number of bytes, the remaining odd byte at the end of the file is filled with an ASCII null (0 Hex) character.

## **B.2 File Structure**

Data files are delimited in the card using markers called filemarks. A filemark is written to the card when either a CSM1 with card is powered up or if a card is plugged into a CSM1 that is powered up. Filemarks can also be written into a card under software control in telecommunications mode or with some dataloggers. Filemarks are also written into the card, with dataloggers that fully support the Storage Modules, when a program is recompiled in the datalogger. A filemark is not written into the card if a filemark already exists in the preceding data location.

A filemark is the byte pair 7C 01 Hex stored in the card. If you send this sequence to the CSM1 as data it is interpreted as a filemark.

To allow for future developments the CSM1 always places a filemark in the first location in a card during the process of erasing a card. It is necessary to jump over this filemark, by placing the display pointer at location 2, before you can read the first data file in the card.

## **B.3 Program Files**

Datalogger programs are stored in the card in the next available area of memory (as would be new data). Datalogger programs always start with the byte sequence 011111101 XXXXXXXX 1XXXXXXX (X = don't care). A filemark may also be found before this sequence if no data has been stored in the card since the card was last powered down. To enable the CSM1 to find each program quickly without having to scan through the whole card, the card holds a record (in the reserved memory) of where each program starts.

## **B.4 Data Pointers**

256 bytes of the card are reserved to hold working data relevant to the management of data in the card. Within this area are stored three pointers you can observe or control via the status line:

The storage reference pointer (R): this points to the location where the CSM1 will write the next data value in the card.

The display location pointer (L): this points to the location that will be the first value output in response to the C, F or HF commands. If this pointer points to a filemark or start of a program, no data is output by these commands. You must move the pointer past the filemark to get further data. The simplest way of doing this is to use the NFM command. (Appendix D.)

The dump pointer (D): this is used by the PC software to recall the point from where new data should be collected. The dump pointer points to the first location where new data should be collected. It is up to the PC software to move the dump pointer after successful data collection. The CSM1 does not move this pointer.

# Appendix C. Card Batteries

---

*The make and type of battery supplied with the memory card can vary with the size and manufacturer of the card. Campbell Scientific reserves the right to supply cards of different manufacture with different battery sizes and types.*

*The batteries fitted to the cards are readily obtainable from electrical suppliers. You must ensure that the replacement battery is compatible with those recommended by the manufacturer of the card. Please refer to the manufacturer's instructions for further details on removing and replacing the batteries.*

## C.1 General

There are two technologies used in the batteries, both of which are lithium based:

- a) Lithium polycarbonmonofluoride (Lithium CFX), which is the newer technology. These batteries are rated to very low temperatures, e.g. -40°C, and usually have a slightly higher initial capacity than those below. Manufacturers' part numbers for these batteries normally start with a 'BR' suffix.
- b) Lithium manganese batteries, which are currently more readily available. These batteries operate at temperatures down to -20°C. The suffix for these batteries is generally 'CR'.

For some battery sizes either type of technology is available. It is important to match the digits after the suffix, e.g. BR2032 is equivalent to CR2032.

Both battery types have reduced capacity at low temperatures, e.g. at -20°C the capacity may only be 50% of that at room temperature. However, this is not as critical as one might imagine, for the following reasons:

- a) Power is only taken from the back-up battery when the card is not powered from the CSM1. Therefore if the card is plugged into a CSM1 that is plugged into a datalogger, the back-up battery is not in use.
- b) Power consumption from the back-up battery is highly dependent on temperature. For example, typical figures for a 1Mb card are 0.7mA at -30°C, 2mA at 25°C and 16mA at 60°C. Therefore, the reduced capacity is offset by the lower power consumption.
- c) Although the effective capacity of a battery is reduced at sub-zero temperatures, if the card warms up, the effective capacity increases again.

To optimize the life of the back-up battery:

- a) Leave the card plugged into a powered-up CSM1 for as long as possible.
- b) When using the cards at very high or low temperatures, try to return them to a temperature close to room temperature as soon as they are unplugged from the CSM1, e.g. in a very cold situation, unplug the card and transfer it to the inside pocket of your coat.
- c) When the card will not be used for long periods, remove the battery from the card and store it separately.

## C.2 Battery Status Detection

There are three levels of battery status detection. The exact detection is done within the card itself; the status line only reflects the output from the card. The exact battery voltage thresholds the card indicates vary with the card manufacturer. The terminal voltage of the battery also varies with battery type and with temperature. Therefore the exact capacity of battery available when the level detector indicates the intermediate capacity (1 on the status line, LOW with CSMCOM), is not fixed. In particular, at low temperatures where the battery voltage is reduced, the LOW indication may be rather pessimistic.

The CSM1 only prevents data storage in the card when the card indicates the battery is dead (0 on status line). If the card is being used at very low temperatures and is almost exhausted, the dead indication may be given when the card is still able to hold data. In an emergency, you may be able to continue to use the card for a short period by warming the card, which will cause the battery voltage to rise slightly, before placing it in the CSM1.

---

**NOTE**

Some card types only measure the battery voltage when the card is first powered up. The level of indication is then held in a latch. Other cards monitor the battery voltage continuously.

---

# Appendix D. Telecommunications Commands

---

*Once in the telecommunications mode the CSM1 responds to the commands listed below. To enter a command enter the characters followed by a carriage return character. The CSM1 does not support text correction using the backspace/delete characters; if an unexpected character is read, command entry is aborted and the prompt is returned. With the exception of the F and OH commands, all commands finish with a carriage return/linefeed/prompt sequence on completion. Some commands show the status line before the prompt.*

Command	Description
<b>A Status</b>	<p>This command returns the status of the card on a single line with each parameter preceded by a character. The parameters are as follows:</p> <p><i>Vn.m</i>, where n is the CSM1 PROM version number and m is the card version number.</p> <p><i>Mnn</i>, where nn is the number of 16 kbyte pages of memory in the card.</p> <p><i>Bn</i>, where n is 0, 1, or 2 and indicates the battery status. 0 is dead, 1 is low (less than 3% capacity left) and 2 is OK. The CSM1 does not attempt to store data if the battery is indicated as being dead.</p> <p><i>En</i>, where n in the range of 0..254 indicates the number of bad characters received from a datalogger, e.g. framing errors. If n=255, this indicates that the CSM1 has detected some corruption of its reserved memory area in the card, which could indicate some of the other values on the status line are bad.</p> <p><i>Pn</i>, where n indicates the number of programs stored in the card.</p> <p><i>Annnnnnn</i>, where nnnnnn is the number of free storage locations.</p> <p><i>Rnnnnnn</i>, where nnnnnn is the position of the storage reference pointer, i.e. where the next new data value will be stored.</p> <p><i>Lnnnnnn</i>, where nnnnnn is the position of the display pointer, which indicates where subsequent instructions that display or output data will start.</p> <p><i>Dnnnnnn</i>, where nnnnnn is the position of the dump pointer, which is used to mark where data was last dumped up to.</p> <p><i>Cnnnnn</i>, where nnnnn is a decimal representation of the checksum of all characters transmitted by the CSM1 since the last prompt character was transmitted. (This includes echoed commands.) The checksum is the standard Campbell Scientific checksum used by the dataloggers when transmitting binary data. Please refer to the datalogger manuals for further details.</p>

An example of a status line is:

V1.1 M16 B2 E0 P3 A80313 R50632 L2 D23455 C23922

<b>nnnnnnG</b> <b>Move</b> <b>display</b> <b>pointer</b>	Moves the display pointer to a specified location nnnnnn. The status line is displayed after the pointer is moved.
<b>nnnnnnC</b> <b>Output</b> <b>ASCII</b> <b>data</b>	Outputs data in a comma delineated, ASCII format. nnnnnn arrays of data are dumped, or until the next filemark is reached. 0C means dump continuously until the next filemark. When the command is completed the display pointer is moved to after the end of the data dumped and the status line is sent.
<b>nnnnnnF</b> <b>Output</b> <b>binary</b> <b>data</b>	Outputs nnnnnn storage locations of data, from the display pointer, in a binary format. The data is output in exactly the same format as was transmitted to the file for storage. Two bytes of data are sent per location. At the end of the block of data, the CSM1 waits. If the next character sent to the module is an 'S' it transmits a two-byte binary checksum that is the signature of the previous data. If any other character is sent the prompt is returned by the CSM1. The display pointer is moved to after the position of the data just dumped. The F command only dumps up to a filemark in the card and returns no further data until other instructions are used to move it past the filemark. It will jump over programs stored between blocks of data providing that the program is not delimited by filemarks. 0F causes a continuous dump until the next filemark is reached.
<b>nnnnnnH</b> <b>F</b> <b>Output</b> <b>binary</b> <b>and ASCII</b> <b>data</b>	This command is similar to the F command except data is sent in both hex and ASCII formats, in a way similar to the DOS DEBUG command. At the start of each line the location of the first value output on that line is printed. This command is intended for diagnostic purposes only. When dumping data, the HF command does not stop at filemarks or programs, but shows the hex representation of the filemark and program data.
<b>NFM</b> <b>Next</b> <b>filemark</b>	Changes the display pointer to point to the location after the next filemark in the card. This instruction can take several seconds to find a filemark if the files are large as the CSM1 has to scan through the data to find the next filemark. The NFM command ignores programs stored in the card, unless the program itself has a filemark at the beginning. The status line is transmitted when the filemark is found.
<b>BFM</b> <b>Back to</b> <b>filemark</b>	Moves the display pointer backwards in memory to find the previous filemark. As with the NFM command the pointer is set to point to the location after the filemark. The positions of the previous two filemarks are recorded in the card to improve the response time of this instruction. Moving further back than two files can take several seconds if the files are large. The status line is displayed when the filemark is found.
<b>9H</b> <b>Force</b> <b>filemark</b>	Forces a filemark to be stored in the card at the current SRP. The status line is then displayed.

<b>0H Receive data</b>	Makes the CSM1 store any subsequent data sent to the CSM1 in the card. The CSM1 sends a carriage return, linefeed and '<' prompt when it is ready to receive the data. This mode can only be terminated by resetting the module, e.g. removing power or dropping the handshaking lines to the SC532(A).
<b>4H Move dump pointer</b>	Moves the dump pointer to the current display pointer position, then displays the status line.
<b>08G Move display pointer</b>	Moves the display pointer to the current position of the dump pointer, then displays the status line.
<b>1248K Full reset and test</b>	This resets all pointers and tests all the RAM in the card. <i>All stored data and programs in the card are erased.</i> The memory test can take a while as it is extremely thorough. There are three phases to the test. The memory is tested in 16 kbyte pages. First, pseudo random data is written to the card and a series of '+' symbols is displayed as each page is written to. The data written is then verified, page by page. A '-' symbol is displayed for each page of good memory. At the same time further test data is written into the card. On a new line the result of a validation check is shown, again showing good pages as '-'. Another test pattern is loaded into the card at this stage. Finally this is validated on the third line and the memory erased at the same time. The status line is then returned. If at any stage an 'x' appears, this indicates a page of memory has failed the test. Please contact Campbell Scientific for further advice if a failure occurs.
<b>1249K Quick reset</b>	This command is similar to the 1248K command — <i>it also erases all memory and programs.</i> However, it does not do a complete memory test, it simply resets the reserved memory and the pointers, which effectively erases the memory. This is much faster than the 1248K command and is designed for routine erasure of the card. Do not use this instruction if the memory card has been corrupted or battery power lost as without doing a full memory test the CSM1 is unable to accurately determine how much good memory the card has — you should use the 1248K command instead. If this command is used by mistake, and no further data is stored in the card, it may be possible to recover old data from the card; please contact Campbell Scientific for further details.
<b>N Signature</b>	Checks and reports the checksum signature of the CSM1 PROM. If the signature is incorrect a zero is returned otherwise the PROM signature is returned. The status line is returned after this command.

<b>nDP Send program</b>	Dumps the program stored in program area n. The CSM1 sends the program followed by a 2-byte binary checksum. The checksum includes all characters transmitted since the last prompt character. If the program area is empty the CSM1 sends the byte sequence 30 07 07 Hex, followed by the checksum and then the prompt.
<b>nSP Store program</b>	Store a program in program area n. The CSM1 responds with a '<' character and then stores any data sent to it in the program area specified. Programs must start with an ASCII character 7D (Hex) and end with two ctrl-E characters (07 07 Hex). At the end of the transmission, detected by the receipt of the 07 07 sequence, the CSM1 sends carriage return, linefeed and a decimal representation of the checksum of the received program file. The status line is then sent.
<b>nCP Clear program</b>	Clears a program from program area n. However, the memory used by the program cannot be used for data storage until the card is erased. The status line is sent after this command.

Commands that send data from the CSM1 support software flow control. Sending an 'XOFF' character (ASCII 11 Hex, ctrl-S) to the CSM1 stops output. Sending any other character restarts output. If the flow is not restarted within 10 seconds the CSM1 continues transmission of the data.

Sending Esc (1B Hex) or ctrl-C (03 Hex) aborts the data output and returns to the CSM1 prompt.



# Appendix E. SC532A

---

The SC532A differs from the SC532 in three ways:

1. Provides 12 V on PERIPHERAL Port pin 8
2. “PROGRAM” mode
3. Jack on SC532A POWER input

The SC532A supplies 12 VDC via PERIPHERAL connector pin 8 for the purpose of powering 12 V peripherals.

The SC532A’s internal jumper selects “SC532” or “PROG” mode. For the CSM1 leave the jumper in the factory default position of “SC532.” “PROG” is a special mode for satellite transmitters only.

The SC532A has a POWER jack allowing convenient replacement in the event of AC adapter failure (see CAUTION below). The jack also facilitates operating the SC532A from datalogger power at remote sites by using optional field cable (item number 14020) in place of the AC adapter. Simply plug the cable’s DB9 connector into the datalogger’s CS I/O port for 12 V power. If you have an earlier datalogger lacking 12 V on CS I/O port pin 8 (DVM test between paper-clip inserted in pin 8 and GND) the field cable can be modified to connect to the datalogger’s power terminals as follows:

1. Unplug field cable from datalogger and SC532A.
2. Cut off DB9 connector.
3. Remove cable sheath exposing positive (RED) wire and negative (BLACK) wire 2 inches back.
4. Strip ¼ inch of insulation from each wire and tin ends.
5. Connect RED wire to datalogger “12 V” and BLACK wire to datalogger power ground terminal. The correct barrel connector polarity is (+) on the inner bore and (–) on the outer sleeve.

The CSM1 requires that a minimum of 6 VDC @ 18 mA be supplied to the SC532A POWER jack from AC adapter or field cable. The maximum voltage that can be safely applied to the SC532A POWER jack is 17 VDC.

## CAUTION

Before plugging the power connector into the SC532A, if you have 1) cut off optional field cable’s DB9, 2) replaced the factory AC adapter, or 3) built your own dc power cable, make sure that the voltage polarity is correct on the barrel connector. Application of REVERSED POLARITY power to the SC532A can damage the SC532A, datalogger, and peripheral (not covered under warranty)!

Barrel connector inner bore (+)

Barrel connector outer sleeve (–)

The maximum POWER input voltage is 17 VDC!

---



# **CSMCOM SOFTWARE**



# ***CSMCOM Table of Contents***

---

<b>1. Installation</b>	<b>1</b>
<b>2. Getting Started</b>	<b>1</b>
<b>3. Menu Options</b>	<b>2</b>
3.1 T — Terminal Emulator:	2
3.2 A — Collect All Data Files	2
3.3 U — Collect Uncollected Data Files	3
3.4 N — Collect Newest Data File	3
3.5 L — Collect One Data File Starting at a Specified Location	3
3.6 P — Collect All Program Files	3
3.7 D — Store a .DLD Program File	4
3.8 F — Store a File	4
3.9 E — Erase, Reset and Test the Card	4
3.10 C — Clear Data Area Quickly	5
3.11 Q — Quit	5
<b>4. Filename Conventions</b>	<b>5</b>
<b>5. Data File Format Options</b>	<b>6</b>
5.1 D — Printable ASCII Data with IDs	6
5.2 C — Comma Delineated ASCII Arrays	7
5.3 A — As Stored (8-Bit Data)	7
<b>6. Additional Information on CSMCOM Operation</b>	<b>7</b>
6.1 Reading Data from More than One Card	7
6.2 Aborting Options	7
6.3 Communication Errors	7
6.4 Battery Voltage Warnings	8
6.5 Card Errors	8

## **Appendices**

<b>A. Command Line Operation</b>	<b>A-1</b>
A.1 Introduction	A-1
A.2 Additional Command Line Parameter Options	A-1
<b>B. COM Port Addresses</b>	<b>B-1</b>
B.1 CSMCOM and COM3, COM4	B-1
B.2 CSMCOM and the PC201 Card	B-1
B.3 PS/2 Computers and CSMCOM	B-1



# CSMCOM

---

*CSMCOM supports Campbell Scientific's Card Storage Module when the CSM1 or MCR1 is used as the card reader. It provides a simple and efficient way for you to extract data from a card. Most program functions can be accessed from the DOS command line, allowing you to incorporate the program in batch files for automated data extraction.*

*SMCREAD supports PC card drives installed in the computer.*

*If you are familiar with Campbell Scientific's SMCOM program you will find many aspects of the operation of CSMCOM to be similar.*

## 1. Installation

CSMCOM consists of a single file that must be copied from the floppy disk supplied either onto a working floppy disk or onto the hard disk of the PC on which it is to be used.

CSMCOM needs an IBM-compatible PC or PS/2 machine with a minimum of 256K of memory. It will run on any monitor capable of displaying an 80 x 25 text screen (see appendix A for older LCD display options). A version of PC or MS-DOS 2.0 or higher is required.

## 2. Getting Started

To run CSMCOM type 'CSMCOM <Enter>'. CSMCOM first asks which COM port the storage module is connected to; enter 1, 2, 3 or 4. CSMCOM now attempts to communicate with the module and ascertain the status of the card. The program supports COM ports 1 to 4, with the addresses and IRQ lines as used by recent releases of Campbell Scientific software (see Appendix B for further details).

The module must be connected to the PC via an SC532(A) interface, or directly to a Campbell Scientific PC201 card with an SC209 cable. If an SM232A interface is being used please refer to Appendix A for further information.

CSMCOM automatically tests the PC speed and sets the communication speed accordingly. If unable to communicate with the module CSMCOM reduces the communication speed (baud rate) and makes two further attempts at establishing communication with the module. Appendix A describes how to adjust the communication speed options manually. It is also possible to increase the speed above the default to increase data transfer rates on newer, faster PCs.

Once communication with the CSM1 has been established CSMCOM displays the status of the card plugged into the module, in a 'Module Status Window' at the top of the screen. If CSMCOM has detected that the card is not plugged in, it asks you to plug in the card before continuing. If any errors in the registers of the card are found or the card battery is low a warning message is displayed.

## 3. Menu Options

CSMCOM displays the menu options as follows:

**CSMCOM options:**

- T -- Terminal emulator**
- A -- Collect All data files**
- U -- Collect Uncollected data files**
- N -- Collect Newest data file**
- L -- Collect one data file starting at a specified Location**
- P -- Collect all Program files**
- D -- Store a .DLD program file in the card**
- F -- Store a text File in the card**
- E -- Erase, reset and test the card**
- C -- Clear data area quickly**
- Q -- Quit**

**Option:**

Select an option by typing the character shown at the start of each line.

### 3.1 T – Terminal Emulator

In this mode, the PC runs as a simple terminal allowing you to issue telecommunication commands directly to the module (see CSM1 manual for further details). You should only need to use this option when trying to diagnose problems with a corrupted card.

To obtain a summary of valid telecommunication commands press F1; a second press of F1 displays a second screen. To leave the terminal emulator, press the Esc key.

The terminal emulator option asks the module for an update of its status after each command. This ensures the module status window is kept up to date.

---

**NOTE**

On a slow PC, or one running at a low baud rate, there can be a slight delay after a command has been actioned. To avoid confusion wait until the prompt is returned before typing a new command.

---

### 3.2 A – Collect All Data Files

This option collects all valid data files in the memory card. CSMCOM first asks for a root filename that is used to name computer data files (see section 4 for details of filename conventions.) After a valid name is entered, CSMCOM prompts for the format in which you wish to write the data on disk (see section 5).

Once a valid format has been entered CSMCOM starts at the beginning of the card memory and scans through, writing each card file containing data to a



new file on the hard disk. For each file the message: '**xxxxxx: writing to file rootnn.dat**' appears, where **xxxxxx** is the location of the start of the file.

If CSMCOM finds a file that contains a program, it jumps over the program and displays 'no data found in this area'.

This option can be aborted by pressing the Esc key.

### 3.3 U – Collect Uncollected Data Files

This option operates in a similar way to the A option, with the exception that it starts reading data from the position of the storage module dump pointer. This pointer is used to mark where data was last collected from. It is automatically moved after successful collection of data by the A or U options of CSMCOM.

Thus the U option can be used to collect any new data files stored in the card since data was last read using CSMCOM.

### 3.4 N – Collect Newest Data File

This option also works in a similar way to the A option with the exception that it collects just the latest file in the card, and the data is written to a specific file rather than one specified with a root filename.

### 3.5 L – Collect One Data File Starting at a Specified Location

This option stores data to a specific file, as with the N option, but CSMCOM first asks for the starting location of the file. This option is mainly used for recollecting old data that has already been collected, or to recover data from a corrupted card.

The L option performs only a limited check on the validity of the start location you enter. It is possible, for example, to request data from any part of the card, including an empty area. The start location should point to the first data value in the file rather than the filemark that precedes it.

### 3.6 P – Collect All Program Files

The P option collects all datalogger programs that have been stored in the card. CSMCOM first asks for a root collection filename. Enter a valid name such as 'ROOT'. CSMCOM then checks which program areas contain programs and reads the programs out to disk. Programs 1 to 8 are stored in files with the numbers 1 to 8 appended on the end of the prefix of the name. The suffix is set to .DLD, e.g. for a root name of ROOT, the files would be named ROOT1.DLD, ROOT2.DLD and so on.

As the program is being read from the card a '.' is displayed for every 100 characters read from the card.

When a program is collected a checksum verification is done. If no errors occur 'OK' is displayed; otherwise a warning is shown. CSMCOM attempts up to three retries if an error occurs.

### 3.7 D – Store a .DLD Program File

This option is used to store a .DLD program file, as created by EDLOG, into one of the program areas of the card. CSMCOM first displays the current status of the program areas in the card to show you which areas are in use. Next CSMCOM asks you to enter the name of the .DLD file. CSMCOM appends the .DLD suffix if necessary.

You are now asked in which card program area you wish to store the program. Enter a number between 1 and 8.

CSMCOM now loads the program into the card, displaying a '.' as each 100 characters are transferred. A checksum verification is used to ensure the program has been loaded without errors. If no errors occurred, 'OK' is added to the end of the string of '.' characters, otherwise an error warning is displayed.

### 3.8 F – Store a File

Option F allows you to store non-program text or data files in the card.

---

**NOTE**

Certain binary strings signify file delimiters (see the CSM1 manual) and these should not be included in the file. In addition, Ctrl-Z (ASCII 26 decimal) is interpreted as being the end of a text file so cannot be included in the file to be stored.

---

If the file stored has an odd number of bytes the storage module appends a null character (ASCII 0 decimal) to the end of the file. A filemark is written at the end of the file stored.

No validation of file storage is done for this option.

### 3.9 E – Erase, Reset and Test the Card

The E option should be used with caution as it completely erases all data and programs in the card. At the same time the card memory is fully checked to ensure the module can store and recall data correctly.

CSMCOM asks you to verify that you want to erase the card before this option runs. Enter Y for yes or N for no as appropriate.

CSMCOM runs the E option by issuing the '1248K' command to the module and displaying the status of the process. First a row of '+' symbols are displayed. One '\_' is written for each 16 kbyte page of memory written to. At the end of this row an equal number of '-' symbols are shown as the data is read out and checked successfully.

A further two rows of '-' symbols are shown as further read/write tests are done on the memory.

If an error occurs at any stage an 'x' is displayed and tests on memory after that page are aborted. After an error is detected the memory that can be used is limited to the pages of good memory tested before the failure.

If an error does occur, use the E option several times to check if the position at which the error occurs is repeatable. If the test fails at different points in the card, then either the card or module may have a serious hardware fault. If other cards work with the module, then the card is at fault and should not be used. If the error always occurs in the same memory page, it is possible to use the card with reduced capacity. Please contact Campbell Scientific for further advice.

At the end of the test, CSMCOM displays a message indicating how many locations are usable.

### 3.10 C – Clear Data Area Quickly

This option is similar to the E option, with the exception that it does not do a full memory test. Instead CSMCOM uses the '1249K' command to erase the card memory and reset the pointers. Before this, CSMCOM temporarily offloads all programs present in the card to disk. After erasure the programs are reloaded into the card so that only data is erased.

As with the E option, you are asked to verify that you wish to proceed with erasing the card.

As with options P and D, program reading and loading are subject to verification procedures. If CSMCOM is unable to reload a program back into the card, a warning is displayed and a temporary file containing the program is retained on disk. Such temporary files are named T1249K1.DLD...T1249K8.DLD depending on the program area in which the program is stored in the card. If an error occurs, you should copy or rename the files that have not been loaded into the card before investigating the problem. The file can later be loaded manually using the D option.

If the card is corrupted do not use the C option initially. Use the E option to fully test and reset the card.

### 3.11 Q – Quit

Q quits CSMCOM and returns to DOS.

## 4. Filename Conventions

For all options where CSMCOM asks you for a filename, if you enter a filename that contains '\*' or a '?' wildcard character a list of files of that specification is displayed on the screen. You are then prompted to enter the filename again.

You can abort an option at this point by not entering any name and pressing the Esc key.

If you have already entered a filename, before running an option, CSMCOM prompts you with the same name again. Pressing the Enter key at this stage accepts the previously entered name. To enter a different name, type out the new name in full and press Enter.

CSMCOM assumes a .DLD suffix for program files and a .DAT suffix for data files, unless you enter an alternative suffix or specify no suffix by terminating the filename with a '.'

For options where only one file can be read from, or stored in the card, e.g. the name of a program to be stored in the card, CSMCOM asks you for a specific filename, where the prefix can be up to eight characters long.

Where the option can generate several files on disk, e.g. options A or P, CSMCOM asks for a 'root' filename. You should enter a name that has six characters or less. CSMCOM then appends a number to the name, which is incremented with each file collected. Where the filename is six characters long the numbers 01 to 99 are appended to the name. Shorter names have a three-digit number appended, i.e. 001..999.

When you first specify the root filename, CSMCOM scans the disk for files with the same root name. If none are found, CSMCOM starts numbering the files at 1, e.g. ROOT001.DAT. If files with the same root name exist, new files are numbered sequentially starting with the first number that does not exist on the disk, e.g. if ROOT001.DAT and ROOT002.DAT already exist, the first new file will be called ROOT003.DAT.

If you specify a file that already exists on disk you are asked if you wish to overwrite the file. If you reply 'No', the option aborts and returns to the option menu.

## 5. Data File Format Options

Once you have specified a filename for a data collection option you are asked which format you want to use for storing the data on disk. The following message is shown:

<p><b>File Formats:</b></p> <p><b>D    --   Printable ASCII arrays with IDs</b></p> <p><b>C    --   Comma delineated ASCII arrays</b></p> <p><b>A    --   As stored (8 bit data)</b></p> <p><b>Esc --   Escape</b></p> <p><b>Format:</b></p>
--

### 5.1 D – Printable ASCII Data with IDs

This is the format that is most readable when printed. Each data value is preceded by an identifier (see example in section 4 of PC208 manual). This format occupies the most disk space on the computer, and can be difficult to read into some other analysis packages. The format is ambiguous for arrays of 100 or more points; it is therefore not recommended for purposes other than printing.

## 5.2 C – Comma Delineated ASCII Arrays

This format has array elements separated by commas. No element identifiers are included and all leading and trailing zeros are removed. Each array is terminated with a carriage return and linefeed. Comma delineated data is reasonably compact, can be imported into most analysis packages, and can also be read when printed (see example in section 4 of PC208 manual).

## 5.3 A – As Stored (8-Bit Data)

When this option is used CSMCOM does not perform any decoding on the data in the card. Data is read out of the module and is written to disk as it is.

If data has been stored from a datalogger in a binary format, the A 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. The PC208 SPLIT program can be used to decode the data. If you wish to decode data with your own software please refer to the datalogger manual.

---

**NOTE**

If the data was stored from the datalogger in ASCII format, the A format must be used to collect the data from the card. Storing data directly in ASCII format is inefficient in terms of use of the card memory, as printable ASCII requires five storage locations in the card (10 bytes) as opposed to one for binary data.

---

---

**NOTE**

Programs are skipped over when reading the card with this option. Only data files are collected.

---

# 6. Additional Information on CSMCOM Operation

## 6.1 Reading Data from More than One Card

As a general principle, where more than one card is to be read or reconfigured, it is advisable to complete the operations on one card, then quit CSMCOM, change to the next card and restart the program. This reduces the likelihood of confusing the source of data when transferring the data to disk.

## 6.2 Aborting Options

Most CSMCOM operations can be stopped by pressing the Esc Key. You are then returned to the option menu.

## 6.3 Communication Errors

CSMCOM collects data in blocks. As each block is collected a '.' is displayed. If CSMCOM detects an error during collection of a block, an 'x' is displayed and the block is requested again. Occasional block retries can be caused by corruptions as a result of electronic noise. If repeated 'x's are displayed this

would normally indicate either too high a baud rate has been selected for the computer hardware or that there is a hardware fault.

## 6.4 Battery Voltage Warnings

If when the program is run, you get a warning in the module status window that shows the card battery as being 'DEAD!', the battery in the card is close to being, or is completely discharged.

---

### CAUTION

Do not unplug the card from the module or unplug the module from its power source at this stage. Collect any data that is present in the card before removing power from the card.

---

It is also possible to change the battery in the card, while it is still powered by the CSM1, to prevent data loss from the card. On some types of memory cards it is not possible for the CSM1 to detect the new battery until the card is unplugged from the CSM1 and plugged back in again. With these cards the warning that the battery is dead is displayed until the card is removed and reinserted.

Normally you should collect all data, remove the card, replace the battery and then rerun CSMCOM using the E option to reformat and test the card. When you first plug in a card that has lost power CSMCOM indicates that the card is unplugged or corrupted and can show spurious numbers in the module status window. Continue past the warning message and select the E option. After the card has been erased and formatted the module status returns to normal.

If the battery status displays LOW you should change the battery as soon as possible as it may have less than 3% of its original capacity. If the battery status is 'DEAD' the CSM1 will not allow you to store data in the card.

## 6.5 Card Errors

If CSMCOM detects that the error counter in a memory card is reading 255, a warning is displayed in the module status window indicating that the card has been corrupted. This means that some of the reserved memory in the card has been corrupted. As the reserved memory area is used to store the various pointers to data in the card, the process of reading data out of the card may be prone to errors.

As a first step in reading a corrupted card, try using the normal procedures to read the data out of the card. If the data files are corrupted, not collected, or not complete then you should try to find out where the starts of the files are by running the Terminal Emulator (T) option. Position the display pointer at the beginning of the card using the 1G command and then scan through the card repeatedly using the NFM command to find the start of each file. Record the position of the start of each file as you go. Once you have determined where each file starts you can use option L to try to read out each file.

If you are unable to recover your data please contact Campbell Scientific for further advice.

# ***Appendix A. Command Line Operation***

---

## **A.1 Introduction**

All options in CSMCOM can be specified as command line parameters when the program is run. This enables semi-automated reading of the cards, as CSMCOM can be called from a DOS batch file.

The parameters should be entered in the order that they would be entered when running CSMCOM. If you wish you can spell out the name of the option selected in full, as only the first character of the options is read by CSMCOM.

An example is:

```
csmcom 3 all dfiles comma programs pfiles download prog1 1 q
```

This is equivalent to:

```
csmcom 3 a dfiles c p pfiles d prog1 1 q
```

This executes CSMCOM and communicates on COM3. All data files are collected and stored in files with a root name of DFILES.DAT in comma delineated format. All the program files are then collected and stored in a series of files with a root name PFILES. A new program stored in PROG1.DLD is then loaded into program area 1 in the card. Finally, CSMCOM quits.

## **A.2 Additional Command Line Parameter Options**

### **A.2.1 NOCOLOR**

If poor contrast is apparent as a result of trying to simulate CSMCOM's screen colors when running CSMCOM on a computer with an LCD screen, the contrast may be improved by entering NOCOLOR as the first command line parameter (see the SM232A option below).

### **A.2.2 SM232A**

If you wish to connect the CSM1 to a computer using an old SM232A storage module interface, you must enter SM232A as the first parameter of the command line. CSMCOM then asks you to toggle the switch on the SM232A, to wake up the module, when you start the program.

If you need to specify both the NOCOLOR and SM232A options append the two options together, with no spaces in between, as the first command line parameter, e.g.

```
csmcom NOCOLORSM232A
```

### A.2.3 Communications Baud Rate

The normal communication speed between the PC and the CSM1 is 19200 baud. However, when CSMCOM first starts it performs a simple test on the speed of access to the serial port. If access is slow, as happens with older IBM PC computers, the baud rate is reduced to 9600.

Some recent high-performance computers will reliably support serial communications at speeds higher than 19200 baud. For such computers, the speed of offloading data from the CSM1 can be increased by manually setting the baud rate to 38400 baud.

You can set the baud rate at which CSMCOM initially tries to communicate by specifying the baud rate, followed by 'B', as the command line parameter that follows NOCOLOR/SM232A (if used). Valid baud rates are 300, 600, 1200, 2400, 4800, 9600, 19200, and 38400. In common with the DOS MODE command, you do not have to specify the trailing zeros, i.e. 12B is equivalent to 1200B.

Examples:

```
csmcom 384B
```

```
csmcom SM232A 9600B
```

---

**NOTE**

CSMCOM has the same capability as GRAPHTERM and SMCOM to send character strings, with wait periods. Please refer to the PC208 manual.

---



# Appendix B. COM Port Addresses

---

## B.1 CSMCOM and COM3, COM4

While the addresses and interrupts associated with COM1 and COM2 are well known and supported, some variation exist in the interrupts and addresses associated with COM3 and COM4. The addresses and interrupts listed in the table below for COM3 and COM4 are 'standard' in many cases. Listed in the table are the addresses and interrupts (IRQ) supported by CSMCOM for all four COM ports. Any serial port or PC201 card used with CSMCOM must be set so the addresses and interrupts match the table. The appropriate switch and jumper settings for the PC201 card are also given.

These settings allow CSMCOM to be used with most standard hardware available for IBM PC/XT/AT computers (i.e. add-on serial cards, etc.). Most boards use switches or jumpers to set the addresses and interrupts. Please check the documentation for the board for details.

## B.2 CSMCOM and the PC201 Card

The PC201 card's switches and jumpers allow it to be configured as any of the COM ports given in the table. The settings for COM3 and COM4 differ from those given in the PC201 manual. CSMCOM does not support the COM3 and COM4 settings given in the PC201 manual.

## B.3 PS/2 Computers and CSMCOM

COM1 or COM2 should be used with IBM PS/2 type computers. COM3 and COM4 are not supported.

**TABLE B-1. Use of COM3 and COM4 with CSMCOM**

Addresses and interrupts (IRQ) supported by CSMCOM			PC201 CARD Settings	
Com Port	IRQ	Address	Switch 1234567	Jumper
COM1	4	03F8	1111111	3 & 4
COM2	3	02F8	1111101	1 & 2
COM3	4	03E8	1011111	3 & 4
COM4	3	02E8	1011101	1 & 2
1=CLOSED, 0=OPEN				



# **SMCREAD SOFTWARE**



# ***SMCRead Card Reading Software User Guide Table of Contents***

---

<b>1. Getting Started.....</b>	<b>1</b>
1.1 Installation.....	1
1.2 Running SMCRead .....	1
<b>2. Menu Options .....</b>	<b>2</b>
2.1 Terminal Emulator .....	2
2.2 A - Read All Data Files .....	2
2.3 U - Read Unread Data Files .....	3
2.4 N - Read Newest Data File.....	3
2.5 L - Read One Data File Starting at a Specified Location .....	3
2.6 P - Read All Program Files .....	3
2.7 D - Store a .DLD Program File .....	4
2.8 E - Erase, Reset and Teat the Card.....	4
2.9 C - Clear Data Area Quickly .....	5
2.10 S - Show the Position and Size of the Files in the Card.....	5
2.11 Q - Quit .....	6
<b>3. File Name Conventions.....</b>	<b>6</b>
<b>4. Data File Format Options .....</b>	<b>7</b>
4.1 D - Printable ASCII Data with IDs .....	7
4.2 Command Delineated ASCII Arrays.....	7
4.3 A - As Stored (8-Bit Data) .....	7
<b>5. Additional Information on SMCRead Operation .....</b>	<b>8</b>
5.1 Reading Data From More Than One Card .....	8
5.2 Aborting Options.....	8
5.3 Battery Voltage Warnings.....	8
5.4 Card Errors.....	8
5.5 Running SMCRead Under Windows 3.11 .....	9
5.5.1 SystemSoft Drivers.....	9
5.5.2 Ventura/Award Drivers .....	9
5.5.3 Phoenix Drivers.....	10

## **Appendices**

<b>A. PCMCIA Driver Software Notes.....</b>	<b>A-1</b>
A.1 General .....	A-1
A.2 Loading the Drivers.....	A-1
A.2.1 General .....	A-1

A.2.2 SystemSoft CardSoft Drivers.....	A-2
A.2.3 Ventura Micro's CardWare Drivers.....	A-2
A.2.4 Phoenix's CardManager Drivers.....	A-2
A.3 Adjusting Your CardServices Set-up.....	A-2
A.3.1 How to Choose Which Area of Memory to Allocate to CardServices .....	A-2
A.3.2 SystemSoft's CardSoft Drivers.....	A-3
A.3.3 Phoenix CardManager Card Services Driver.....	A-3
A.3.4 Ventura/Award Card Services Driver.....	A-3
A.4 Using MemMaker (supplied with MSDOS 6.xx).....	A-3
A.4.1 General.....	A-4
A.4.2 SystemSoft's CardSoft Drivers.....	A-4
 <b>B. Command Line Operation .....</b>	<b>B-1</b>
B.1 Introduction .....	B-1
B.2 Additional Command Line Parameter Options.....	B-1
 <b>C. SMCRead Error Messages.....</b>	<b>C-1</b>

# SMCRead

---

*SMCRead provides a simple and efficient way for you to extract data from a CSM1 card and write the data to disk on a PC fitted with a PCMCIA card slot. Most program functions can be accessed from the DOS command line, allowing you to incorporate the program in batch files for automated data extraction.*

*If you are familiar with Campbell Scientific's SMCOM or CSMCOM programs you will find many aspects of the operation of SMCRead to be similar.*

## 1. Getting Started

### 1.1 Installation

For SMCRead to operate correctly the PC on which it is run must have a PCMCIA compliant card slot and have software drivers loaded which comply with PCMCIA Card Services 2.1. Such software drivers are normally supplied with the PC hardware. Please refer to the manuals for the PC to find out which drivers are supplied and how to load them. Generally the process of loading the drivers is straightforward as some form of loading utility is provided. All that is normally required is to select and load the options needed to support Type 1 SRAM cards.

SMCRead has been tested with most of the drivers currently available. Up to date information on the tested drivers and tips on how to get them to work is given in Appendix A.

To install the SMCRead software simply copy all the files on the distribution disk into your working directory on your hard disk. In the first instance verify and test SMCRead while running DOS alone. If you wish to run SMCRead as a DOS program under Windows, please refer to section 5.5 of this manual.

### 1.2 Running SMCRead

To run SMCRead insert the card in the PCMCIA slot then type **SMCRead** and press ENTER. SMCRead will look for the card in logical slot zero. If it does not find a card there it will prompt you to remove the card and plug it back in again. SMCRead is then told by the system which slot the card is in.

You can also specify which slot to use on the command line, if you are unable to use slot zero and wish to avoid having to unplug and replace the card each time you use SMCRead (see Appendix B for further details).

If valid PCMCIA drivers are found and a card is found SMCRead displays the status of the card in a 'Module Status Window' at the top of the screen. If any errors in the registers of the card are found or the card battery is low a warning message is displayed. The status window also displays the version of Card Services drivers it has found and the Card Slot number.

## 2. Menu Options

SMCRead now displays the menu options as follows:

SMCRead options:

```
T -- Terminal emulator
A -- Read All data files
U -- Read Unread data files
N -- Read Newest data file
L -- Read one data file starting at
    a specified Location
P -- Read all Program files
D -- Store a .DLD program file in
    the card
E -- Erase, reset and test the card
C -- Clear data area quickly
S -- Show the position and size of
    the files in the card
Q -- Quit
```

Option:

Select an option by typing the character shown at the start of each line.

### 2.1 T — Terminal Emulator

In this mode, the PC emulates running a terminal connected to a CSM1 Card Storage Module allowing you to issue telecommunication commands directly as if sent to a module (see CSM1 manual for further details). You should only need to use this option when trying to diagnose problems with a corrupted card.

To obtain a summary of valid commands press F1; a second press of F1 displays a second screen. To leave the terminal emulator, press the ESC key.

SMCRead only supports a subset of the telecommunications commands the CSM1 offers. The commands not supported are 0H, nDP, nSP and N.

### 2.2 A — Read All Data Files

This option reads all valid data files in the memory card. SMCRead first asks for a root file name that is used to name computer data files (see below for details of file name conventions). After a valid name is entered, SMCRead prompts for the format in which you wish to write the data on disk (see Section 4).

Once a valid format has been entered SMCRead starts at the beginning of the card memory and scans through, writing each card file containing data to a new file on the hard disk. For each file the message: `'xxxxxx: writing to`



`file rootnn.dat` appears, where `'xxxxx'` is the location of the start of the file.

If SMCRead finds a file that contains a program, it jumps over the program and displays `'no data found in this area'`.

While reading data from the card SMCRead writes a `'.'` to the screen for every block of data read. The size of the block is approximately 1000 locations, but the exact block size can vary depending on the structure of the data stored in the card.

This option can be aborted by pressing the ESC key.

## 2.3 U — Read Unread Data Files

This option operates in a similar way to the A option, with the exception that it starts reading data from the position of the Storage Module Dump pointer. This pointer is used to mark the position from where data was last read. It is automatically moved after successful collection of data by the A or U options of SMCRead.

Thus the U option can be used to read any new data files stored in the card since data was last read using SMCRead.

## 2.4 N — Read Newest Data File

This option also works in a similar way to the A option with the exception that it reads just the latest file in the card and the data is written to a specific file rather than one specified with a root file name.

## 2.5 L — Read One Data File Starting at a Specified Location

This option stores data to a specific file, as with the N option, but SMCRead first asks for the starting location of the file. This option is mainly used for re-reading old data that has already been read, or to recover data from a corrupted card.

The L option performs only a limited check on the validity of the start location you enter. It is possible, for example, to request data from any part of the card, including an empty area. The start location should point to the first data value in the file rather than the filemark that precedes it.

## 2.6 P — Read All Program Files

The P option reads all datalogger programs that have been stored in the card. SMCRead first asks for a root collection file name. Enter a valid name such as `'ROOT'`. SMCRead then checks which program areas contain programs and reads the programs out to disk. Programs 1 to 8 are stored in files with the numbers 1 to 8 appended on the end of the prefix of the name. The suffix is set to `.DLD`, e.g. for a root name of `ROOT`, the files would be named `ROOT1.DLD`, `ROOT2.DLD` and so on.

As the program is being read from the card a '.' is displayed for every 100 characters read from the card.

## 2.7 D — Store a .DLD Program File

This option is used to store a .DLD program file, as created by Edlog, into one of the program areas of the card. SMCRead first displays the current status of the program areas in the card to show you which areas are in use. Next SMCRead asks you to enter the name of the .DLD file. SMCRead appends the .DLD suffix if necessary.

You are now asked in which card program area you wish to store the program. Enter a number between 1 and 8.

SMCRead now loads the program into the card, displaying a '.' as each 100 characters is transferred.

## 2.8 E — Erase, Reset and Test the Card

### CAUTION

The E option completely erases all data and programs in the card. At the same time the card memory is fully checked to ensure the module can store and recall data correctly.

SMCRead asks you to verify that you want to erase the card before this option runs. Press Y for yes or N for no as appropriate.

SMCRead runs the E option by issuing the '1248K' command to the module and displaying the status of the process. First a row of '+' symbols is displayed. One '+' is displayed for each 16 kbyte page of memory written. At the end of this row an equal number of '-' symbols is shown as the data is read out and checked successfully.

A further two rows of '-' symbols are shown as further read/write tests are performed on the memory.

If an error occurs at any stage an 'x' is displayed and tests on memory after that page are aborted. After an error is detected the memory that can be used is limited to the pages of good memory tested before the failure.

If an error does occur, use the E option several times to check if the position at which the error occurs is repeatable. If the test fails at different points in the card, the card may have a serious hardware fault. If the error always occurs in the same memory page, it is possible to use the card with reduced capacity. Please contact Campbell Scientific for further advice.

At the end of the test, SMCRead displays a message indicating how many locations are usable.

## 2.9 C — Clear Data Area Quickly

This option is similar to the E option, with the exception that it does not do a full memory test. Instead SMCRead uses the '1249K' command to erase the card memory and reset the pointers. Before this, SMCRead temporarily offloads all programs present in the card to disk. After erasure the programs are reloaded into the card so that only data is erased.

As with the E option, you are asked to verify that you wish to proceed with erasing the card.

As with options P and D, program reading and loading are subject to verification procedures. If SMCRead is unable to reload a program back into the card, a warning is displayed and a temporary file containing the program is retained on disk. Such temporary files are named T1249K1.DLD...T1249K8.DLD depending on the program area in which the program is stored in the card. If an error occurs, you should copy or rename the files that have not been loaded into the card before investigating the problem. The file can later be loaded manually using the D option.

If the card is corrupted do not use the C option initially. Use the E option to fully test and reset the card.

## 2.10 S — Show the Position and Size of the Files in the Card

This option asks SMCRead to scan through the card and report the position and types of files found in the card. A listing is written to the screen which is similar to a DOS directory listing, where the type of file, its size plus the beginning and end locations are shown. This information can be used to determine how many files are in the card and detect any wasted space. The start position of data files can be used with the L option to read specific files from the card.

There are three different types of files that the Show option will list:

- A data file, i.e. datalogger data. The Show option assigns a number to the file which starts at one at the beginning of the card and increments for each new data file.
- A program file, shown with the number assigned when the program was loaded into the card.
- A deleted program file, which is an area of the card which was previously used by a program but which is now unused because a program with the same number has been loaded again.

In addition, the program and deleted programs may be shown as being hidden if the program is stored next to, or within, a data file without filemarks at the beginning and end of the program. Where the program is stored in the middle of a data file (a rare event) the start and end of the data file will encompass both the program and data, but the file size will be the length of the data alone.

Once you start this option the speed of operation will depend on the size of files stored in the card. The screen will scroll up when the listing reaches the bottom of the screen. You can pause the scanning of the card by pressing any key. ESC will quit out of the S option altogether.

## 2.11 Q — Quit

Q quits SMCRead and returns to DOS.

# 3. File Name Conventions

For all options where SMCRead asks you for a file name, if you enter a file name that contains '\*' or a '?' wildcard character a list of files of that specification is displayed on the screen. You are then prompted to enter the file name again.

You can abort an option at this point by not entering any name and pressing the ESC key.

If you have already entered a file name, before running an option, SMCRead prompts you with the same name again. Pressing the ENTER key at this stage accepts the previously entered name. To enter a different name, type out the new name in full and press ENTER.

SMCRead assumes a .DLD extension for program files and a .DAT extension for data files, unless you enter an alternative extension or specify no extension by terminating the file name with a '.'

For options where only one file can be read from, or stored in the card, e.g. the name of a program to be stored in the card, SMCRead asks you for a specific file name, where the prefix can be up to eight characters long.

Where the option can generate several files on disk, e.g. options A or P, SMCRead asks for a 'root' file name. You should enter a name that has six characters or less. SMCRead then appends a number to the name, which is incremented with each file read. Where the file name is six characters long the numbers 01 to 99 are appended to the name. Shorter names have a three-digit number appended, i.e. 001..999.

When you first specify the root file name, SMCRead scans the disk for files with the same root name. If none are found, SMCRead starts numbering the files at 1, e.g. ROOT001.DAT. If files with the same root name exist, new files are numbered sequentially starting with the first number that does not exist on the disk, e.g. if ROOT001.DAT and ROOT002.DAT already exist, the first new file will be called ROOT003.DAT.

If you specify a file that already exists on disk you are asked if you wish to overwrite the file. If you reply 'N' or 'No', the option aborts and returns to the options menu.

## 4. Data File Format Options

Once you have specified a file name for a data collection option you are asked which format you want to use for storing the data on disk. The following message is shown:

```
File Formats:
  D -- Printable ASCII arrays with IDs
  C -- Comma delineated ASCII arrays
  A -- As stored (8 bit data)
  Esc -- Escape
Format:
```

### 4.1 D — Printable ASCII Data With IDs

This is the format that is most readable when printed. Each data value is preceded by an identifier (see example in the PC208 manual). This format occupies the most disk space on the computer, and can be difficult to read into some other analysis packages. The format is ambiguous for arrays of 100 or more points; it is therefore not recommended for purposes other than printing.

### 4.2 C — Comma Delineated ASCII Arrays

This format has array elements separated by commas. No element identifiers are included and all leading and trailing zeros are removed. Each array is terminated with a carriage return and line feed. Comma delineated data is reasonably compact, can be imported into most analysis packages, and can also be read when printed (see example in the PC208 manual).

### 4.3 A — As Stored (8-Bit Data)

When this option is used SMCRead does not perform any decoding on the data in the card. Data is read out of the card and is written to disk as it is.

If data has been stored from a datalogger in a binary format, the A 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. The Split program can be used to decode the data. If you wish to decode data with your own software please refer to the datalogger manual.

---

<b>NOTE</b>	If the data was stored from the datalogger in ASCII format, the A format must be used to read the data from the card. Storing data directly in ASCII format is inefficient in terms of use of the card memory, as printable ASCII requires five storage locations in the card (10 bytes) as opposed to one for binary data.
-------------	---

---



---

<b>NOTE</b>	Programs are skipped over when reading the card with this option. Only data files are read.
-------------	---

---

## 5. Additional Information on SMCRead Operation

### 5.1 Reading Data from More Than One Card

SMCRead automatically detects the removal of a card from the card slot and will update the status window accordingly. SMCRead indicates the removal and insertion of a card by generating a series of short beeps.

### 5.2 Aborting Options

Most SMCRead operations can be stopped by pressing the ESC key. You are then returned to the options menu.

### 5.3 Battery Voltage Warnings

If when the program is run you get a warning in the module status window that shows the card battery as being 'DEAD!', the battery in the card is close to being, or is completely discharged.

---

**CAUTION**

Do **not** unplug the card from the PC or power down the PC at this stage. Read any data that is present in the card before removing power from the card.

---

It cannot be assumed that the card is powered just because it is in a slot. Exercise caution when changing the battery and ensure that any data present in the card is read first. The normal procedure for replacing the battery is outlined below.

Read all data, remove the card, replace the battery and then rerun SMCRead using the E option to reformat and test the card. When you first plug in a card that has lost power SMCRead indicates that the card is unplugged or corrupted and can show spurious numbers in the module status window. Continue past the warning message and select the E option. After the card has been erased and formatted the module status returns to normal.

If the battery status displays LOW you should change the battery as soon as possible as it may have less than 3% of its original capacity. If the battery status is 'DEAD' a CSM1 will not allow you to store data in the card from a datalogger.

### 5.4 Card Errors

If SMCRead detects that the error counter in a memory card is reading 255, a warning is displayed in the module status window indicating that the card has been corrupted. This means that some of the reserved memory in the card has been corrupted. As the reserved memory area is used to store the various pointers to data in the card, the process of reading data out of the card may be prone to errors.

As a first step in reading a corrupted card, try using the normal procedures to read the data out of the card. If the data files are corrupted, not read, or not complete then you should try to find out where the starting points of the files are by running the Terminal Emulator (T) option. Position the display pointer at the beginning of the card using the 1G command and then scan through the card repeatedly using the NFM command to find the start of each file. Record the position of the start of each file as you go. Once you have determined where each file starts you can use option L to try to read out each file.

If you are unable to recover your data please contact Campbell Scientific for further advice.

## 5.5 Running SMCRead Under Windows 3.1

The current version of SMCRead is not written specifically to run under Windows. It is, however, possible to run the program under Windows in most instances. However, early versions of some of the PCMCIA drivers do not correctly support running DOS programs under Windows. There follows details of known problems with some of the drivers and ways of getting around the driver limitations. If your computer is supplied with a later version of the driver than described it is possible that the problems have been resolved and you should try running the program using either a default DOS PIF file, or the PIF file provided for the Ventura drivers.

The PIF files mentioned below must be linked to the Program Item manually using the Program Manager **'File,Properties'** options and editing the working directory appropriately (refer to your Windows manual for further details). A suitable icon is provided called SMCREAD.ICO, which can also be associated with the Program Item using Program Manager.

### 5.5.1 SystemSoft Drivers

#### ***Socket Services 2.1 Version 1.00, Card Services 2.1 Version 2.05***

These SystemSoft drivers do not perform correctly with DOS programs running under Windows. To use SMCRead under Windows with these drivers you must use the PIF file provided (SMC\_CSO.PIF). This forces the program to run full-screen, and prevents it from running in the background. You should include the CRITICAL switch in the command line; e.g. **SMCRead CRITICAL**. This will prevent Windows from switching tasks while SMCRead is active, and so SMCRead will be the only active task.

### 5.5.2 Ventura/Award Drivers

#### ***Socket Services Level 2.1 Release 1.01, Card Services Level 2.1 Release 1.00***

These drivers function correctly with DOS programs under Windows. The PIF file SMC\_AWD.PIF should be used. It allows SMCRead to run as a window. SMCRead *must not* be invoked with the CRITICAL option. Windows *must* be allowed to switch tasks (so do not mark the application as EXCLUSIVE), so that the Ventura/Award drivers can become the active task when necessary. SMCRead *must* be allowed to operate in the background. This is so that it can perform services when asked to by the Award drivers.

### 5.5.3 Phoenix Drivers

#### ***Socket Services 2.1 Version 1.00, Card Services Release 2.1 Version 1.00***

These Phoenix drivers do not perform correctly with DOS programs running under Windows. To use SMCRead under Windows with these drivers you *must* use the PIF file provided (SMC\_TOS.PIF). This forces the program to run full-screen, and prevents it from running in the background. You must *not* use the CRITICAL switch in the command line.



# Appendix A. PCMCIA Driver Software Notes

---

*These notes are intended as a supplement to the documentation shipped with your PCMCIA drivers. For notes about running MemMaker or other automatic memory optimizers, read the MS DOS manual and also the section at the end of this document, entitled 'Using MemMaker'.*

## A.1 General

The software is designed to work with the PCMCIA drivers: Card Services 2.1, which in turn requires socket services 2.1. Memory Technology Drivers are not required nor is any other special software. Any drivers which fully comply with this standard should work successfully with SMCRead, whether or not it has been tested with them.

The various vendors have different ways of expressing the specification number (2.1) and their own version number. Phoenix Card Services calls its driver 'Release 2.1 version 1.00', yet Ventura Card Services calls its driver '2.1 Release 1.01'. However the various drivers are described, they must be compliant with the PCMCIA Specification 2.10.

If your drivers are prior to version 2.1 you *must* upgrade them. Card Services 2.1 drivers are incompatible with Card Services 2.0 and so any programs written for 2.0 will not work on 2.1. Please contact the supplier of the original PCMCIA drivers for an update, not Campbell Scientific Ltd.

Details of the tested software and specific set-up information follows. Please note that:

amendments to your CONFIG.SYS file or to the configuration of card services will only take effect after the machine is rebooted.

the names given for Socket Services (normally including SS in the driver name) may vary with the hardware of your machine, as there are different drivers to match the specific low-level hardware.

## A.2 Loading the Drivers

### A.2.1 General

If your system drivers are set up correctly you should have no problems running SMCRead. If SMCRead fails with a 'due to lack of resources' message (the actual wording of the message will vary depending upon the driver in use) then the following notes will explain how to adjust your set-up. Specific notes are provided for users with either SystemSoft, Ventura, or Phoenix drivers.

## A.2.2 SystemSoft CardSoft Drivers

### Socket Services 2.1 Version 1.00, Card Services 2.1 Version 2.05

The minimum required drivers for SMCRead are shown as follows as they would appear in your CONFIG.SYS file (your PATH may differ):

```
DEVICE=C:\PCMCIA\SSVADEM.EXE
DEVICE=C:\PCMCIA\CS.EXE
DEVICE=C:\PCMCIA\CSALLOC.EXE C:\PCMCIA\CSALLOC.INI
```

## A.2.3 Ventura Micro CardWare Drivers

### Socket Services Level 2.1 Release 1.01, Card Services Level 2.1 Release 1.00

The minimum required drivers for SMCRead are shown as follows as they would appear in your CONFIG.SYS file (your PATH may differ):

```
DEVICE=C:\PCMCIA\PCSS.EXE
DEVICE=C:\PCMCIA\PCCS.EXE
DEVICE=C:\PCMCIA\PCENABLE.EXE /F=C:\PCMCIA\CARDWARE.INI
```

## A.2.4 Phoenix CardManager Drivers

### Socket Services 2.1 Version 1.00, Card Services Release 2.1 Version 1.00

The minimum required drivers for SMCRead are shown as follows as they would appear in your CONFIG.SYS file (your PATH may differ):

```
DEVICE=C:\PCMCIA\CNFIGNAM.EXE /DEFAULT
DEVICE=C:\PCMCIA\PCMSS.EXE
DEVICE=C:\PCMCIA\PCMCS.EXE
REM the following line is not required, but recommended
DEVICE=C:\PCMCIA\PCMRMAN.SYS
```

## A.3 Adjusting Your CardServices Set-up

CardServices has the ability to map a region of PCMCIA card to a region of PC memory in the high-memory area. This is the same area of memory used by programs loaded high with DEVICEHIGH or LOADHIGH. If all the high memory is allocated then CardServices will not be able to map regions of SRAM cards and SMCRead will fail with an error 'lack of resources'. 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. The memory in that address space is never used. If you have EMM386 loaded (which you will have if you load things into high memory) you must exclude an area of memory for card service.

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) you decide that the memory area D0000 to DFFFF may be used by CardServices, then change your lines as shown below.

```
DEVICE=C:\DOS\HIMEM.SYS
```

```
DEVICE=C:\DOS\EMM386.EXE NOEMS /X=D000-DFFF
```

---

**NOTE**

There is one 0 and one F missing out 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 high performance 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.

**Windows Users:**

In your SYSTEM.INI file there should be a line which looks something like

```
EMM386Exclude=D000-DFFF
```

representing your 'exclude' range.

### A.3.1 How to Choose Which Area of Memory to Allocate to CardServices

The address range used by CardServices can be from A0000 to FFFFF. However not all machines are able to let CardServices 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 CardServices come with their own .INI files, which specify the regions of memory which Card Services will use. Many implementations of card services also have utility programs which automatically detect the free memory and automatically create the .INI file accordingly. This may not be the case with your implementation, or it may be that Card Services will try and use a region which was excluded for use by some other device such as a high

performance graphics card. Details are given here for expressly specifying to Card Services which region it is to use.

If your system is already set up as described below and SMCRead still fails due to lack of resources, then 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.

### A.3.2 SystemSoft CardSoft Card Services Driver

There are two .INI files, CARDID.INI and CSALLOC.INI. It is only CSALLOC.INI we are concerned with. A sample file is shown below:

```
MEM=D000-DFFF
IOP=100-16F,178-1EF,1F8-1FF,208-2E7,2F0-2F7,300-36F,380-3BF
RIOP=170-177,2E8-2EF,2F8-2FF,370-377,3E8-3F7
IRQ=3,5,A-B,F
```

The only setting that will need changing is the MEM line. There is also a program called CSALLOC.EXE. To automatically generate the CSALLOC.INI file from your PCMCIA drivers directory simply type:  
`csalloc /g`

### A.3.3 Phoenix CardManager Card Services Driver

Phoenix drivers ship on many Toshiba and IBM machines. SMCRead has been tested with these drivers. The standard install for these drivers does not provide a big enough range for SMCRead to work so Phoenix users probably will need to increase the size of the range.

The .INI file is PCM.INI. It is not a simple file. However a program PCMRMAN is provided which lets you alter this file in a friendly way, so you should not need to edit it directly if you do not wish to.

While running PCMRMAN, if you choose to DISPLAY MEMORY ranges, you will plainly see which regions are available. You may choose to MODIFY MEMORY ranges, and you may SAVE the set-up using the FILE menu.

The driver CNFIGNAM is good at automatically detecting the right configuration so you may not have to change the INI file at all, (provided you have excluded EMM386 from using an address that your machine is capable of using!).

If you do increase the exclude range in EMM386, but SMCRead still fails due to lack of resources, run PCMRMAN to see if CardServices is making use of those memory ranges. If it is not, you will need to change USER-OVERRIDE on that range to INCLUDE the range. Then from the FILE menu save the changes to create a new .INI file. After a reboot, SMCRead should function correctly.

### A.3.4 Ventura/Award Card Services Driver

The .INI file is CARDWARE.INI. The format of this file is explained in the file CWINI.TXT. The Ventura drivers that SMCRead was tested with did not auto detect memory ranges, so you must edit CARDWARE.INI. This is not a fearsome task however.

The memory range is specified by a line, near the top of the file, such as

[Resources]

**MEM=0xD0000,0x10000,E**

The range is a FULL address and not a SEGMENT address as in the EMM386 line in the CONFIG.SYS file. The second number is not the end of the range but the size of the range. The 0x signifies the number is hexadecimal. The above example specifies the range D0000 to DFFFF. If the size was 0x9000 the range specified would be D0000 to D8FFF.

Finally, the E must be supplied, unless you already know of a good reason not to do so.

## A.4 Using MemMaker (supplied with MS-DOS 6.xx)

This section does not apply specifically to SMCRead, but you may wish to run MemMaker after adjusting the CardServices memory range. These notes will help you to that end.

Before running MemMaker, make sure that you specify the exclude range on the EMM386 line of CONFIG.SYS. If you have any drivers which must *not* be loaded into high memory you should follow the instructions below for Phoenix users.

### A.4.1 Notes for those with Phoenix Drivers

You should choose CUSTOM SET-UP when running MemMaker, and make sure the following options are set:

- Specify which drivers and TSRs to include in optimization **YES**
- Keep current EMM386 memory inclusions and exclusion **YES** (default)

Because CNFIGNAM cannot be loaded high, you should tell MemMaker (when asked) that CNFIGNAME is not to be included in the optimization process.

### A.4.2 Notes for those with Ventura/Award or SystemSoft Drivers

All of your drivers can be loaded high, and so you may run MemMaker in express mode.



# ***Appendix B. Command Line Operation***

---

## **B.1 Introduction**

All options in SMCRead can be specified as command line parameters when the program is run. This enables semi-automated reading of the cards as SMCRead can be called from a DOS batch file.

The parameters should be entered in the order that they would be entered when running SMCRead. If you wish, you can spell out the name of the option selected in full, as only the first character of the option is read by SMCRead.

An example is:

```
SMCRead all dfiles comma programs pfiles download prog1 1 q
```

This is equivalent to:

```
SMCRead a dfiles c p pfiles d prog1 1 q
```

This executes SMCRead and all data files are collected and stored in files with a root name of DFILES.DAT in comma delineated format. All the program files are then collected and stored in a series of files with a root name PFILES. A new program stored in PROG1.DLD is then loaded into program area 1 in the card. Finally, SMCRead quits.

There are some special command line parameter options (see below) which should be entered before parameters which you wish to use to select normal menu options.

## **B.2 Additional Command Line Parameter Options**

### **NOCOLOR**

If poor contrast is apparent as a result of trying to simulate SMCRead's screen colors when running SMCRead on a computer with an LCD screen, the contrast may be improved by entering NOCOLOR before menu commands.

### **Q**

If you enter Q before menu command line parameters SMCRead runs in quiet mode, i.e. it does not beep.

### **Slot n or Socket n**

Specifies the default slot/socket number n.

## Critical

This parameter is used when running SMRead under Windows to prevent task switching (see section 5.5).

### NOTE

---

SMRead has the same capability as GraphTerm, CSMCOM and SMCOM to send character strings.

---



# ***Appendix C. SMCRead Error Messages***

---

**Error: No card found in slot number #**

The PC cannot find the card in the slot specified.

**Error writing data to filename, check disk space and try again.**

A file system error has occurred when writing a program file to disk. Check for available disk space or disk corruption.

**Program nn not found.**

There is no program nn in the card.

**Error - no room on card.**

Shown when loading a .DLD program into a memory card when there is not enough space left.

**Directory does not exist**

When requesting the program to display a list of files in a specific directory you have entered an invalid directory name.

**No matching files found**

When requesting the program to display a list of files using a wildcard character, no files of that type could be found.

**Inadequate disk space available!**

When collecting data the program has found there is not enough space to write the next data block to disk.

**Filename: xxxxxxxx is too long - maximum 6 characters.**

The file name you have specified is too long. There must be space for the 2-digit index that SMCRead appends to the end of the prefix of the name.

**Error opening xxxxxxxx check if the file exists, or is empty.**

SMCRead was unable to open a specified file.

**Error creating xxxxxxxx check filename.**

When creating a new file a disk error has occurred. This may be because you have specified an invalid file name.

**Error writing data to disk, collection aborted.**

An unforeseen error has occurred when writing data to disk.

**No data stored in the card!**

The newest file does not contain any data!

**Invalid location specified in command line parameter!**

For the L option you have specified an invalid display location number in the DOS command line.

**Unable to move the display pointer to that location! - check validity.**

The value entered may be too large for the card size in use.

**No data found in the area specified**

Self explanatory - only displayed when using the L option.

**No data stored in this area.**

Displayed when collecting several files. This usually means a program file was encountered between two data files.

**No data found in the card!, or file error.**

After attempting to collect several files, the program has not successfully written any data to disk.

**No uncollected data found in the card!**

**Error: no such command**

While in the 'Terminal Emulator' you have typed an incorrect command.

**The Card in the socket is write protected**

The switch on the card is set to write protect. It should be set to the unprotected position for operation with SMCRead.

**Card Services Error Code xxx - xxxxxxxxxxxx**

SMCRead has encountered an error when communicating with the card services driver. There are many errors that could be reported, each detailing a possible cause. Please contact Campbell Scientific if the solution is not evident.