CR510 DATALOGGER
OPERATOR'S MANUAL

9/01

CR510 DATALOGGER OVERVIEW

The CR510 is a fully programmable datalogger/controller with non-volatile memory and a battery backed clock in a small, rugged module. The combination of reliability, versatility, and telecommunications support make it a favorite choice for networks and single logger applications.

Campbell Scientific Inc. provides four aids to operating the CR510:

1. *This Overview*
2. *The CR510 Operator's Manual*
3. *The CR510 Prompt Sheet*
4. *Short Cut*

This Overview introduces the concepts required to take advantage of the CR510's capabilities. Hands-on programming examples start in Section OV5. Working with a CR510 will help the learning process, so don't just read the examples, do them. If you want to start this minute, go ahead and try the examples, then come back and read the rest of the Overview.

The sections of the Operator's Manual which should be read to complete a basic understanding of the CR510 operation are the Programming Sections 1-3, the portions of the data retrieval Sections 4 and 5 appropriate to the method(s) you are using (see OV6), and Section 14 which covers installation and maintenance.

Section 6 covers details of serial communications. Sections 7 and 8 contain programming examples. Sections 9-12 have detailed descriptions of each programming instruction, and Section 13 goes into detail on the CR510 measurement procedures.

The Prompt Sheet is an abbreviated description of the programming instructions. Once familiar with the CR510, it is possible to program it using only the Prompt Sheet as a reference, consulting the manual if further detail is needed.

Short Cut is an easy-to-use DOS-based software program. It features point-and-click menus to guide you through the process of creating simple CR510 programs. In addition to the downloadable program file, Short Cut creates a table to simplify wiring sensors to the CR510.

Read the Selected Operating Details and Cautionary Notes at the front of the Manual before using the CR510.

OV1. PHYSICAL DESCRIPTION

The CR510 was designed to provide a rugged datalogger with a low per unit cost. Some of its distinguishing physical features are:

- The CR510 does not have an integral keyboard/display. The user accesses the CR510 with the portable CR10KD Keyboard Display or with a computer or terminal (Section OV2).

- The power supply is external to the CR510. This gives the user a wide range of options (Section 14) for powering the CR510.

OV1.1 ANALOG INPUTS

The terminals labeled 1H to 4L are analog inputs. These numbers refer to the high and low inputs to the differential channels 1 and 2. In a differential measurement, the voltage on the H input is measured with respect to the voltage on the L input. When making single-ended measurements, either the H or L input may be used as an independent channel to measure voltage with respect to the CR510 analog ground (AG). The single-ended channels are numbered sequentially starting with 1H; e.g., the H and L sides of differential channel 1 are single-ended channels 1 and 2; the H and L sides of differential channel 2 are single-ended channels 3 and 4, etc.

OV1.2 EXCITATION OUTPUTS

The terminals labeled E1, and E2 are precision, switched excitation outputs used to supply programmable excitation voltages for resistive bridge measurements. DC or AC excitation voltages between -2500 mV and +2500 mV are user programmable (Section 9).

OV1.3 PULSE INPUTS

The terminals labeled P1, P2, and P3 are the pulse counter inputs for the CR510. P1 and P2 are programmable for high frequency pulse, low level AC, or switch closure (Section 9, Instruction 3). C2/P3 can be configured to count switch closures up to 40 Hz.

OV1.4 DIGITAL I/O PORTS

Terminal C1 is a digital Input/Output port. On power-up it is configured as an input port, commonly used for reading the status of an external signal. High and low conditions are: 3V < high < 5.5V; -0.5V < low < 0.8V.

Configured as output the port allows on/off control of external devices. A port can be set high (5V ± 0.1V), set low (<0.1V), toggled or pulsed (Sections 3, 8.3, and 12).

Port C2/P3 can be configured as pulse counters for switch closures (Section 9, Instruction 3) or used to trigger subroutine execution (Section 1.1.2), or serial SDI-12 communication.

OV1.5 ANALOG GROUND (AG)

The AG terminals are analog grounds, used as the reference for single-ended measurements and excitation return.

OV1.6 12V, POWER GROUND (G), AND EARTH TERMINALS

The 12V and power ground (G) terminals are used to supply 12V DC power to the datalogger. The extra 12V and G terminals can be used to connect other devices requiring 12V power.

The G terminals are also used to tie cable shields to ground, and to provide a ground reference for pulse counters and binary inputs. The G terminals are directly connected to the Earth terminal. For protection against transient voltage spikes, Earth Ground should be connected to a good earth ground (Section 14.7.1).

OV1.7 5V OUTPUT

The 5V (±0.2%) output is commonly used to power peripherals such as the QD1 Incremental Encoder Interface and AVW1 Vibrating Wire Interface.

The 5V output is common with pin 1 on the 9 pin serial connector; 200 mA is the maximum combined current output.

OV1.8 SERIAL I/O

The 9 pin serial I/O port contains lines for serial communication between the CR510 and external devices such as computers, printers, Storage Modules, etc. **This port does NOT have the same configuration as the 9 pin serial ports currently used on many personal computers**. It has a 5VDC power line which is used to power peripherals such as the Storage Modules. The same 5VDC supply is used for the 5V output on the terminal strip. It also has a continuous 12 V power supply on pin 8 for external communication devices such as the COM200 and COM300. Section 6 contains technical details on serial communication.

OV1.9 CONNECTING POWER TO THE CR510

The CR510 can be powered by any 12VDC source. The green power connector is a plug in connector that allows the power supply to be easily disconnected without unscrewing the terminals. The Terminal Strip power connection is reverse polarity protected. See Section 14 for details on power supply connections.

> **CAUTION**: The metal surfaces of the CR510 Terminal Strip, and CR10KD Keyboard Display are at the same potential as power ground. To avoid shorting 12 volts to ground, connect the 12 volt lead first, then connect the ground lead.

When primary power falls below 9.6 VDC, the CR510 stops executing its programs. The Low Voltage Counter (∗B window 9) is incremented by one each time the primary power falls below 9.6 VDC and E10 is displayed on the CR10KD. A double dash (--) in the 9th window of the ∗B mode indicates that the CR510 is currently in a low primary power mode. (Section 1.6)

The datalogger program and stored data remain in memory, and the clock continues to keep time when power is disconnected. The clock and Static Random Access Memory (SRAM) are powered by an internal lithium battery.

OV2. MEMORY AND PROGRAMMING CONCEPTS

OV2.1 INTERNAL MEMORY

The standard CR510 has 128 K of Flash Electrically Erasable Programmable Read Only Memory (EEPROM) and 128 K Static Random Access Memory (SRAM). The Flash EEPROM stores the operating system and user programs. RAM is used for data and running the program. Data Storage can be expanded with an optional Flash EEPROM (Figure OV2.1-1). The use of the Input, Intermediate, and Final Storage in the measurement and data processing sequence is shown in Figure OV2.1-2. The five areas of SRAM are:

1. **System Memory** - used for overhead tasks such as compiling programs, transferring data, etc. The user cannot access this memory.

2. **Program Memory** - available for user entered programs.

3. **Input Storage** - Input Storage holds the results of measurements or calculations. The ∗6 Mode is used to view Input Storage locations for checking current sensor readings or calculated values. Input Storage defaults to 28 locations. Additional locations can be assigned using the ∗A Mode.

4. **Intermediate Storage** - Certain Processing Instructions and most of the Output Processing Instructions maintain intermediate results in Intermediate Storage. Intermediate storage is automatically accessed by the instructions and cannot be accessed by the user. The default allocation is 64 locations. The number of locations can be changed using the ∗A Mode.

5. **Final Storage** - Final processed values are stored here for transfer to printer, solid state Storage Module or for retrieval via telecommunication links. Values are stored in Final Storage only by the Output Processing Instructions and only when the Output Flag is set in the user's program. Approximately 62,000 locations are allocated to Final Storage on power up. This number is reduced if Input or Intermediate Storage is increased.

While the total size of these areas remains constant, memory may be reallocated between the areas to accommodate different measurement and processing needs (∗A Mode, Section 1.5).

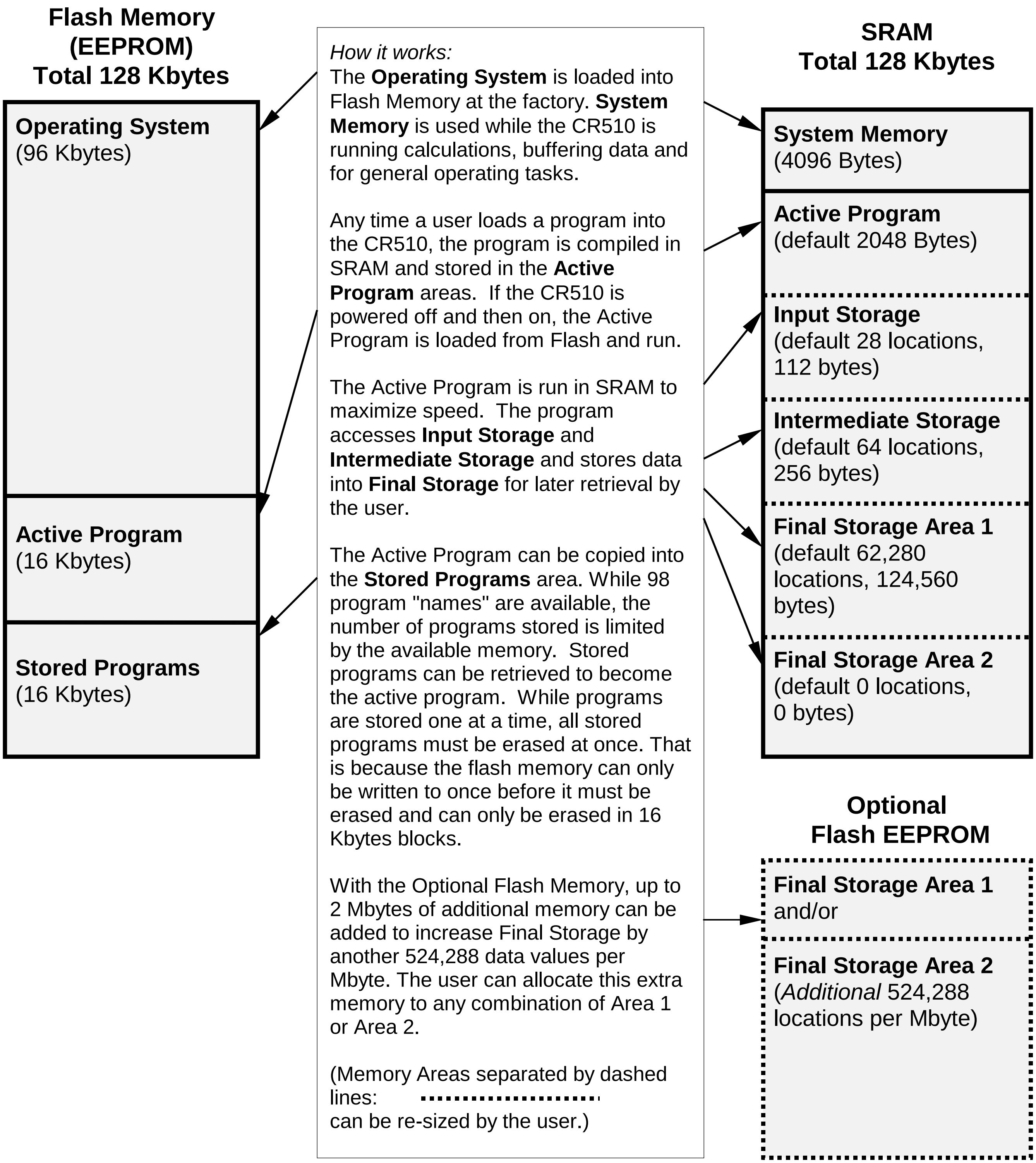


FIGURE OV2.1-1. CR510 Memory

OV2.2 PROGRAM TABLES, EXECUTION INTERVAL AND OUTPUT INTERVALS

The CR510 must be programmed before it will make any measurements. A program consists of a group of instructions entered into a **program table**. The program table is given an **execution interval** which determines how frequently that table is executed. When the table is executed, the instructions are executed in sequence from beginning to end. After executing the table, the CR510 waits the remainder of the execution interval and then executes the table again starting at the beginning.

The interval at which the table is executed generally determines the interval at which the sensors are measured. The interval at which data are stored is separate from how often the table is executed, and may range from samples every execution interval to processed summaries output hourly, daily, or on longer or irregular intervals.

Programs are entered in Tables 1 and 2. Subroutines, called from Tables 1 and 2, are entered in Subroutine Table 3. The size of program memory can be fixed or automatically allocated by the CR510 (Section 1.5).

Table 1 and Table 2 have independent execution intervals, entered in units of seconds with an allowable range of 1/8 to 8191 seconds. Subroutine Table 3 has no execution interval; subroutines are only executed when called from Table 1 or 2.

OV2.2.1 THE EXECUTION INTERVAL

The execution interval specifies how often the program in the table is executed, which is usually determined by how often the sensors are to be measured. *Unless two different measurement rates are needed, use only one table*. A program table is executed sequentially starting with the first instruction in the table and proceeding to the end of the table.

Table 1. Execute every x sec. $0.125 \le x \le 8191$
Instructions are executed sequentially in the order they are entered in the table. One complete pass through the table is made each execution interval unless program control instructions are used to loop or branch execution.
Normal Order: MEASURE PROCESS CHECK OUTPUT COND. OUTPUT PROCESSING

Table 2. Execute every y sec. $0.125 \le y \le 8191$
Table 2 is used if there is a need to measure and process data on a separate interval from that in Table 1.

Table 3. Subroutines
A subroutine is executed only when called from Table 1 or 2.
Subroutine Label Instructions End
Subroutine Label Instructions End
Subroutine Label Instructions End

FIGURE OV2.2-1. Program and Subroutine Tables

Each instruction in the table requires a finite time to execute. If the execution interval is less than the time required to process the table, an execution interval overrun occurs; the CR510 finishes processing the table and waits for the next execution interval before initiating the table. When an overrun occurs, decimal points are shown on either side of the G on the display in the LOG mode (*0). Overruns and table priority are discussed in Section 1.1.

OV2.2.2. THE OUTPUT INTERVAL

The interval at which output occurs must be an integer multiple of the execution interval (e.g., a table cannot have a 10 minute execution interval and output every 15 minutes).

A single program table can have many different output intervals and conditions, each with a unique data set (Output Array). Program Control Instructions are used to set the Output Flag. The Output Processing Instructions which follow the instruction setting the Output Flag determine the data output and its sequence. Each additional Output Array is created by another Program Control Instruction checking a output condition, followed by Output Processing Instructions defining the data set to output.

OV2.3 CR510 INSTRUCTION TYPES

Figure OV2.3-1 illustrates the use of three different instruction types which act on data. The fourth type, Program Control, is used to control output times and vary program execution. Instructions are identified by numbers.

1. **INPUT/OUTPUT INSTRUCTIONS** (1-12, 16-29, 105-106, 114, 117, 130, 131, Section 9) control the terminal strip inputs and outputs (Figure OV1.1-2), storing the results in Input Storage (destination). Multiplier and offset parameters allow conversion of linear signals into engineering units. The Digital I/O Ports are also addressed with I/O Instructions.

2. **PROCESSING INSTRUCTIONS** (30-68, Section 10) perform numerical operations on values located in Input Storage and store the results back in Input Storage. These instructions can be used to develop high level algorithms to process measurements prior to Output Processing.

3. **OUTPUT PROCESSING INSTRUCTIONS** (69-82, Section 11) are the only instructions which store data in Final Storage. Input Storage values are processed over time to obtain averages, maxima, minima, etc. There are two types of processing done by Output Instructions: **Intermediate** and **Final**.

   Intermediate processing normally takes place each time the instruction is executed. For example, when the Average Instruction is executed, it adds the values from the input locations being averaged to running totals in Intermediate Storage. It also keeps track of the number of samples.

   Final processing occurs only when the Output Flag is high (Section 3.7.1). The Output Processing Instructions check the Output Flag. If the flag is high, final values are calculated and output. With the Average, the totals are divided by the number of samples and the resulting averages sent to Final Storage. Intermediate locations are zeroed and the process starts over. *The Output Flag, Flag 0, is set high by a Program Control Instruction which must precede the Output Processing Instructions in the user entered program.*

4. **PROGRAM CONTROL INSTRUCTIONS** (83-98, 111, 120-121, Section 12) are used for logic decisions, conditional statements, and to send data to peripherals. They can set flags and ports, compare values or times, execute loops, call subroutines, conditionally execute portions of the program, etc.

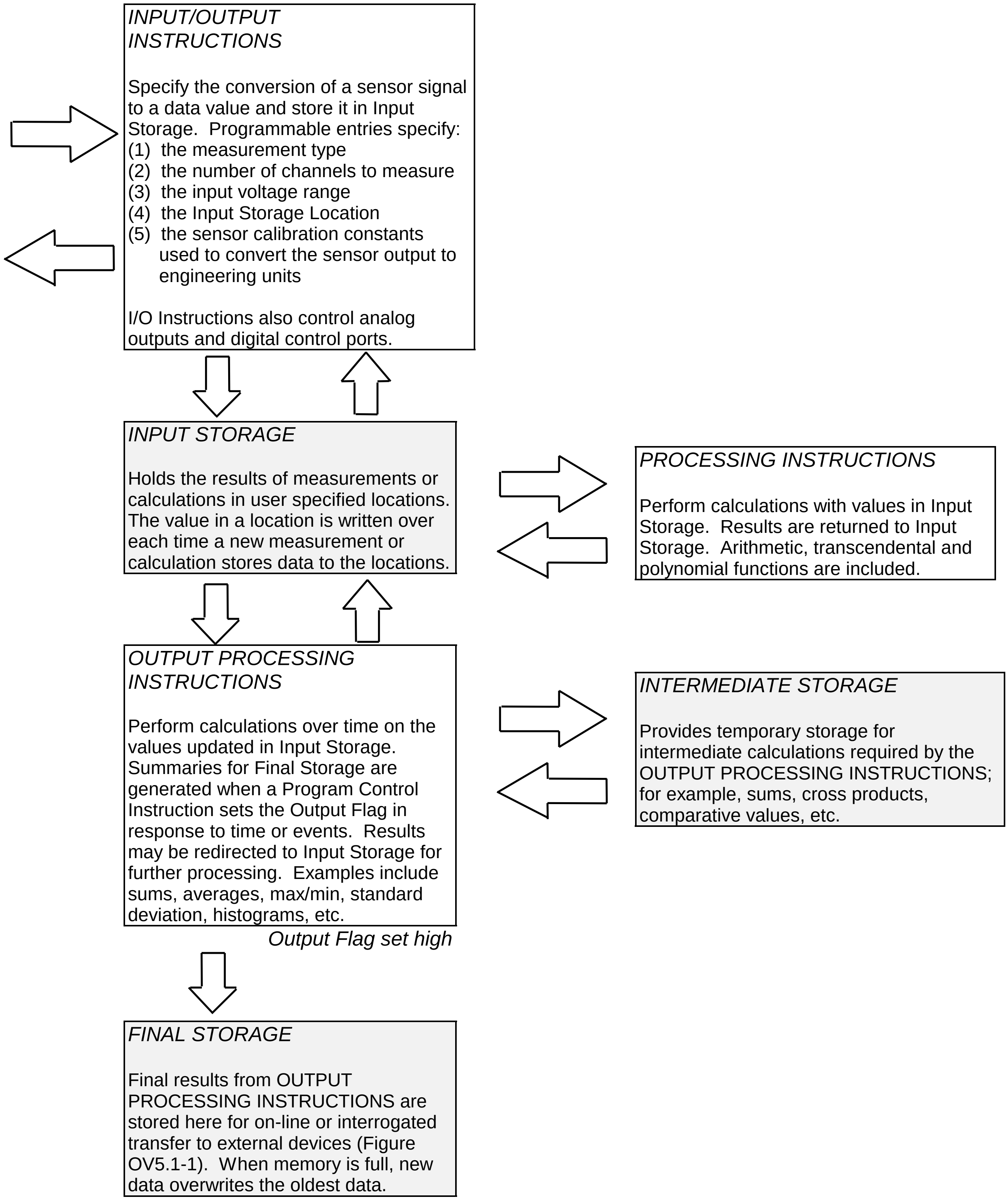


FIGURE OV2.3-1. Instruction Types and Storage Areas

OV3. COMMUNICATING WITH CR510

An external device must be connected to the CR510's Serial I/O port to communicate with the CR510. This may be either Campbell Scientific's CR10KD Keyboard Display or a computer/terminal.

The CR10KD is powered by the CR510 and connects directly to the serial port via the SC12 cable (supplied with the CR10KD). No interfacing software is required.

Computer communication and program editing is accomplished using Campbell Scientific's datalogger support software. This package contains a program editor (EDLOG), datalogger communications, automated telecommunications data retrieval, a data reduction program, and programs to retrieve data from Campbell Scientific Storage Modules.

To participate in the programming examples (Section OV5) you must communicate with the CR510. Read Section OV3.1 if the CR10KD is being used or Section OV3.2 if datalogger support software is being used.

OV3.1 KEYBOARD/DISPLAY

The SC12 cable (supplied with the CR10KD) is used to connect the Keyboard/Display to the 9 pin Serial I/O port on the CR510.

If the Keyboard/Display is connected to the CR510 prior to being powered up, the "HELLO" message is displayed while the CR510 checks memory. The total size of memory is then displayed (256 for 256 K bytes of memory). When the CR10KD is plugged in after the CR510 has powered up, the display is meaningless until "∗" is pressed to enter a mode.

This manual describes direct interaction with the CR510. If you have a CR10KD, work through the direct programming examples in this overview in addition to using EDLOG and you will have the basics of CR510 operation as well as an appreciation for the help provided by the software.

OV3.1.1 FUNCTIONAL MODES

CR510/User interaction is broken into different functional MODES (e.g., programming the measurements and output, setting time, manually initiating a block data transfer to Storage Module, etc.). The modes are referred to as Star (∗) Modes since they are accessed by first keying ∗, then the mode number or letter. Table OV3.1-1 lists the CR510 Modes.

TABLE OV3.1-1. ∗ Mode Summary

Key	Mode
∗ 0	LOG data and indicate active Tables
∗ 1	Program Table 1
∗ 2	Program Table 2
∗ 3	Program Table 3, subroutines only
∗ 4	Parameter Entry Table
∗ 5	Display/set real time clock
∗ 6	Display/alter Input Storage data, toggle flags or control ports.
∗ 7	Display Final Storage data
∗ 8	Final Storage data transfer to peripheral
∗ 9	Storage Module commands
∗ A	Memory allocation/reset
∗ B	Signature/status
∗ C	Security
∗ D	Save/load Program
∗ #	Used with TGT1 satellite transmitter

OV3.1.2 KEY DEFINITION

Keys and key sequences have specific functions when using the CR10KD keyboard or a computer/terminal in the remote keyboard state (Section 5). Table OV3.1-2 lists these functions. In some cases, the exact action of a key depends on the mode the CR510 is in and is described with the mode in the manual.

TABLE OV3.1-2 Key Description/Editing Functions

Key	Action
0 - 9	Key numeric entries into display
*	Enter Mode (followed by Mode Number)
A	Enter/Advance
B	Back up
C	Change the sign of a number or index an input location to loop counter
D	Enter the decimal point
#	Clear the rightmost digit keyed into the display
# A	Advance to next instruction in program table (*1, *2, *3) or to next Output Array in Final Storage (*7)
# B	Back up to previous instruction in program table or to previous Output Array in Final Storage
# D	Delete entire instruction
# 0	(then A) Back up to the start of the current array.

When using a computer/terminal to communicate with the CR510 (Telecommunications remote keyboard state) there are some keys available in addition to those found on the CR10KD. Table OV3.1-3 lists these keys.

TABLE OV3.1-3. Additional Keys Allowed in Telecommunications

Key	Action
-	Change Sign, Index (same as C)
CR	Enter/advance (same as A)
:	Colon (used in setting time)
S or ^S	Stops transmission of data (10 second time-out; any character restarts)
C or ^C	Aborts transmission of Data

OV3.2 USING COMPUTER WITH DATALOGGER SUPPORT SOFTWARE

Direct datalogger communication programs in the datalogger support software provide menu selection of tools to perform the datalogger functions (e.g., set clock, send program, monitor measurements, and collect data). The user also has the option of directly entering keyboard commands via a built-in terminal emulator (Section OV3.3).

When using the support software, the computer's baud rate, port, and modem types are specified and stored in a file for future use.

The simplest and most common interface is the SC32A Optically Isolated RS232 Interface. The SC32A converts and optically isolates the voltages passing between the CR510 and the external terminal device.

The SC12 Two Peripheral cable which comes with the SC32A is used to connect the serial I/O port of the CR510 to the 9 pin port of the SC32A labeled "Datalogger". Connect the "Terminal/Printer" port of the SC32A to the serial port of the computer with a straight 25 pin cable or, if the computer has a 9 pin serial port, a standard 9 to 25 pin adapter cable.

OV3.3 ASCII TERMINAL OR COMPUTER WITH TERMINAL EMULATOR

Devices which can be used to communicate with the CR510 include standard ASCII terminals and computers programmed to function as a terminal emulator. See Section 6.7 for details.

To communicate with any device other than the CR10KD, the CR510 enters its Telecommunications Mode and responds only to valid telecommunications commands. Within the Telecommunications Mode, there are 2 "states"; the Telecommunications Command state and the Remote Keyboard state. Communication is established in the Telecommunications command state. One of the commands is to enter the Remote Keyboard state (Section 5).

The Remote Keyboard state allows the keyboard of the computer/terminal to act like the CR10KD keyboard. Various datalogger modes may be entered, including the mode in which programs may be keyed in to the CR510 from the computer/terminal.

OV4. PROGRAMMING THE CR510

A datalogger program is created on a computer using EDLOG or one of the programming aids such as Short Cut. A program can also be entered directly into the datalogger. Section OV4.3 describes options for loading the program into the CR510.

OV4.1 PROGRAMMING SEQUENCE

In routine applications, the CR510 measures sensor output signals, processes the measurements over some time interval and stores the processed results. A generalized programming sequence is:

1. Enter the execution interval. In most cases, the execution interval is determined by the desired sensor scan rate.

2. Enter the Input/Output instructions required to measure the sensors.

3. If processing in addition to that provided by the Output Processing Instructions (step 5) is required, enter the appropriate Processing Instructions.

4. Enter the Program Control Instruction to test the output condition and set the Output Flag when the condition is met. For example, use

   Instruction 92 to output based on time.

   Instruction 86 to output every execution interval.

   Instruction 88 or 89 to output based on a comparison of values in input locations.

   This instruction must precede the Output Processing Instructions which store data in Final Storage. Instructions are described in Sections 9 through 12.

5. Enter the Output Processing Instructions to store processed data in Final Storage. The order in which data are stored is determined by the order of the Output Processing Instructions in the table.

6. Repeat steps 4 and 5 for additional outputs on different intervals or conditions.

> **NOTE**: The program must be executed for output to occur. Therefore, the interval at which the Output Flag is set must be evenly divisible by the execution interval. For example, with a 2 minute execution interval and a 5 minute output interval, the program will only be executed on the even multiples of the 5 minute intervals, not on the odd. Data will be output every 10 minutes instead of every 5 minutes.

Execution intervals and output intervals set with Instruction 92 are synchronized with datalogger time starting at midnight.

OV4.2 INSTRUCTION FORMAT

Instructions are identified by an instruction number. Each instruction has a number of parameters that give the CR510 the information it needs to execute the instruction.

The CR510 Prompt Sheet has the instruction numbers in red, with the parameters briefly listed in columns following the description. Some parameters are footnoted with further description under the "Instruction Option Codes" heading.

For example, Instruction 73 stores the maximum value that occurred in an Input Storage location over the output interval.

```
P73  Maximum
     1:    Reps
     2:    TimeOption
     3:    Loc
```

The instruction has three parameters (1) REPetitionS, the number of sequential Input Storage locations on which to find maxima, (2) TIME, an option of storing the time of occurrence with the maximum value, and (3) LOC the first Input Storage location operated on by the Maximum Instruction. The codes for the TIME parameter are listed in the "Instruction Option Codes".

The repetitions parameter specifies how many times an instruction's function is to be repeated. For example, four 107 thermistor probes may be measured with a single Instruction 11, Temp-107, with four repetitions. Parameter 2 specifies the input channel of the first thermistor (the probes must be connected to sequential channels). Parameter 4 specifies the Input Storage location in which to store measurements from the first thermistor. If location 5 were used and the first probe was on channel 1, the temperature of the thermistor on channel 1 would be stored in input location 5, the temperature from channel 2 in input location 6, etc.

Detailed descriptions of the instructions are given in Sections 9-12. Entering an instruction into a program table is described in OV5.

OV4.3 ENTERING A PROGRAM

Programs are entered into the CR510 in one of three ways:

1. Keyed in using the CR10KD keyboard.
2. Loaded from a pre-recorded listing using the ∗D Mode. There are 2 types of storage/input:
   a. Stored on disk/sent from computer.
   b. Stored/loaded from Storage Module.
3. Loaded from internal Flash Memory or Storage Module upon power-up.

A program is created by keying it directly into the datalogger as described in Section OV5, or on a PC using EDLOG or a programming aid such as Short Cut.

Program files (.DLD) can be downloaded directly to the CR510 using Campbell's datalogger support software. Communication via direct wire, telephone, or Radio Frequency (RF) is supported.

Programs can be copied to a Storage Module with the appropriate software. Using the ∗D Mode to save or load a program from a Storage Module is described in Section 1.8.

Once a program is loaded in the CR510, the program will be stored in flash memory and will automatically be loaded and run when the datalogger is powered-up.

The program on power up function can also be achieved by using a Storage Module. Up to 8 programs can be stored in the Storage Module, the programs may be assigned any of the numbers 1-8. If the Storage Module is connected when the CR510 is powered-up the CR510 will automatically load program number 8, provided that a program 8 is loaded in the Storage Module (Section 1.8). The program from the Storage Module will replace the active program in flash memory.

OV5. PROGRAMMING EXAMPLES

The following examples stress direct interaction with the CR510 using the CR10KD. At the beginning of each example is an EDLOG listing of the program. You can also participate in the example by entering the program in EDLOG and sending it to the CR510 and viewing measurements with Campbell's datalogger support software. If you have the CR10KD, work through the examples as well as using EDLOG. You will learn the basics of CR510 operation as well as an appreciation for the help provided by the software.

We will start with a simple programming example. There is a brief explanation of each step to help you follow the logic. When the example uses an instruction, find it on the Prompt Sheet and follow through the description of the parameters. Using the Prompt Sheet while going through these examples will help you become familiar with its format. Sections 9-12 have more detailed descriptions of the instructions.

Connect the CR510 to the CR10KD Keyboard/Display or a terminal (Section OV3). Hook up the power leads as described in Section OV1.2. The programming steps in the following examples use the keystrokes possible on the keyboard/display. With a terminal, some responses will be slightly different.

If the CR10KD is connected to the CR510 when it is powered up, the display will show:

Display	Explanation
HELLO	On power-up, the CR510 displays "HELLO" while it checks the memory (this display occurs only with the CR10KD).

after a few seconds delay

:0256	The size of the machine's total memory, 256 K (1280 if 1 meg option).

When primary power is applied to the CR510, it tests the FLASH memory and loads the current program to RAM. After the program compiles successfully, the CR510 begins executing the program. If the ring line on the 9 pin connector is raised while the CR510 is testing memory, there will be a 128 second delay before compiling and running the program. This can be used to edit or change the program before it starts running. To raise the ring line, press any key on the CR10KD keyboard display or call the CR510 with the computer during the power up sequence (i.e., while "HELLO" is displayed on the CR10KD).

In order to ensure that there is no active program in the CR510, we will load an empty program using the *D Mode:

Key	Display Will Show: (ID:Data)	Explanation
*	00:00	Enter mode
D	13:00	Enter *D Mode
7	13:7	7 is command to load program from flash
A	07:00	Execute command 7, CR510 is ready for program number
0	07:0	Load Program 0 (empty program)
A		Execute program load, after a short wait, the display will show
	13:0000	Indicating that the command is complete.

OV5.1 SAMPLE PROGRAM 1

EDLOG Listing Program 1:

```
*Table 1 Program
01:       5.0       Execution Interval (seconds)

1: Internal Temperature (P17)
  1:       1        Loc [ CR510Temp ]

2: Do (P86)
  1:      10        Set Output Flag High

3: Sample (P70)
  1:       1        Reps
  2:       1        Loc [ CR510Temp ]
```

In this example the CR510 is programmed to read its own internal temperature (using a built in thermistor) every 5 seconds and to send the results to Final Storage.

Key	Display Will Show: (ID:Data)	Explanation
*	00:00	Enter mode.
1	01:0000	Enter Program Table 1.
A	01:0.0000	Advance to execution interval (In seconds)
5	01:5	Key in an execution interval of 5 seconds.
A	01:P00	Enter the 5 second execution interval and advance to the first program instruction location.
1 7	01:P17	Key in Instruction 17 which directs the CR510 to measure the internal temperature in degrees C. This is an Input/Output Instruction.
A	01:0000	Enter Instruction 17 and advance to the first parameter.
1	01:1	The input location to store the measurement, location 1.
A	02:P00	Enter the location # and advance to the second program instruction.

The CR510 is now programmed to read the internal temperature every 5 seconds and place the reading in Input Storage Location 1. The program can be compiled and the temperature displayed.

Key	Display Will Show: (ID:Data)	Explanation
* 0	LOG 1	Exit Table 1, enter *0 Mode, compile table and begin logging.
* 6	06:0000	Enter *6 Mode (to view Input Storage).
A	01:21.234	Advance to first storage location. Internal datalogger temp. is 21.234ºC (display shows actual temperature so exact value will vary).

Wait a few seconds:

Key	Display	Explanation
	01:21.423	The CR510 has read the sensor and stored the result again. The internal temp is now 21.423 ^oC. The value is updated every 5 seconds when the table is executed. At this point the CR510 is measuring the temperature every 5 seconds and sending the value to Input Storage. No data are being saved. The next step is to have the CR510 send each reading to Final Storage. (Remember, the Output Flag must be set first.)
∗ 1	01:0000	Exit ∗6 Mode. Enter program table 1.
2 A	02:P00	Advance to 2nd instruction location (this is where we left off).
8 6	02:P86	This is the DO instruction (a Program Control Instruction).
A	01:00	Enter 86 and advance to the first parameter (which will specify the command to execute).
1 0	01:10	This command sets the Output Flag. (Flag 0)
A	03:P00	Enter 10 and advance to third program instruction.
7 0	03:P70	The SAMPLE instruction. It directs the CR510 to take a reading from an Input Storage location and send it to Final Storage (an Output Processing Instruction).
A	01:0000	Enter 70 and advance to the first parameter (repetitions).
1	01:1	There is only one input location to sample; repetitions = 1.
A	02:0000	Enter 1 and advance to second parameter (Input Storage location to sample).
1	02:1	Input Storage Location 1, where the temperature is stored.
A	04:P00	Enter 1 and advance to fourth program instruction.
∗	00:00	Exit Table 1.
0	LOG 1	Enter ∗0 Mode, compile program, log data.

The CR510 is now programmed to measure the internal temperature every 5 seconds and send each reading to Final Storage. Values in Final Storage can be viewed using the ∗7 Mode.

Key	Display Will Show: (ID:Data)	Explanation
∗ 7	07: 13.000	Enter ∗7 Mode. The Data Storage Pointer (DSP) is at Location 13 (in this example).
A	01: 0102	Advance to the first value, the Output Array ID. 102 indicates the Output Flag was set by the second instruction in Program Table 1.
A	02: 21.23	Advance to the first stored temperature.
A	01: 0102	Advance to the next output array. Same Output Array ID.
A	02: 21.42	Advance to 2nd stored temp, 21.42 deg. C.

There are no date and time tags on the data. They must be put there with Output Instruction 77. Instruction 77 is used in the next example.

If a terminal is used to communicate with the CR510, Telecommunications Commands (Section 5) can be used to view entire Output Arrays (in this case the ID and temperature) at the same time.

OV5.2 EDITING AN EXISTING PROGRAM

When editing an existing program in the CR510, entering a new instruction inserts the instruction; entering a new parameter replaces the previous value.

To insert an instruction, enter the program table and advance to the position where the instruction is to be inserted (i.e., P in the data portion of the display) key in the instruction number, and then key A. The new instruction will be inserted at that point in the table, advance through and enter the parameters. The instruction that was at that point and all instructions following it will be pushed down to follow the inserted instruction.

An instruction is deleted by advancing to the instruction number (P in display) and keying #D (Table 4.2-1).

To change the value entered for a parameter, advance to the parameter and key in the correct value then press A. Note that the new value is not entered until A is keyed.

SAMPLE PROGRAM 2

Instruction # (Loc:Entry)	Parameter (Par#:Entry)	Description
∗1		Enter Program Table 1
01:60		60 second (1 minute) execution interval
Key [#] [D] until is displayed	01:P00	Erase previous Program before continuing.
01:P11		Measure reference temperature
	01:1	Store temp in Location 1
	02:5	
	03:3	
	04:1	
	05:1.0	
	06:0.0	
02:P92		If Time instruction
	01:0	0 minutes into the interval
	02:60	60 minute interval
	03:10	Set Output Flag 0

The CR510 is programmed to measure the datalogger internal temperature every sixty seconds. The If Time instruction sets the Output Flag high at the beginning of every hour. Next, the Output Instructions for time and average are added.

Instruction # (Loc:Entry)	Parameter (Par#:Entry)	Description
03:P77		Output Time instruction
	01:110	Store Julian day, hour, and minute
04:P71		Average instruction
	01:1	one repetition
	02:1	Location 1 - source of temps. to be averaged
05:P92		If Time instruction
	01:0	0 minutes into the interval
	02:1440	1440 minute interval (24 hrs.)
	03:10	Set Output Flag 0
06: P77		Output Time instruction
	01:100	Store Julian day

Instruction # (Loc.:Entry)	Parameter (Par.#:Entry)	Description
07: P73		Maximize instruction
	01:1	One repetition
	02:10	Output time of daily maximum in hours and minutes
	03:2	Data source is Input Storage Location 1.
08: P74		Minimize instruction
	01:1	One repetition
	02:10	Output the time of the daily minimum in hours and minutes
	03:1	Data source is Input Storage Location 1.

The program to make the measurements and to send the desired data to Final Storage has been entered. At this point, Instruction 96 is entered to enable data transfer from Final Storage to Storage Module.

09:P96		Activate Serial Data Output.
	1:71	Output Final Storage data to Storage Module.

OV5.3 SETTING THE DATALOGGER TIME

The next example shows how to set the datalogger date and time using the CR10KD. Here the example reverts back to the key-by-key format.

Key	Display	Explanation
[*] [5]	00:21:32	Enter ∗5 Mode. Clock running but perhaps not set correctly.
[A]	05:0000	Advance to location for year.
[1] [9] [9] [8]	05:1998	Key in year (1998).
[A]	05:0000	Enter and advance to location for Julian day.
[1] [9] [7]	05:197	Key in Julian day.
[A]	05:0021	Enter and advance to location for hours and minutes (24 hr. time).
[1] [3] [2] [4]	05:1324	Key in hrs.:min. (1:24 PM in this example).
[A]	:13:24:01	Clock set and running.
[*] [0]	LOG 1	Exit ∗5, compile Table 1, commence logging data.

OV6. DATA RETRIEVAL OPTIONS

There are several options for data storage and retrieval. These options are covered in detail in Sections 2, 4, and 5. Figure OV6.1-1 summarizes the various possible methods.

Regardless of the method used, there are three general approaches to retrieving data from a datalogger.

1) On-line output of Final Storage data to a peripheral storage device. On a regular schedule, that storage device is either "milked" of its data or is brought back to the office/lab where the data is transferred to the computer. In the latter case, a "fresh" storage device is usually left in the field when the full one is taken so that data collection can continue uninterrupted.

2) Bring a storage device to the datalogger and milk all the data that has accumulated in Final Storage since the last visit.

3) Retrieve the data over some form of telecommunications link, whether it be RF, telephone, short haul modem, or satellite. This can be performed under program control or by regularly scheduled polling of the dataloggers. Campbell Scientific's Datalogger Support Software automates this process.

Regardless of which method is used, the retrieval of data from the datalogger does NOT erase those data from Final Storage. The data remain in the ring memory until:

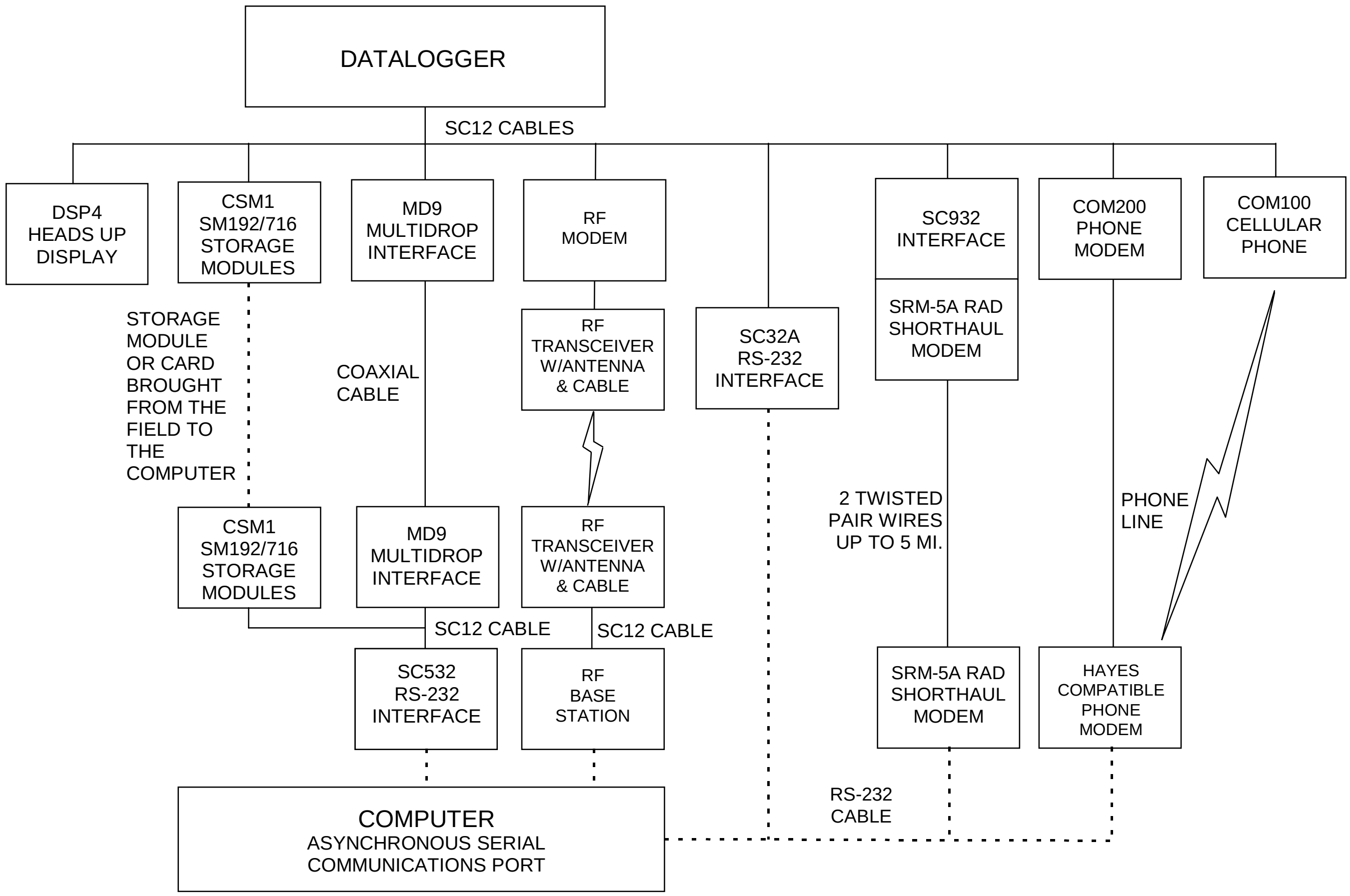
They are written over by new data (Section 2.1)

Memory is reallocated or the CR510 is reset (Section 1.5)

Table OV6.1-1 lists the instructions used with the various methods of data retrieval.

TABLE OV6.1-1. Data Retrieval Methods and Related Instructions

Method	Instruction/Mode	Section in Manual
Storage Module	Instruction 96	4.1, 12
	∗8	4.2
	∗9	4.5
Telecommunications	Telecommunications Commands	5
	Instruction 97	12
Printer or other Serial device	Instruction 96	4.1, 12
	∗8	4.2



NOTES:

1. ADDITIONAL METHODS OF DATA RETRIEVAL ARE:
   A. SATELLITE TRANSMISSION
   B. DIRECT DUMP TO PRINTER
   C. VOICE PHONE MODEM TO VOICE PHONE OR PC WITH HAYES COMPATIBLE PHONE MODEM

2. THE DSP4 HEADS UP DISPLAY ALLOWS THE USER TO VIEW DATA IN INPUT STORAGE. ALSO BUFFERS FINAL STORAGE DATA AND WRITES IT TO PRINTER OR STORAGE MODULE.

3. ALL CAMPBELL SCIENTIFIC RS-232 INTERFACES HAVE A FEMALE 25 PIN RS-232 CONNECTOR.

FIGURE OV6.1-1. Data Retrieval Hardware Options

OV7. SPECIFICATIONS

Electrical specifications are valid over a -25° to +50°C range unless otherwise specified; non-condensing environment required. To maintain electrical specifications, yearly calibrations are recommended.

PROGRAM EXECUTION RATE

System tasks initiated in sync with real-time up to 64 Hz. One measurement with data transfer is possible at this rate without interruption.

ANALOG INPUTS

NUMBER OF CHANNELS: 2 differential or 4 single-ended, individually configured.

RANGE AND RESOLUTION:

Full Scale Input Range (mV)	*Resolution (µV)*	
	Differential	*Single-Ended*
±2500	333	666
±250	33.3	66.6
±25	3.33	6.66
±7.5	1.00	2.00
±2.5	0.33	0.66

INPUT SAMPLE RATES: Includes the measurement time and conversion to engineering units. The fast and slow measurements integrate the signal for 0.25 and 2.72 ms, respectively. Differential measurements incorporate two integrations with reversed input polarities to reduce thermal offset and common mode errors.

Fast differential voltage: 4.2 ms
Slow differential voltage: 9.2 ms
Differential with 60 Hz rejection: 25.9 ms

ACCURACY: ±0.1% of FSR (-25° to 50°C);
±0.05% of FSR (0° to 40°C);
e.g., ±0.1% FSR = ±5.0 mV for ±2500 mV range

INPUT NOISE VOLTAGE (for ±2.5 mV range):
Fast differential: 0.82 µV rms
Slow differential: 0.25 µV rms
Differential with 60 Hz rejection: 0.18 µV rms

COMMON MODE RANGE: ±2.5 V

DC COMMON MODE REJECTION: > 140 dB

NORMAL MODE REJECTION: 70 dB (60 Hz with slow differential measurement)

INPUT CURRENT: ±9 nA maximum

INPUT RESISTANCE: 20 Gohms typical

ANALOG OUTPUTS

DESCRIPTION: 2 switched excitations, active only during measurement, one at a time.

RANGE: ±2.5 V

RESOLUTION: 0.67 mV

ACCURACY: ±2.5 mV (0° to 40°C);
±5 mV (-25° to 50°C)

CURRENT SOURCING: 25 mA

CURRENT SINKING: 25 mA

FREQUENCY SWEEP FUNCTION: The switched outputs provide a programmable swept frequency, 0 to 2.5 V square wave for exciting vibrating wire transducers.

RESISTANCE MEASUREMENTS

MEASUREMENT TYPES: The CR510 provides ratiometric bridge measurements of 4- and 6-wire full bridge, and 2-, 3-, and 4-wire half bridges. Precise dual polarity excitation using any of the switched outputs eliminates dc errors. Conductivity measurements use a dual polarity 0.75 ms excitation to minimize polarization errors.

ACCURACY: ±0.02% of FSR plus bridge errors.

PERIOD AVERAGING MEASUREMENTS

DEFINITION: The average period for a single cycle is determined by measuring the duration of a specified number of cycles. Any of the 4 single-ended analog input channels can be used. Signal attenuation and ac coupling is typically required.

INPUT FREQUENCY RANGE:

Signal peak-to-peak[1] *Min.*	*Max.*	*Min. Pulse w.*	*Max Freq.*[2]
500 mV	5.0 V	2.5 µs	200 kHz
10 mV	2.0 V	10 µs	50 kHz
5 mV	2.0 V	62 µs	8 kHz
2 mV	2.0 V	100 µs	5 kHz

RESOLUTION: 35 ns divided by the number of cycles measured

ACCURACY: ±0.03% of reading

TIME REQUIRED FOR MEASUREMENT: Signal period multiplied by the number of cycles measured plus 1.5 cycles + 2 ms.

PULSE COUNTERS

NUMBER OF CHANNELS: 2 eight-bit or 1 sixteen-bit; software selectable as switch closure, high frequency pulse, or low-level ac modes. An additional channel (C2/P3) can be software configured to read switch closures at rates up to 40 Hz.

MAXIMUM COUNT RATE: 16 kHz, eight-bit counter; 400 kHz, sixteen-bit counter. Channels are scanned at 8 or 64 Hz (software selectable).

SWITCH CLOSURE MODE:
Minimum Switch Closed Time: 5 ms
Minimum Switch Open Time: 6 ms
Maximum Bounce Time: 1 ms open without being counted

HIGH FREQUENCY PULSE MODE:
Minimum Pulse Width: 1.2 µs
Maximum Input Frequency: 400 kHz
Maximum Input Voltage: ±20 V
Voltage Thresholds: Count upon transition from below 1.5 V to above 3.5 V at low frequencies. Larger input transitions are required at high frequencies because of input filter with 1.2 µs time constant. Signals up to 400 kHz will be counted if centered around +2.5 V with deviations ‡ – 2.5 V for ‡ 1.2 µs.

LOW LEVEL AC MODE:
(Typical of magnetic pulse flow transducers or other low voltage, sine wave outputs.)

Input Hysteresis: 14 mV
Maximum ac Input Voltage: ±20 V
Minimum ac Input Voltage:

*(Sine wave mV rms)**	*Range (Hz)*
20	1 to 1000
200	0.5 to 10,000
1000	0.3 to 16,000

*16-bit config. or 64 Hz scan req'd for freq. > 2048 Hz

DIGITAL I/O PORTS

DESCRIPTION: Port C1 is software selectable as a binary input, control output, or as an SDI-12 port. Port C2/P3 is input only and can be software configured as an SDI-12 port, a binary input, or as a switch closure counter (40 Hz max).

OUTPUT VOLTAGES (no load): high 5.0 V ±0.1 V; low < 0.1 V

OUTPUT RESISTANCE: 500 ohms

INPUT STATE: high 3.0 to 5.5 V; low -0.5 to 0.8 V

INPUT RESISTANCE: 100 kohms

SDI-12 INTERFACE STANDARD

DESCRIPTION: Digital I/O Ports C1-C2 support SDI-12 asynchronous communication; up to ten SDI-12 sensors can be connected to each port. Meets SDI-12 standard Version 1.2 for datalogger and sensor modes.

EMI and ESD PROTECTION

The CR510 is encased in metal and incorporates EMI filtering on all inputs and outputs. Gas discharge tubes provide robust ESD protection on all terminal block inputs and outputs. The following European CE standards apply.

EMC tested and conforms to BS EN61326:1998.

Details of performance criteria applied are available upon request.

CPU AND INTERFACE

PROCESSOR: Hitachi 6303.

PROGRAM STORAGE: Up to 16 kbytes for active program; additional 16 kbytes for alternate programs. Operating system stored in 128 kbytes Flash memory.

DATA STORAGE: 128 kbytes SRAM standard (approximately 62,000 values). Additional 2 Mbytes Flash available as an option.

OPTIONAL KEYBOARD DISPLAY: 8 digit LCD (0.5" digits).

PERIPHERAL INTERFACE: 9 pin D-type connector for keyboard display, storage module, modem, printer, card storage module, and RS-232 adapter.

BAUD RATES: Selectable at 300, 1200, and 9600, 76,800 for certain synchronous devices. ASCII communication protocol is one start bit, one stop bit, eight data bits (no parity).

CLOCK ACCURACY: ±1 minute per month

SYSTEM POWER REQUIREMENTS

VOLTAGE: 9.6 to 16 Vdc

TYPICAL CURRENT DRAIN: 1.3 mA quiescent, 13 mA during processing, and 46 mA during analog measurement.

BATTERIES: Any 12 V battery can be connected as a primary power source. Several power supply options are available from Campbell Scientific. The model CR2430 lithium battery for clock and SRAM backup has a capacity of 270 mAhr.

PHYSICAL SPECIFICATIONS

SIZE: 8.4" x 1.5" x 3.9" (21.3 cm x 3.8 cm x 9.9 cm). Additional clearance required for serial cable and sensor leads.

WEIGHT: 15 oz. (425 g)

WARRANTY

Three years against defects in materials and workmanship.

We recommend that you confirm system configuration and critical specifications with Campbell Scientific before purchase.

CAMPBELL SCIENTIFIC, INC.
815 W. 1800 N. • Logan, Utah 84321-1784 • (435) 753-2342 • FAX (435) 750-9540
Offices also located in: Australia • Brazil • Canada • England • France • South Africa • Spain