1902 mk IV
Command set

9 Oct 2009

First edition Oct 2009 PDR

Published by:

Cambridge Electronic Design Limited
Science Park
Milton Road
Cambridge
CB4 0FE
UK

Telephone: Cambridge (01223) 420186
Fax: Cambridge (01223) 420488
Web: www.ced.co.uk
Email: info@ced.co.uk

Table of contents

Introduction

The 1902 command set has evolved through a number of versions as the hardware and our understanding of its capabilities has evolved. Version 1 is the original command set for the mk III hardware, including extensions added during the lifetime of the mk III. Version 2 is used in the original release of software for the mk IV hardware and version 3 (also for the mk IV) has been extended to include support for digital filtering, floating-point parameters and feature support. Unless the version where a command was added is specified, commands are in the version 1 command set and available with all types of 1902.

The 1902 serial line is fixed at 9600 baud, 8 data bits, no parity and 1 stop bit. It will accept commands in upper or lower case ASCII terminated by either a semicolon or a return character. Data sent from the 1902 to the host (except the ADC data) consists of one or more ASCII strings, each terminated with a return character.

Set channel number CHn (?CH;n)

This command is used to select a channel for set-up, from 0 to 31. Only the 1902 with the correct channel will respond to commands after this command is sent; the host software should use timeouts to detect if no 1902 is set to that channel. The 1902 channel number is set using the rear-panel rotary switch and Opt0 of the internal switch pack, with Opt1 reserved for future channel expansion. If channel –1 is selected, commands will be accepted by all 1902 units, but only the unit on channel 0 will reply to the host computer.

Initialise IN

This command initialises the 1902 hardware to a known state: the same state as after power-up. The 1902 will behave as nearly as possible to a piece of wire (unity gain, no filters, DC coupling, no offset). The signal source is set to the single-ended transducer input, and the trigger source is set to trigger1. Character echo is turned off and the error and overrange information is cleared.

Hard Reset HR

This command causes the 1902 to fully restart, which includes reading the contents of the EEPROM. In ARM-based 1902s, any changes to the EEPROM data are written to flash before the restart. If the EEPROM update fails, the 1902 will not restart and the command fails with error code `'V'`.

Program flash PGn (?PG)

The `PGn` command (added in version 2) is used to command the 1902 to enter a flash memory programming mode. As this can cause damage to the flash memory, only one value of n (not documented here) is acceptable. If the 1902 is the currently selected unit, it will read a flash image from the serial line, test the image checksum, and if it is correct, the image is loaded into flash memory and the 1902 restarts. If the 1902 is not the current unit, the serial line input is monitored, but the command is ignored. The flash programming mechanism mimics the behavior of the boot loader LdFlash. The mimicry is sufficiently close that the BootLoad program can use this command instead of loading LdFlash.

The ?`PG` command returns the image number that is currently running. The return value is 0 if the 1902 is running from the factory set slot and 1 if running from the alternate slot.

Read input selection ?IS

This command returns information on the inputs available. The first line returned holds the number of possible inputs, from 1 to 20. This is followed by one line for each input, containing up to 16 characters describing the input.

Input control IPn (?IP)

This command is used to select the input used from the selection provided by the ?IS command. A value of '1' selects the first item in the list, and so forth. ?IP returns the index of the currently selected input.

Selections 1 to 4 will normally be the standard 1902 instrumentation amplifier inputs (ground, normal differential, reverse differential and single-ended), while selections 5 and upwards will vary with the front-end module fitted. Information on the front-end module is returned by the IF command (below).

If the 1902 front-end board is fitted with the input clamping option, in addition to the four non-isolated input selections and the two isolated selections (Grounded EEG and Unclamped EEG), the `?IP` command will return 13 new isolated input settings. They are clamp times in milliseconds. The standard set of clamp times is

0.5ms, 1.0ms, 1.5ms, 2.0ms, 3ms, 4ms, 5ms, 6ms, 7ms, 8ms, 10ms, 12ms, 14ms

but a different multiple of these time values may have been fitted at manufacture according to customer requirements.

As far as the user is concerned, the clamped selections are normal EEG inputs. However, the clamp circuit will be armed, and on a trigger, will clamp for the selected length of time.

Read front end data ?IF

This command returns information on the front-end unit fitted as a string of from 2 to 18 characters in length. The first character holds the front-end identification code, between 0 and 7, which is read from the front-end hardware. The second character is a '0' if the front-end cannot be offset, and a '1' if it can. This character is modified by adding 2 if the front end AC/DC control via the `AC` command is available. The rest of the string, up to 16 characters in length, describes the front-end module fitted. If no front-end card is fitted in the 1902 the front end description will be "`No front end`" and the first two characters will be undefined.

Read gain selection ?GS

This command returns information on the gains available with the currently selected input. The gain information is returned as a series of numbers, one per line. The first line specifies the number of gain values available, from 1 to 20. The n following lines hold the gain values 1 to n. Each gain value is a positive floating point number, indicating attenuation when less than unity.

Set gain GNn (?GN;n)

This command is used to set the gain of the 1902 from the selection available. The data value can run from 1 to the maximum possible. The selection used is determined by the input currently selected.

Offset controls

The offsetting system of the 1902 is slightly different from the simple 1-of-n selections used elsewhere. The offset control is provided using a signed 16-bit control value and the information commands give details of the ranges available and the offsets that these ranges allow.

Read offset ranges ?OS

This command returns information on the offset ranges available like the gain range information supplied by `?GS`. The first number returned is the number of offset ranges available, from 0 to 8. Following lines give the full-scale offset available from that range in volts; specifically the signed offset value that would be obtained using an offset control value of 32768 (Note that the actual offset values available run from –32768 to 32767.) Offsets unavailable are indicated by zero ranges. It is assumed that an offset control value of zero will always give zero offset. The availability of offsets, as well as the ranges available, may change with the input selection.

Select offset range ORn (?OR;n)

This command sets the offset range from the ranges provided by the `?OS` command. A value of '1' selects the first setting in the list, as usual. `?OR` returns the number of the currently selected offset range.

Set offset value OFn (?OF;n)

This command is used to write an offset value to the DAC. The command data is a sixteen bit signed (2's complement) number.

Read low-pass filter data ?LF and ?LS

These two commands return information on the low-pass filter module and the cut-off frequencies available. The `?LF` command returns a string, of up to 16 characters, describing the filter. The `?LS` command returns information on the filter settings available in the same manner as the `?GS` command; the first line holds the number of filter settings, while the following lines (up to 20) return the cut-off frequencies in Hz for each setting as floating point numbers. If zero filter ranges are specified, the filter is not fitted.

Low-pass filter LPn (?LP;n)

This command sets the 1902 low-pass filter settings from the range provided by the `?LS` command. A value of '1' selects the first setting in the list, as usual. In addition, a value of '0' will cause the filter to be disabled.

Read high-pass filter data ?HF and ?HS

These two commands return information on the high-pass filter module and the cut-off frequencies available. The `?HF` command returns a string, of up to 16 characters, describing the filter. The `?HS` command returns information on the filter settings available in the same manner as the `?GS` command; the first line holds the number of filter settings, while the following lines (up to 20) return the cut-off frequencies in Hz for each setting as floating point numbers. If zero filter ranges are specified, the filter is not fitted.

High-pass filter HPn (?HP;n)

This command sets the 1902 high-pass filter settings from the range provided by the `?HS` command. A value of '1' selects the first setting in the list, as usual. In addition, a value of '0' will cause the filter to be disabled.

Notch filter enable NFn (?NF;n)

This command is used to enable and disable the mains frequency notch filter. A parameter value of 0 disables the filter, while non-zero value enables it. The default state has the notch filter disabled. The notch filter set-up can be determined by using the `NT` command (below).

Read notch filter type ?NT

This command returns information on the notch filter 50 Hz/60 Hz build option. It returns an integer code; either '50' or '60' as appropriate or 0 if the notch filter is not fitted.

AC/DC coupling ACn (?AC;n)

This command is used to select AC or DC coupling of the input signal. A data value of '0' sets DC coupling, while '1' sets AC coupling.

Low and high pass digital filters

With digital filtering, the original arrangement of filters are extended to provide multiple types of filter selectable by the user. Each filter type will have a name and will be controlled by either a 1 of n selector or a continuously variable floating point value, though currently all use a floating-point variable. The two sets of commands for high pass and low pass digital filters are effectively identical, most documentation will refer only to the low-pass variant.

Digital filters available ?DF;n

The `?DF` command (added in version 3) can be used to query for the availability of digital filters. If the 1902 hardware and monitor ROM versions returned by `?RV` are high enough (at least 2 and 22 respectively), then this command can be used and if it returns non-zero then you can make use of the extended digital filter commands and the other digital filter-based commands only available in version 3 of the 1902 command set.

The value returned by the ?DF command does more than indicate if digital filters are available; it also indicates if a standard filter arrangement is in use so that software that uses the 1902 can avoid the time delays caused by querying all the filters available. A value returned of 1 (currently the only supported value) means:

> 4 filters available for each of low and high pass: 2 pole Butterworth and Bessel and 3-pole Butterworth and Bessel (in that order). All filters are continuously variable, low-pass have an allowed range from 1 to 10000 Hz, high-pass from 0.01 to 1000 Hz. For each filter a set of 13 standard settings are supplied – see the 1902 support code if you need to know what they are.

Other known values of the ?DF response may be added to the 1902 repertoire as the system is developed, a value of -1, while indicating that digital filters are available, will always require that the filters available be queried using the LD and HD commands.

Digital filter set selection
LDn (?LDn)
HDn (?HDn)

These commands (added in version 3) are used to find out about the digital filters available and to select a filter to use. These commands differ from almost all other 1902 commands in that the query form of the command includes an integer argument that is used to select the data to be queried.

?LD0 will return the number of filter sets available.

?LDn returns information on filter n (from 1 up). The first line returned starts with a single character holding the flags followed by the filter name (up to 16 characters). The flags value can be obtained by subtracting '0' from the flags character. Currently the only flag defined is bit zero, which is set for 1 of n filter selectors, clear for continuously variable. For continuously variable filters, the next two lines specify the low and high limits to the allowed range. There then follows, for all filter types, an item count followed by the specified number of items, one per line, note that unlike ?LS this response will include a first item of zero for filter off. This list of items is the available selections for 1 of N types and a default set of options for continuously variable filters.

LDn, where n is 1 to the value returned by ?LD0, will select low pass filter set n. ?LD-1 will return the number (1 to n) of the currently selected filter set. Note that selecting a filter set will not change the filtering until the LO (HO) command is used.

Digital filter control
LOv (?LO;v)
HOv (?HO;v)

These commands (added in version 3) are used to set and get the filter cut-off frequency. Note that, unlike most 1902 commands, these commands use a floating-point parameter.

?LO will return the current setting for the currently selected filter. LOv sets the current filter cutoff (for continuously variable filters) or sets the current selection (for 1 of N types), note v can be floating point or integer as required.

Digital rectification
DRn (?DR;n)

This command (added in version 3) enables and disables digital rectification of the data after the high-pass filter and before the low-pass filter. The DR0 command turns off rectification while DR1 turns it on. ?DR reports the rectify state. The IN command turns rectification off.

Digital gain
DGv (?DG;v)

This command (added in version 3) adds additional digital gain to the output. You might use this after rectifying a signal or to increase the output resolution when using a low-pass filter, or to invert a signal (specify a negative value for the gain). The gain is supplied as a floating point number in the range –10000.0 to 10000.0. Values outside this

range or between –0.0001 and 0.0001 cause a `'V'` error and are ignored. The `IN` command resets the digital gain to 1.0.

Digital offset DOn (?DO;n)

This command (added in version 3) adds a digital offset to the output. The offset is supplied as a floating point number in the range -1 to 1, where 1 corresponds to the full-scale positive excursion of the DAC. Values outside this range cause a `'V'` error and are ignored. The `IN` command resets the digital offset to 0.0.

Facilities management

In addition to adding the digital filters, the version 3 command set also includes new commands aimed at managing as yet un-designed facilities, since clearly we will find ourselves adding new digital capabilities to the design. The facilities model is as follows:

A set of up to 8 optional facilities will exist, each 1902 can provide up to 4 (maybe more) of these. Each facility will have a name, a unique ID (from 0 to 7) and be controlled by either a 1 of n selector or a continuously variable floating point value. The 1902 control panel will be able to enquire about the facilities available from each 1902, find out about each one and generate dialog controls to allow the user to manipulate the settings. The unique ID will be used to keep current settings for a facility (as stored on disk) separate from settings for other facilities. The digital EMG filter option uses this mechanism and has facility ID 0.

Read facility data FAn (?FAn)

This command (added in version 3) is used to find out about the facilities available and to select a facility to use. The command differs from almost all other 1902 commands in that the query form of the command includes an integer argument that is used to select the data to be queried.

`?FA0` will return the number of facilities available

`?FAn` will return information for facility n (from 1 to the result from `?FA0`). The results will be on multiple lines – first line is character for flags, then a character for unique facility ID plus name of facility (up to 16 characters). The flag value is the character value minus the zero character, bit zero is set for 1 of n selectors, clear for continuously variable. The unique facility ID is a number generated by subtracting the character value zero from the character, giving a value from zero upwards. It is intended as an index into the stored values in conditioner settings. The second line is the number of settings or the low limit, following lines are the individual settings or just one line with the upper value. Note that the individual settings are strings of up to 16 characters.

`FAn` will select facility n (from 1 to n) to be controlled by following FS commands. `?FA-1` will return the currently selected facility number, from 1 to the number returned by `?FA0`.

Facility setting FSv (?FS;v)

The `FS` command (added in version 3) gets and sets the value for the currently selected facility. `?FS` will return the value for the currently selected facility, while `FSv` sets the value.

Miscellaneous commands

These commands are used for various purposes including the original test and setup of the 1902.

Direct DAC access FDn, RDn and SDn

The FDn command can be used to turn digital filtering on and off – it disables and enables the ADC hardware generating the data; a parameter value of 0 turns activity off while 1 turns it on. With digital filtering off, you can use the RDn command to write a value to the DAC directly – n is a signed 16-bit value from –32768 to 32767. The SDn command also writes 16-bit data to the DAC, but adjusts it beforehand with the built-in offset and gain compensation values.

Compensation values GCv, OCn, HCn and SC

The GC command is used to read and set the digital filter gain compensation value. Similarly, the OC command reads and sets the digital offset compensation value used when the high pass filter is not in use and HC (added in version 2.4) the separate offset compensation used when there is an HP filter. All of these values are held in EEPROM and loaded into memory when the 1902 starts up, the GC, OC and HC commands only modify this working value in memory. If the SC command is used, any changes to the working values are made permanent by writing them back to the EEPROM.

Select trigger TGn (?TG;n)

This command is used to select which trigger input drives the trigger output. A parameter value of 1 selects trigger 1, while 2 selects trigger 2. The default state is trigger 1. Note that the clamp circuit (if fitted) always uses trigger 2 and there is no need to select this explicitly.

Trigger polarity TPn (?TP;n)

This command (added in version 3) is used to select which direction of edge on the selected trigger input causes the trigger output. A parameter value of 1 selects a rising edge, while 0 selects a falling edge. The default state is a rising edge. Note that, in early versions of the 1902 mk IV only a falling edge can be used, though this can be changed by a factory hardware upgrade.

Set control byte X0n and X1n (?X0;n and ?X1;n)

These two commands are used to write to the two control bytes of the front-end module. The function of these bytes is entirely determined by the front-end module fitted.

Set debug address ADn (?AD;n)

This command sets the address to be used for the poke and peek commands (below). It is a sixteen-bit number specifying an absolute (6501) address. Though the command is present in the ARM 1902 designs (identifiable as being hardware version 2 or higher returned from the ?RV command), it is of no use.

Peek\Poke PKn (?PK;n)

This command writes a byte value to the debug address. The query form of the command reads from the same location. These commands must be used with great care and should not be used at all with the ARM version of the monitor.

Character echo ECn (?EC;n)

This command allows you to control whether characters sent to the 1902 are automatically echoed to the host. A data value of '0' sets no echo, while a value of '1' enables the echo; the default state is no echo. Add 2 to the data value to force LF as well as the normal CR.

Multiplexer control MXn (?MX;n)

This command controls the multiplexer that routes signals to the calibration Subclic connector on the 1902 PCB. The values for n are:

Value	Signal name	Comment

0	ANA	Analogue ground
1	NOTCH	Notch amplifier output
2	DIFF_OFFS	Instrumentation amplifier ref. input
3	COND_OP	1× / 3× amplifier output
4	ACDC_IP	After 1 µF capacitor & CMOS switch
5	DECADE	Decade gain amplifier output
6	ADC_ OUT	Main output from filter card
7	DAC_OFFS	Transducer socket pin 6

The program Try1902 saves the user from needing detailed knowledge of the table above.

Read revision level ?RV;1902xyz

This command returns a seven-character identifying string to the host. The first four characters are '1902' to allow identification of the hardware. The next two characters are the 1902 EEPROM software version numbers (i.e. version *x*.*y*) while the last character is the hardware revision level. The software for the 1902 mk III uses software versions 1.*z* and reports hardware version 1, the ARM version has software versions 2.*z* and hardware version 2.

Read CPLD version ?CV;n

This command (added after version 3) returns the 1902 CPLD version number as read from the revision ID register on the CPLD. It is only available with 1902 mk IVs, the command was added in version 2.3 of the 1902 monitor.

Read overrange flag ?OV;n

This command returns a flag indicating if an overrange signal has been detected. A '1' is returned if an overrange has been detected, otherwise the command returns '0'. The flag is cleared by using this command, or by initialising the 1902.

Read error ?ER;xxx

This command returns three characters giving details of the most recent error detected. The first two characters returned are the command identification characters for the command that caused the error, the third indicates the form of the error. Using this command will clear all error information to the 'no error' state, in this state the command will return three zero characters. The currently defined error form characters are:

U	Unknown command
L	Bad command length or form
I	Illegal or invalid parameter
V	Unacceptable parameter value
O	Serial line buffer overflow
F	Serial line framing error

In the case of the two serial line errors, the command identification characters will be 'RS'.

Reading sampled data

The ADC option in the 6502-based 1902 allowed limited sampling of analogue data through the serial line. The latest ARM-based code emulates this with some extensions and changes to the original command set. The commands are:

Sample rate ATn (?AT)

This sets the desired rate for the data for the AR command, in Hz. The rate is achieved by returning every n^{th} sample from the internal ADC, which runs at 30 kHz, so the rate you get will not, in general, be the same as the rate you request. The standard rate of 100 Hz is always achievable by using `AT100`. The `?AT` command returns the actual rate achieved as a floating point number.

You can set rates from 0.001 to 480 Hz, however, the maximum sustained serial line rate is approximately 960 characters a second, so the maximum sample rate available depends on the output data format. 480 Hz sampling can only be achieved with binary format. The maximum rate for hexadecimal data is 192 Hz with CR only and 160 Hz with CRLF. The worst-case rate for decimal numbers with a CR terminator is 137 Hz, and with CRLF it is 120 Hz.

You should use the LO command to set a low pass filter that is significantly less than ½ of your sampling rate to avoid aliasing of the output. For example, for 100 Hz samples, you might want to use the LO`30` command.

Difference from the 6502 version

This command was implemented differently in the 6502 version of the system; in that implementation, the command was `ATn`, where n set the output rate to 100/n Hz. Use `AT` with care if compatibility with early 1902s is desired.

Output format AFn (?AF;n)

This command sets (and reads back) the output format. The allowed values of n are: 0=decimal, 1=hexadecimal, 2=binary, 3=hexadecimal with no end of line. The table summarises the formats. Items in round brackets are present if required by the number. {`LF`} is present if set by the `EC` command.

	Name	Format	Example	Max rate	with LF
0	Decimal	(-)(1-4 digits) digit CR {`LF`}	-31297 CR	137	120
1	Hexadecimal	4 hex digits CR {`LF`}	85BF CR	192	160
2	Binary	MS byte, LSByte	133 191	480	-
3	Hexadecimal	4 hex digits	85BF	240	-

The returned numbers are 16-bit 2's complement values as written to the output DAC in the 1902. The maximum value returned is 32767, the minimum is –32768, an output of

zero corresponds to zero volts. Decimal format uses the minimum number of characters possible for each number, but this can be five digits plus a sign in the worst case. Hexadecimal output always uses 4 digits, with 0-9,A-F representing 0 to 15. Binary format packs the 16-bit number into two consecutive serial line characters with the first character holding bits 15-8 and the second holding bits 7-0.

Difference from the 6502 version

The 6502 version of the 1902 command set does not support format 3 (hexadecimal with no end of line characters).

Single sample
AS

Read a single number in the current format. The most recent sampled value is returned. If the rate set by the AT command is `r` Hz, this can be up to 1/`r` seconds in the past.

Repeated samples
ARn

This command returns data values starting as soon as the command is received at the rate set by the `AT` command and in the format set by the `AF` command. If `n` is present, it sets the maximum number of values to return. If `n` is absent or 0, there is no limit on the number of values. Negative values of `n` are reserved, and cause a `'V'` error. Data is returned until the count set by a positive `n` is exhausted, or until a serial character is received by the 1902. The 1902 will always send complete data values, it is possible for the 1902 to return no values at all if a serial line character is received before the first data value is ready.

If you have set an over-optimistic rate in the `AT` command for the data format set by the `AF` and `EC` commands, the output will have missing data points. There is no warning for this condition; it is a good idea to set your maximum rates to be a few Hz less than the maximum rates given in the table in the `AF` command description.

Difference from the 6502 version

The 6502 version of the 1902 does not support the use of `n` to limit the number of returned values.