



USER GUIDE

Acknowledgements

We would like to thank all those who have contributed to the development of this product, and in particular our customers, who have made it possible for us to produce it.

Manual authors: Martin Walker & Gwyn Lewis.

Text editing: Martin Peapell & Gwyn Lewis.

Centurion programmed by: Martin Walker.

(C) 1987 Chalice Software & Swindon ITEC Ltd.

Published by:
Swindon ITEC Ltd
6 Oppenheimer Centre
Greenbridge Road
Swindon
SN3 3JD
England.
tel. : (0793) 611808

CENTURION MANUAL

CONTENTS

SUBJECT	PAGE
Contents	1
Introduction	2
Fitting instructions	3
Command operation	4
Command syntax	6
Centurion internal variables	8
CONFIGURATION AND STATUS commands	9
FILE commands	16
MEMORY commands	23
OPER commands	31
ADFS commands	38
DUTILS commands	42
BASIC commands	48
Command summary	61
Copyright and conditions	66

CHALICE CENTURION INTRODUCTION

The Chalice Centurion ROM is designed for the BBC Master & Compact computer series. It provides approximately 100 commands, of use to every kind of computer user.

The commands are divided into the following categories:
BASIC, OPER, MEMORY, DUTILS, FILE, ADFS.

Almost all of the commands are fully TUBE compatible, the current exceptions being some of the BASIC commands.

The commands are designed to be simple in operation so they can be used with little reference to the instructions in this manual. Many commands are designed to allow the entry of lists of parameters, e.g

*BASE 12 45 66.

would display the numbers 12 45 & 66 in Hex, Decimal & Binary. The Centurion has the unique feature on a toolkit ROM of having its own set of accessible internal variables which allow interaction with the BASIC language, and give added power and convenience in other respects without the ROM interfering with the operation of BASIC by directly affecting the resident integer variables.

The Centurion is one of very few ROMs designed specifically for the Master series, and as such it has been programmed around all the new features available. For example the ROM claims a page of workspace, something which was impossible to do (without upsetting almost all customers) on the BBC B. It is also programmed using the more powerful and compact CMOS 6502 instruction set.

FITTING INSTRUCTIONS

You can either fit your new Centurion ROM inside your Master computer, or fit it into a Master cartridge according to the instructions supplied with the cartridge.

INSERTING CENTURION INTO THE MASTER

1. Ensure the computer is switched off and unplugged at the mains.
2. Remove the four screws holding the cover. These are all underneath the machine. They are labelled 'fix'.
3. Identify the ROM slot area. This can be found at the right-hand side of the main circuit board, just above the two cartridge slots. It might be easier to remove the plastic base for the cartridge slot, as it simply lifts out. Take care with the battery.

The operating system, BASIC, DFS, View and Viewsheet are all in the big chip, the 'BIGROM' above socket A. ROM numbers 0 to 3 are normally mapped to the two cartridge slots and ROM numbers 4 to 7 are mapped to the three sockets.

Sockets A and C are 32k sockets and socket B is a 16k socket. Normally the four banks of sideways RAM are mapped to sockets A and C.

1. Your ROM must be inserted into socket B. It must have the small notch pointing out towards the left-hand side of the board.
2. Refit the cover.
3. Switch the machine back on. If the Centurion has been installed correctly, then typing *ROMS will display it. If not check that the pins are not bent on the Centurion ROM.

COMMAND OPERATION

Commands are entered by typing the command in either upper or lower case (or a mixture of both) preceded by a "*". It is possible to abbreviate the command by typing a "." after enough letters of the command have been typed to identify the command in preference to those available either through the "MOS" or higher priority ROMs.

To help avoid conflict with higher priority ROMs, the letter "Y" can be used at the start of any Centurion command. To further identify the command, if you enter an *uppercase* "Y", then the command will be searched for as a Centurion command; and if you enter a *lowercase* "y", then the command will be passed through to lower priority ROMs (without the "y").

If a command is entered without its parameters or with too few parameters, then an error message will be generated, and the correct syntax of the command will be displayed.

COMMAND OPERATION

trings may be entered with or without quotation marks. If entered without quotation marks the first space or "," encountered is taken as the end of the string. If entered with quotation marks then all text between the quotes will be accepted. It is also possible to enter quotes in the text by placing two adjacent quotes. e.g "A""B" will be interpreted as A"B. If the string is the last item on the command line then the usual "Missing "" error generated in the case of an uneven number of quotes will not occur.

Numbers may be entered in HEX, DECIMAL or BINARY, as one of the resident integer variables A%-Z%, or as one of the Centurion's own internal variables @@-@Z.

By default the commands will accept hex values where a memory address is required, and decimal at all other times.

It is possible to enter a number in a specific base at any time by preceding the number with "&" for hex, "#" for decimal, "%" for binary.

ROMS can be specified by either their socket number (which can be found by using the command *ROMS), or by the ROM title with preceding quote. The ROM title may be abbreviated to as few letters as are needed to exclusively identify it; for example "BA" would specify BASIC.

COMMAND SYNTAX

The Centurion contains six categories of commands:
ADFS, MEMORY, BASIC, FILE, DUTILS, OPER.

Typing *HELP followed by one of these words will display the syntax of each the commands available in the category. The syntax display contains information which will enable you to understand the operation of many commands without reference to the user guide so long as you are familiar with the points set out below. There are many other items which will appear in the syntax, but these are the main ones, and also those which have complicated aspects to them.

<>	Compulsory parameter.
(<>)	Optional parameter.
<address>	Memory address. By default entered in hex.
<>...	List requiring at least one item to be entered.
(<>)...	Options list. If no data items are entered, then a default will be used.
<ROM>	One of the sideways ROM/RAM banks available on the Master series. These are numbered 0 to 15.
(<ROM>)	As above, but the ROM number is optional; if it is not entered then the ROM number defined with either the NROM command or the configuration command ROM will be used.
<track(s)>	Where track or tracks is specified you can use any number between 0 - 159, numbers above 79 being mapped to either drive 2 or 3. This is to fit in with the ADFS standard of treating two sides of a disk as a single surface. You can use this to advantage on the DFS, since a protected double-sided disk could be backed up with a single command.

COMMAND SYNTAX

- (<drive>) The disk drive you wish a command to operate. If you do not specify a drive, then the current default drive of the ADFS or DFS will be used. Even if you are operating under ADFS you can usually select drive 2 or 3 to access the second side of a disk.
- <lit spec> This allows you to specify a file specification which will match a file or a number of files within an ADFS directory. You may use the wildcard characters "*" and "#" in the filename. These wildcards operate in exactly the same way as within the ADFS.

CENTURION INTERNAL VARIABLES

The Centurion internal variables are designed to give many of the commands more versatility, and to make them more user friendly. Some of the internal variables are reserved for special purposes, and some of them may be used to store any numbers that you wish to use regularly. There are 26 variables called @@-@Z.

- @@-@O These locations are used to store the first 16 finds of the **SFIND** and **BFIND** commands. This area is also used to store IDs read with the **IDDUMP** command.
- @P This variable is set on exit from the **MDIS** and **MEDIT** commands to the last address accessed by the command.
- @X Last address entered by the user to the Centurion's main memory pointer. This will for example be the start address of the **MDIS**, **MEDIT** and many other commands.
- @Y The number of occurrences of a string or set of bytes found with the **SFIND** and **BFIND** commands. The number of IDs read in with the **IDDUMP** command.
- @Z The start of memory in the I/O processor which will be used by commands which require a large buffer area. On start-up this will be set to the value of OSHWM (normally &E00). Those commands which make use of this buffer are noted as such in their instructions. Certain commands such as **PBACKUP** which make use of large amount of memory **DO NOT** use this feature since it would limit their operation. It is envisaged that this feature is useful as a means of avoiding corruption of an area of memory you are operating on.

CONFIGURATION AND STATUS COMMANDS

Configuration commands

The configuration commands are designed to set features of the Centurion that will affect the operation of many of the Centurion commands. The configurations you set will remain in operation throughout <BREAK> and <CTRL><BREAK>.

This is different to the standard type of configuration commands available on the Master series, in that the Centurion configurations will be lost if the power is turned off. The Centurion configurations can also be lost in other extreme circumstances, such as the Centurion's workspace being overwritten. In this sort of situation the Centurion will detect that this has happened, and set up the default configurations. You can display a list of the configuration commands available by typing *CONFIGURE.

To enter a configuration command, for instance the **DATE** command, you must enter *CONFIGURE DATE. This can usually be abbreviated to *CON.DATE. Commands can be entered in the standard manner of other commands within the Centurion, e.g. they can be entered in any case, and they can, where possible, be abbreviated.

Status commands

The actual status corresponding to the configuration commands can be displayed by using *STATUS, which will display a complete list of configurations including those associated with the operating system; or by using *STATUS along with one of the configurations which will display the status of that configuration.

CONFIGURATION AND STATUS COMMANDS

CMOS & NOCMOS

These commands control the actions of the disassembly commands when they encounter op-codes which are part of the 65C02 extended instruction set used on the Master series.

If the configuration is set to CMOS (which is the default), then these op-codes will be displayed; and in the case of the SDIS & PDIS commands which have to interpret op-codes so as to decide where labels should be placed, then mos codes will be interpreted.

If NOCMOS is used, then these opcodes will be considered as data bytes.

This feature is useful since many pieces of code produced for the Master series do not use the CMOS instruction set for compatibility with earlier Acorn computers, therefore it is easier to understand code, and labels will be more accurately placed if these codes are correctly interpreted.

minimum abbreviation: Cm. & NoCm.

default: CMOS

associated commands: *MDIS, *SDIS, *PDIS

CONFIGURATION AND STATUS COMMANDS

OM [<D>]

This command selects which of the 16 paged ROM/RAM banks available on the Master series will be used by default if you do not specify a ROM number when entering a command. It will also come into use on those commands such as SFIND which do not accept a ROM number when the command is entered, but do in fact operate on a ROM when addressing memory between &8000 - &BFFF. The NROM command will also set this feature.

minimum abbreviation: ROM

default: The BASIC ROM.

associated commands: NROM

SEIO & USETUBE

This configuration will set memory accessing commands, so that they will either access memory in the Master series computer (I/O processor), or the memory of an attached second TUBE processor.

If you set the USETUBE configuration you can still access memory in the IO processor by specifying a ROM number when entering a command. If there is no second processor attached then the I/O memory will be accessed automatically.

minimum abbreviations: USEIO & USET.

default: USETUBE

CONFIGURATION AND STATUS COMMANDS

MOSRAM & NOMOSRAM

The Master Series computers contain an area of memory not normally accessible to the user which is used to store the function key definitions, the character set definitions, etc. The MosRAM configuration will cause memory accesses in the IO processor to be directed into this RAM.

The NOMOSRAM configuration will allow accesses in this area to look at the selected paged ROM.

minimum abbreviations: MOS.& NOM.

default: NOMOSRAM

FILERAM & NOFILERAM

The Master series has two forms of memory which can be mapped in to the area between the addresses &C000-&DFFF. They are either the operating system ROM, or the RAM allocated to the paged ROMs and used by the filing systems such as the ADFS and DFS. Some of this memory is also used by the operating system. For further information on this subject you are advised to read the the Master series Advanced User Guides.

The RAM used by the filing system etc is made accessible by the configuration command FILERAM, and the operating system ROM is accessed with the command NOFILERAM.

minimum abbreviations: FILER. & NOFI.

default: FILERAM

CONFIGURATION AND STATUS COMMANDS

HADOW & NOSHADOW

The Master series computer has two RAM banks between addresses \$3000-&\$7FFF, either of which can be used to store the screen memory. These commands select whether the normal screen memory (which is also, as a rule, where programs are stored) or the shadow screen RAM will be used when memory between these addresses is accessed.

Minimum abbreviations: SHA. & NOS.

Default: NOSHADOW

DATE & NODATE

The DATE configuration command will cause all files saved out on the ADFS to have a date and time stored in their load and execution addresses in the catalogue.

This command can be used to keep a record of when a file was last saved. The DINFO command can then be used to recall the date and time. Date stamping can be used on all BASIC & VIEW files and files from most other languages. It is however important that you do not try to save files that have relevant load and execution addresses, since these addresses will be overwritten. The reason why most languages will accept this command is that they will always load files to a specific address corresponding to their own workspace, and make no use of the file's execution address. Once selected, the DATE command will remain in effect until the NODATE configuration command is entered.

Minimum abbreviations: DA. & NOD.

Default: NODATE

Associated commands: DATESTAMP, DINFO

CONFIGURATION AND STATUS COMMANDS

AUTOSAVE & NOAUTOSAVE

The AUTOSAVE configuration is designed to make the saving of files easier and less error prone. Once set, loading and saving of files takes place in the normal manner. The advantage comes if you save a file with a null filename, e.g SAVE "", in which case the file will be saved under the name of the last file loaded. This means for example that if you are developing BASIC programs using a number of files you will no longer run the risk of overwriting another file by entering the wrong filename. It also encourages regular saving of files as you develop them.

minimum abbreviations: AU. & NOA.

default: NOAUTOSAVE

RECALL & NORECALL

The RECALL configuration allows the last 63 commands entered in BASIC, VIEW and most other languages to be stored in one of the sideways RAM banks. The commands can then be recalled or saved to disk with the RECALL and FRECALL commands.

minimum abbreviations: REC. & NOR.

default: NORECALL

CONFIGURATION AND STATUS COMMANDS

CENTURION

The CENTURION configuration will set the DATE, RECALL & AUTOSAVE configuration commands. It is provided as a separate command because it is envisaged that many people will want these features set on switch-on. It is however a very bad policy to set features which alter the operation of a computer automatically on switch-on since this can lead to incompatibilities with other software. When used as a STATUS command, all the Centurion configurations will be displayed.

minimum abbreviation: CE.

default: Not applicable

associated commands: DATE, AUTOSAVE, RECALL

FILE COMMANDS

The FILE commands will operate on and are relevant to most filing systems available on the Master series, especially the DFS and the ADFS.

*RSAVE <ROM> (<filename>)

Dumps the contents of one of the sideways ROMs to disk. If you do not specify a filename then a filename will be constructed from the name of the ROM image. Using this command in association with the RLOAD command, you can collect a library of images which can later be loaded and used as required. Some ROMs may be protected against this.

NOTE: Unless you have purchased the original ROM you are saving to disk, then you are breaking copyright law, and may be liable to prosecution.

minimum abbreviation: *RS.

associated commands: *RLOAD

*RLOAD <filename> <ROM>

Installs a ROM image on disk into any bank of sideways RAM. The command will cater for any non standard bank of RAM on the Master as long as it will be written to when its socket is selected. To activate the ROM image you will have to press **<CTRL> <BREAK>**.

NOTE: Unless you have purchased the original ROM you are loading from disk, then you are breaking copyright law and may be liable to prosecution.

minimum abbreviation: *RL.

associated commands: *RSAVE

FILE COMMANDS

KSAVE <filename>

This will save the definitions of the red user-definable function keys from your computer keyboard to disk. These definitions can then be loaded back in to your computer at any later time, using the KLOAD command.

Minimum abbreviation: *KS.

Associated commands: *KLOAD

KLOAD <filename>

This loads a set of function key definitions (previously saved with the KSAVE command) into the computer.

Minimum abbreviation: *KL.

Associated commands: *KSAVE

FILE COMMANDS

***FONTSAVE <filename>**

This command will save the Master series character set, generally speaking as redefined using VDU 23. For details of user-defined character sets look in your Master series user guide.

minimum abbreviation: *FONTS.

associated commands: *FONTLOAD

***FONTLOAD <filename>**

The *FONTLOAD command is used in conjunction with the *FONTSAVE to load and save character set definitions. On the Master series all characters can be redefined without taking up any of the memory used by programs. This gives you the opportunity to have your own personal fonts, even on most commercial programs.

minimum abbreviation: *FONTL.

associated commands: *FONTSAVE

FILE COMMANDS

SDIS <filename> <address> <address> (<ROM>)

The *SDIS command will allow you to generate labelled disassembled code, compatible with the BBC BASIC assembler, which will be saved to disk. The data saved to disk is in a spooled format. To produce a BASIC program, *EXEC this file. For more information on the technical aspects of this command, see the PDIS command.

minimum abbreviation: *SD.

associated commands: *PDIS, *MDIS

NOTE: Uses a buffer starting from @Z. Please refer to notes on the internal variables.

***FRECALL <filename>**

This will save all the commands that have been stored since the RECALL configuration was set. This feature used in conjunction with a word processor (the VIEW word processor can be used) allows you to create command files which can be read with EXEC. In a typical instance a set of GREPLACE commands could be saved, allowing you to perform the same sequence of edits on a set of files.

minimum abbreviation: *FREC.

associated commands: *RECALL

FILE COMMANDS

*SCRAMBLE <filename> <password> (<filename>)

This command is a file security feature. It is useful if you have a file on a disk which you do not want other people with access to the disk to read. The command will encrypt the file using the password, which can contain any keyboard character apart from <TAB>. If you do not specify the optional filename then the encrypted data will be saved under the same filename, which adds to the security. Before using this command on important data we suggest that you familiarise yourself with the operation, since the encryption is so tight that we cannot offer under any circumstances a data recovery service. To recover the data from an encrypted file you must use the *UNSCRAMBLE command with exactly the same password.

minimum abbreviation: *SCR.

associated commands: *UNSCRAMBLE

NOTE: Uses a buffer starting from @Z Please refer to notes on the internal variables.

*UNSCRAMBLE <filename> <password> (<filename>)

The operation of this command is the inverse of the SCRAMBLE command and should be used on files that have been encrypted with the SCRAMBLE command to return them to a readable state. As a point of interest these two commands are reversible, so you may may use *UNSCRAMBLE to encrypt a file and *SCRAMBLE to decrypt a file.

minimum abbreviation: *UNS.

associated commands: *SCRAMBLE

NOTE: Uses a buffer starting from @Z Please refer to notes on the internal variables.

FILE COMMANDS

*ALTER <filename> <load> <exec>

This will directly alter the load and execution address of a file on a disk, without loading the file. This command has the use for example of converting a text file into a command file capable of being exec'd by *<filename> by setting the load and execution addresses to &FFFFF.

minimum abbreviation: *ALT.

*CHECK <filename 1> <filename 2>

This will check the contents of one file against those of another, to see if they are identical. Any differences in the file will cause the location within the file to be displayed, and the contents of this location of both files. If one of the files is of a different length, this will also be displayed.

minimum abbreviation: *CHE.

FILE COMMANDS

***FILTER <filename> <filename> <first> <last>**
<bytelist>

The filter command will copy one file to another using a set of filters to determine selectively which bytes will be copied. Bytes in the file between the ranges of ASCII values first and last will be copied across (as a basic rule) and the list of bytes following first and last will act as exception conditions, those bytes in the file within the specified range but also in the list will be excluded and those bytes outside the range but in the list will be included.

First	Last	Exception list	
32,	126,	13	copy all text codes normally.
0,	255,	32	Strip all spaces from a file.
32,	255,	13, 7, 34	Produce a file of only displayable characters but stripped of any quotation marks.

The command can also produce an identical copy of a file without destroying any program held in memory, by using first=0, last=255.

minimum abbreviation: *FI.

MEMORY COMMANDS

The commands in this section are orientated to the reading and writing of the Master series memory. The actual memory selected will be effected by the configuration commands and the parameters entered. The addresses entered to these commands must be in the range &0000-&FFFF. On most of the commands a rom number can be entered to access one of the paged roms. If the USETUBE configuration is operative and you are using a second processor then entering a rom number will cause memory in the IO processor to be accessed even if the address is not in the range of &8000-&BFFF. This allows simple control of which processor you want to look at. For more information on accessing the different kinds please refer to the configuration command section.

***MDIS <address> (<rom>)**

This will enter the machine code disassembler. This will display over the whole of the screen and will allow you to scroll both forward and backwards through memory displaying disassembled opcodes, by pressing the up and down cursor keys. Any memory locations that do not correspond to opcodes will be shown as EQUB <hex number>. If you press the <TAB> button then the display will toggle to the memory editor MEDIT. To exit the MDIS command you must press <ESCAPE>. The command can be configured whether to understand the CMOS 6502 instruction set with the configurations CMOS and NOCMOS. The information under the MEDIT command on the internal variables also applies to this command.

minimum abbreviation: *MD.

associated commands: *MEDIT, *SDIS, *PDIS

MEMORY COMMANDS

*PDIS <address> <address> (<rom>)

PDIS will allow you to generate labelled disassembled code compatible with the BBC BASIC assembler which will be dumped to the printer and the screen. The code generated will contain two forms of labels ".L<address>" and "W<address>" (*representation of a start of an area of code with no apparent call to it, this can be due to access from outside the addresses specified or access by means of an indirect jump or some other means beyond the scope of this command*). The command operates by creating a stack of the addresses to which JMPs, JSRs & branches arrive at within the code and will produce the labels based on this stack. The routine does its best to be accurate but inevitably it can get confused by embedded data etc. It is however an extremely valuable tool in that it produces code that makes far more sense than with no labels at all. This command makes use of the internal variable @Z to define the address at which to build the stack of label addresses. The contents of this variable will normally be &E00 on the Master series. The stack will always be stored in the IO processor. If it is known that the code does not contain any of the new cmos opcodes then the NOCMOS should be set since this will tend to produce more accurate labeling.

minimum abbreviation: *PD.

associated commands: *SDIS, *MDIS

NOTE: Uses a buffer starting from @Z please refer to notes on the internal variables.

MEMORY COMMANDS

***MEDIT <address> (<rom>)**

The MEDIT command will invoke the Centurion memory editor. This is a full screen editor which can be scrolled to different addresses using the cursor keys. Editing of memory locations is performed in two different ways. You may enter values in Hex to any location, or the "<" and ">" will decrement or increment a location. If you press the <TAB> key then the MDIS command will be invoked. To exit from this routine press the <ESCAPE> key. On exit from this command the internal variable @P will be set to the point where you were in the memory. This means that you can enter this command where you left off by using this variable as the address. You may also re-enter at the original address by using the internal variable @X.

minimum abbreviation: *ME.

associated commands: *MDIS

***SFIND <string> (<address>) (<address>)**

The SFIND command will find a string in memory. If no addresses are specified the whole of memory will be searched. If only one address is specified then that address until the end of memory will be searched and if both addresses are specified locations between these addresses will be searched. If no strings are found the "not found" error will be generated. All occurrences of the string will cause their memory location to be displayed. The number of finds will be stored in the internal variable @Y and the locations of the first 16 finds will be stored in the internal variables @@ to @O.

minimum abbreviation: *SF.

associated commands: *BFIND

NOTE: It is likely that if you are using this command that the string you are searching for will be found in areas such as the keyboard buffer etc.

MEMORY COMMANDS

***BFIND <address> <address> <byte>...**

This will search memory for a sequence of bytes. The addresses to search between must both be specified and at least one byte must be entered to search for. The operation of this command as regards internal variables etc is the same as the SFIND command.

minimum abbreviation: *BF.

associated commands: *SFIND

***MSHIFT <address> <address> <bytes> (<rom>)**

This is a command to move a range of memory. The first parameter is the source address the second parameter is the destination address, the third parameter is the number of bytes to be moved and the last parameter is an optional parameter specifying the rom number to be accessed. This command is intelligent in that it can move memory up or down through overlapping ranges without corruption occurring.

minimum abbreviation: *MS.

MEMORY COMMANDS

***MATCH <address> <address> <bytes> (<rom>)**

This will compare memory from the first address with memory from the second address for the number of bytes specified with <bytes>. Any locations with different values will be displayed along with the contents of memory.

minimum abbreviation: *MAT.

***MFILL <address> <address> (<bytes>)...**

The MFILL command is a general memory fill command, if no bytes are specified then memory between the addresses will be set to 0. If a sequence of bytes is specified then memory will be filled with this pattern of bytes.

minimum abbreviation: *MF.

MEMORY COMMANDS

*NROM (<rom>)

The NROM command is equivalent to the configuration command *CONFIGURE ROM. It will control which of the sideways ROM/RAM banks will be selected by default by many of the commands.

minimum abbreviation: *NR.

*XVECTORS

This will display the contents of the machines vectors stored between &200-&235. If tube memory is selected then the TUBE vectors will be displayed and if the IO memory is selected then the vectors relating to the IO machine is displayed. If the vector is an extended vector pointing into a paged rom then this will be displayed with an additional number specifying the particular rom. For more information on the uses of the vectors refer to the user guide.

minimum abbreviation: *XV.

MEMORY COMMANDS

***XRAM (<number>)...**

This command will display the contents of a list of the non-volatile RAM locations in the Master series (Locations 1 - 49) or if no list is entered then the entire contents will be displayed. These locations contain the Master series configuration values which will set various features of the master on power up or <CTRL><BREAK>.

minimum abbreviation: *XR.

***CRC <address> <address> (<rom>)**

This will calculate and display a C.R.C (Cyclic redundancy check) value from the memory starting at the first address specified and up to but excluding the second address specified.

minimum abbreviation: *CRC

MEMORY COMMANDS

***RECALL (<number>)**

This will recall up to the last 63 commands that have been stored since the RECALL configuration was set or just a specific command identified by a number between 1 & 63. The commands will be displayed in the reverse order so that the last command entered (the RECALL command will not be displayed) will be conveniently just above the command line so that it may be copied and re-entered.

minimum abbreviation: *REC.

associated commands: *FRECALL

OPER COMMANDS

These are operating system utilities covering machine code, memory manipulation and other general utilities.

*ROFF <rom>...

This can be used to prevent access to a number of roms in a similar way to the UNPLUG command. The differences being that with this command you can enter a series of roms which can be identified by either their socket numbers or a string (in quotation marks). The command will keep the rom inaccessible through **<CTRL> <BREAK>** but not through power off. It is possible to switch off the Centurion with this command in which case the only way to regain access to the Centurion and any other roms switched off is to turn off your computer.

minimum abbreviation: *ROFF

associated commands: *RON, *KEEP

*KEEP (<rom>)...

KEEP is a very useful method of eliminating interference caused by roms fitted in the machine. The command will turn off ALL roms in the machine with the following exceptions. The BASIC rom, the current filing system rom, the current language, the Centurion rom and any roms specified in the list entered to the command. This global effect will generally instantly dispose of any software problems caused by wayward roms.

minimum abbreviation: *KEEP

associated commands: *ROFF, *RON

OPER COMMANDS

*RON (<rom>)...

This will reactivate any rom switched off with the ROFF command. If no list of roms is entered then the command will turn on all roms that have been switched off.

minimum abbreviation: *RON.

associated commands: *ROFF, *KEEP

*SEND <rom> <command line>

Despite the number of rom management features included in the Centurion it may still be the case that you have a rom in your Master series computer which has a command in it that cannot be conveniently be accessed, this command will allow you to send the command directly to any rom including the Centurion (filing system commands cannot be accessed with this method).

minimum abbreviation: *SEND

associated commands: *ROFF, *RON, *KEEP

OPER COMMANDS

*SET <@A-@Z> <number>

This command will set one of the Centurion's internal variables to a particular number (including one of BASIC's INTEGER variables). This is useful either if you want to constantly access a command with a particular parameter or as a means of passing commands to Centurion from BASIC. The BASIC command OSCLI can also be used for this purpose but can be rather obtuse.

minimum abbreviation: *SET.

associated commands: *ISET, *SYS.

*ISET <A%-Z%> <number>

This is the Centurion's method of interfacing with the BASIC language. Any result of a Centurion command which set one of the internal variables can have this result transferred to an INTEGER variable for use by BASIC.

minimum abbreviation: *IS.

associated commands: *SET, *SYS

OPER COMMANDS

*SYS

The SYS command will display the contents of all the Centurion's internal variables. For a description of the internal variables and their uses see the section on internal variables.

minimum abbreviation: *SYS

*SLOW (<0 - 255>)

This command will slow down the operation of the computer by an amount depending on the number entered. The higher the number the slower the computer will operate, it is likely you will not be able to set the number as high as 255 because all operations will cease. At lower numbers it is useful for games etc. Entering *SLOW without a parameter or 0 as a parameter will terminate the command.

minimum abbreviation: *SL.

OPER COMMANDS

***BASE <number>...**

The BASE command will accept a list of numbers in any of the forms allowed by the Centurion and will display them in hex, decimal and binary.

minimum abbreviation: *BASE

VIDEO

This command is a simple one that can save a lot of eye strain when using word processors in 80 column mode on an inadequate monitor or TV set. The command will invert the colours on screen which generally make things easier to read. This command will operate in modes 0 - 6 and modes 128 - 134 but not modes 7 or 135 since these modes do not have a reversible palette.

minimum abbreviation: *VI.

OPER COMMANDS

***PC <byte>...**

The PC command will send commands directly to a printer in a manner far easier than from the BASIC language and will make it possible to set up the printer where there are no facilities for this. The BASIC equivalent of this command is VDU 2, 1, byte, 1, byte...., 3 or as an alternative a lot of pressing control key combinations.

minimum abbreviation: *PC

***XCHAR <ascii code>...**

The XCHAR command will display the character definition of a number along with the Hex, Decimal and Binary numbers that would be used with the VDU23 command to define the character.

minimum abbreviation: *XC.

OPER COMMANDS

***XENVELOPE (<number>)...**

This command will display the definitions of any or all of the 16 sound envelopes available for sound effects on the Master series. The definitions are displayed in the form that they would be entered to the BASIC ENVELOPE statement.

minimum abbreviation: *XE.

***BREAK**

This command will perform a clear memory reset which is useful if you wish to ensure data security, by not leaving memory available to be examined.

minimum abbreviation: *BR.

ADFS COMMANDS

The Advanced Disk Filing System on the Master Series is a powerful alternative to the standard DFS. It offers 60% more capacity per disk surface, unlimited files and the ability to address both sides of a disk as a single disk. However it does suffer badly on the Master through a lack of basic utilities such as Format and Verify and also DFS users will be used to the WIPE facility which is not available on the ADFS. The Centurion makes available these utilities and also a selection of more advanced features designed to make use of the ADFS simpler and more attractive.

***AFIND <list spec>**

This command will search through the directory structure of your ADFS disk from the directory you are in and all sub directories, for any file which match the specification entered. Every file found will be displayed in a form allowing you to tell which directory the file is in. By entering "*" as the specification from the \$ directory, you will of course get a display of all the files on your ADFS disk. The command will display a count of the number of files and a separate count of the directories displayed and also the space taken up on the disk by the files and directories. Under the ADFS each directory will take up 5 sectors. The output from this command can be sent to your printer by pressing <**CTRL**>**B** prior to the entry of the command.

minimum abbreviation: *AF.

associated commands: *TREE

ADFS COMMANDS

***TREE**

This command will produce a display of the directory structure of an ADFS disk.

minimum abbreviation: *TR.

associated commands: *AFIND

***AWIPE <list spec>**

This command will allow you to selectively delete files which match the specification entered within an individual directory. The filename of each file which matches the list specification will be displayed and you will be able to press "Y" to delete the file or "N" to move to the next file.

minimum abbreviation: *AW.

***DATESTAMP**

This command will provide automatic datestamping on files saved until the <BREAK> key is pressed or a new filing system is selected. It is provided as an alternative to setting the DATEFILE configuration.

minimum abbreviation: *DATES.

associated commands: *DINFO, *DATEFILE

***DATEFILE <filename>**

This command will datestamp an individual file on an ADFS disk. For details on datestamping see the Date configuration command.

minimum abbreviation: *DATEF.

associated commands: *DINFO, *DATESTAMP

ADFS COMMANDS

***DINFO <list spec>**

This command will display the date and time of creation of any files datestamped either with the DATESTAMP command or automatically datestamped when saved by the Centurion. The dates will be displayed in the format HH:MM:SS DD/MM/YYYY

associated commands: *DATEFILE, *DATESTAMP

minimum abbreviation: *DIN.

***AFORM <track> (drive)**

This command provides an inbuilt formatter for the ADFS. The number of tracks selected can be 40, 80 or 160, and the drive number can be 0 or 1. This utility will verify each track after formatting it.

minimum abbreviation: *AFO.

associated commands: *VERIFY

DUTILS COMMANDS

The DUTILS commands will help maintain your disk system by providing utilities that can recover data from corrupted disks etc. The commands are relevant to both ADFS and DFS users since they will automatically sense which type of disk is in the drive.

***SECTLOAD <address> <track> <sector> (<sectors>)
<drive>)**

This command will load number of sectors off a disk, any errors found will be ignored. The command will run over any number of tracks. The sectors specified will be loaded to the address specified in the processor selected with the USETUBE or USEIO configuration commands.

minimum abbreviation: *SECT.

associated commands: *SECTS

***SECTS <address> <track> <sector> (<sectors>)
<drive>)**

This command will save a number of sectors to a disk, any errors found will be saved from the address specified in the processor selected with the USETUBE or USEIO configuration commands.

minimum abbreviation: *SECTS.

associated commands: *SECTLOAD

DUTILS COMMANDS

*PBACKUP <drive> <drive> <start track> <end track>

This command will invoke a powerful protected disk backup system capable of making copies of most protected disks. You may copy both sides of an 80 track disk at once by making the end track 159. The start track should normally be 0. Before attempting to run a backup copy of a protected disk you should place a write protect tab over the notch on the disk, this is in case the disk is protected by writing to itself.

minimum abbreviation: *PB.

WARNING: It is illegal to copy a commercial diskette unless you own the disk and are making the copy for safety purposes only.

*FSFIND <start track> <end track> <string>

This command will perform a search of a disk track by track for the string specified. Where ever the string is found the display will show the track, sector and offset within the sector of the string. This command is capable of finding a string even if it is split over two tracks.

minimum abbreviation: *FSF.

associated commands: *FBFIND

DUTILS COMMANDS

***DENSITY <track> (<drive>)**

The DENSITY command will read a track of the drive specified and display either "Double" or "Single" for double and single density, or if there is no format on the track then the "Bad track" error will be produced.

***IDDUMP <track> (<drive>)**

This command will display a dump of the track ID's of a particular track. Each sector will have an ID which will specify a track number, a head number, a sector number and a length. For further information on ID's you are referred to the 1770 data sheets. On exit from this command the Centurion internal variable @Y will be set to the number of ID's read and upto the first 16 internal variables will be set to the track and sector number of each ID read. The high byte of each variable will be set to the track number and the low byte of each variable will be set to the sector number.

minimum abbreviation: *ID.

DUTILS COMMANDS

***PBACKUP <drive> <drive> <start track> <end track>**

This command will invoke a powerful protected disk backup system capable of making copies of most protected disks. You may copy both sides of an 80 track disk at once by making the end track 159. The start track should normally be 0. Before attempting to run a backup copy of a protected disk you should place a write protect tab over the notch on the disk, this is in case the disk is protected by writing to itself.

minimum abbreviation: *PB.

WARNING: It is illegal to copy a commercial diskette unless you own the disk and are making the copy for safety purposes only.
Due to the constant development of disk protection we cannot guarantee the effect of this command.

***FSFIND <start track> <end track> <string>**

This command will perform a search of a disk track by track for the string specified. Wherever the string is found the display will show the track, sector and offset within the sector of the string. This command is capable of finding a string even if it is split over two tracks.

minimum abbreviation: *FSF.

associated commands: *FBFIND

DUTILS COMMANDS

***FBFIND <start track> <end track> <bytes>...**

This command will perform a search of a disk track by track for the bytes specified. Where ever the bytes are found the display will show the track, sector and offset within the sector. This command is capable of finding a set of bytes even if they are split over two tracks.

minimum abbreviation: *FBF.

associated commands: *FSFIND

***EDSECT <track> <sector> (<drive>)**

This command will invoke a disk sector editor compatible with both the ADFS and DFS. Once a sector is loaded you may move about the data in the sector with the cursor keys. The data may be edited by entering hex digits if the edit mode (set by pressing <TAB> is HEX) or by simply typing the character you want to enter. You may save the sector you are on by pressing <COPY> (you will be asked to confirm that you want to save a sector). You may move to the next or previous sector by using <SHIFT> along with the up or down cursor keys. Any errors encountered when reading a sector from the disk will be displayed.

minimum abbreviation: *EDS.

DUTILS COMMANDS

***REPAIR <track> (<drive>)**

This command will attempt to read in a track from your disk drive regardless of any errors the track may contain and then it will format the track either to the ADFS or DFS standard and rewrite the data. This should have the effect of repairing a track. Due to the nature of disk errors it is impossible to guarantee the effect of this command and you should make a backup of your disk if it contains vital data using the PBACKUP command before attempting to repair the disk.

NOTE: If a track contains no data at all, then the "Bad track" error will be produced.

minimum abbreviation: *REP.

associated commands: *PBACKUP

BASIC COMMANDS

The commands listed below are of most use to the BASIC programmer, though some of them may have alternative uses.

***BAD**

This is a utility to repair BASIC programs that will not list or as a rule run because their structure in memory is incorrect. A BASIC program is held in memory in the following format.

PAGE : 13

PAGE+1: line number high byte or &FF if end of program.

PAGE+2: line number low byte.

PAGE+3: Length of line.

PAGE+4: Start of text for line.

The number held at PAGE + 3 should normally point to a location containing the number 13 signifying the start of the next line. The command *BAD will look through memory and attempt to make sense of occasions when this structure is wrong.

minimum abbreviation: *BAD

BASIC COMMANDS

***XLIST**

Sometimes programmers will attempt to stop you listing a BASIC program by inserting control characters in it which when listed will cause various effects to occur. This command will replace all such characters will the "l" character thus making the program listable.

minimum abbreviation: *XL.

***SEARCH <string>**

The SEARCH command will search through a BASIC program and display the lines which contain the search string. Lines which contain the search string more than once will still only be displayed once. The search string can contain BASIC keywords, the rule being that the string is tokenised as if it was a BASIC line.

minimum abbreviation: *SEA.

associated commands: *SREPLACE, *GREPLACE

NOTE: This command is not TUBE compatible

BASIC COMMANDS

***SREPLACE <string> <string>**

This command performs a selective search and replace on a BASIC program for each occurrence of the first string entered to the command in each line of the program you will be asked whether you want it replaced with the second string entered to the command. The search and replace string operate in the same manner as described above for the SEARCH command.

minimum abbreviation: *SRE.

associated commands: *SEARCH, *GREPLACE

NOTE: This command is not TUBE compatible

***GREPLACE <string> <string>**

The GREPLACE command is a selective search and replace command which operates in the same manner as the SREPLACE command but it will not ask you whether or not you want each replacement to take place. It is as such a powerful aid if used properly but a dangerous one if used wrongly where the BASIC program has not been saved for safety.

minimum abbreviation: *GRE.

associated commands: *SEARCH, *SREPLACE

NOTE: This command is not TUBE compatible

BASIC COMMANDS

*FDELETE <first> (<last>)

This command is a faster and more versatile version of the BASIC DELETE command. The BASIC DELETE suffers from the problem that when you enter a range of lines it will try to delete on an individual basis each and every possible line number between the range specified, so for example DELETE 2000, 4000 will try and delete 2001 lines whether they exist or not, and for each of the lines that does exist it will have to move much of the computers memory downwards. This of course makes it very slow and impractical to use when deleting large numbers of lines.

This command operates in two ways. If entered with only the first line number, then all the BASIC program after and including this line will be instantly deleted. If you enter the FDELETE command with both the first and last line numbers, then the command will delete these lines almost instantly by means of a far superior line deletion method than that present in BASIC.

minimum abbreviation: *FDEL.

*MERGE <filename>

This command will read a BASIC program from your filing system and merge it with the program currently in memory. The effect of the merge is that all lines from the file will be inserted in the program in memory replacing any lines with the same line numbers.

minimum abbreviation: *MER.

associated commands: *BINSERT, *BAPPEND,
 *PARTSAVE

BASIC COMMANDS

***BINSERT <filename> <line number>**

The BINSERT command allow one BASIC program to be inserted inside another after a particular line number. It should be noted that this command will not change the line numbers of the BASIC program as it inserts. You will have to do this by typing RENUMBER.

minimum abbreviation: *BIN.

associated commands: *MERGE, *BAPPEND, *PARTSAVE

***BAPPEND <filename>**

This command will append a BASIC program on disk onto the top of one held in memory. The new program created will probably contain out of sequence line numbers at the boundary between the old program and the program appended. The RENUMBER command should be issued to return the lines to a sensible order.

minimum abbreviation: *BAP.

associated commands: *BINSERT, *MERGE, *PARTSAVE

BASIC COMMANDS

***PARTSAVE <filename> <first> (<last>)**

The PARTSAVE command will save part of a BASIC program from the first line specified to the last line specified or till the end of the program if no last line is specified. This command in conjunction with the commands above and the command to partially renumber a BASIC program gives you the ability to "mix & match" bits and pieces of BASIC programs to your hearts content.

minimum abbreviation: *PAR.

associated commands: *BINSERT, *MERGE, *BAPPEND

***BLIST <filename>**

This command is to cater for those situations where you are editing a BASIC program in memory and you want to see what is in another BASIC program on the disk. The command will display a listing of the BASIC program which can also be sent to the printer if desired by typing <CTRL>B.

minimum abbreviation: *BLI.

BASIC COMMANDS

***MEMORY**

This command will produce a display of the memory values important to a BASIC program. These are the default value of PAGE, The actual value of PAGE, TOP, LOMEM, HIMEM, the first free location in memory, the amount of memory free, the amount of memory used for variables and the amount of memory used by the BASIC program.

minimum abbreviation: *MEM.

***FLUSH**

This command will clear the resident integer variables A% - Z% to zero.

minimum abbreviation: *FL.

BASIC COMMANDS

*SHRINK

The SHRINK command will act on a BASIC program to reduce the amount of memory taken by it. There are two options available 'Remove Rems (Y/N)" which will control whether or not remarks in the BASIC program will be removed and "Join lines (Y/N)" which will control whether or not lines of a BASIC program will be joined together where possible. Since this command removes excess spaces it will sometimes create lines that you will not be able to enter into a BASIC program without adding spaces.

minimum abbreviation: *SHR.

associated commands: *BEXPAND

NOTE: If your BASIC program contains calculated goto's e.g GOTO 200+10*I then it may not work after SHRINK is used. This is no big deal since a simple RENUMBER on a program containing calculated goto's will have the same effect. You should not really be writing such badly structured programs in the first place.

*BEXPAND

The BEXPAND command is a form of inverted SHRINK command. It's action is to go through each line of a BASIC program and break up multiple statement lines into single statement lines. This is useful if you want to do anything with a compacted program since due to the long lines they are almost impossible to edit or understand.

minimum abbreviation: *BEX.

associated commands: *SHRINK

***LVAR**

This command will list all the active variables in a BASIC program (none will be defined after <BREAK> CLEAR, editing a BASIC line etc). Each variable will be listed according to their type. The resident integer variables will not be displayed. Strings will have their contents displayed with undisplayable characters as "." so as not to cause screen corruption. Integer variables will be displayed in hex and decimal in a format depending on the value of @%. Real variables will be displayed in hex and where possible in decimal in a format according to the value of @%. Arrays will have their dimensions but not their contents displayed.

minimum abbreviation: *LV.

NOTE: This command is not TUBE compatible.

***NUMBER <first> <last> <start> <inc>**

This command will partially renumber a BASIC program between the <first> and <last> lines. These lines will now start at <start> and have an increment of <inc>. This command generate an error if you try and create a situation where the line numbers would start overlapping.

minimum abbreviation: *NUM.

NOTE: This command is not TUBE compatible

BASIC COMMANDS

*MLINES <first> <last> <after line>

This command allows you to move a section of a BASIC program from one part of the program to another. The resulting program will contain out of sequence line numbers and should be numbered prior to use. You can move lines to the start of the program by specifying <after line> to be 0.

minimum abbreviation: *ML.

associated commands: *CLINES

*CLINES <first> <last> <after line>

This command is identical to the MLINE command but will produce a copy of the segment of the program selected without deleting the lines selected from the program. The main use of this feature is when you want to play safe whilst altering the structure of a BASIC program, after all deleting lines using the FDELETE is no problem.

minimum abbreviation: *CL.

associated commands: *MLINES

BASIC COMMANDS

***BMOVE <address>**

The BMOVE command allows you to move a BASIC program in memory. The address to move the program to should be on a PAGE boundary. On exit from this command the value of PAGE will be set to the address.

minimum abbreviation: *BM.

***BCHECK <filename>**

The BCHECK command allows you to verify that a BASIC program in memory is the same as one on the disk. If the program is not identical then a "verify error" will be generated.

minimum abbreviation: *BC.

BASIC COMMANDS

***FLIST**

This command will produce a "prettier" looking listing of a BASIC program in memory. It will divide up program statements onto separate lines. The listing can be sent to the printer by typing <CTRL>B before issuing the command. The command has an additional use in that it can list "bad programs" which the BASIC LIST command is not able to.

minimum abbreviation: *FL.

***XON**

The XON command will activate an alternative handler for errors which occur whilst running BASIC programs. Using this command errors which occur even if the ON ERROR statement is in operation will cause MODE 135 to be selected and the line at which the error occurred to be displayed at the top of the screen. This command can be of particular use to people (especially beginners) attempting to debug a program which uses ON ERROR because the work can be done without having to change the program around.

minimum abbreviation: *XON

associated commands: *XOFF

BASIC COMMANDS

***XOFF**

The XOFF command will turn off the extended error handling if it has previously been enabled with the XON command.

minimum abbreviation: *XOFF

associated commands: *XON

COMMAND SUMMARY

Configuration Commands

JSE IO
JSETUBE
MOSRAM
NOMOSRAM
FILERAM
NOFILERAM
SHADOW
NOSHADOW
DATE
NODATE
AUTOSAVE
NOAUTOSAVE
RECALL
NORECALL
CMOS
NOCMOS
ROM

COMMAND SUMMARY

Operating System

ROFF <rom>...
KEEP <rom>...
RON <rom>...
SEND <rom> <command line>
SET <@A-@Z> <number>
ISET <A%-Z%> <number>
SYS
SLOW (<0 -255>)
BASE <number>...
VIDEO
PC <byte>...
CHARDEF <Ascii code>
XENVELOPE (<number>)...
BREAK

COMMAND SUMMARY

Dutils commands

SECTLOAD <address> <track> <sector> (<sectors>) (<drive>
SECTSAVE <address> <track> <sector> (<sectors>) (<drive>
DENSITY <track> (<drive>
IDDUMP <track> (<drive>
PBACKUP <drive> <drive> <start track> <end track>
FSFIND <string> <start track> <end track>
FBFIND <start track> <end track> <byte>
EDSECT <track> <sector> (<drive>
REPAIR <track> (<drive>

ADFS Commands

AFIND <List Spec>
TREE
AWIPE <List Spec>
DATESTAMP
DATEFILE <filename>
DINFO <List Spec>
AFORM <size> (<drive>
AVERIFY (<drive>
ABACKUP <drive> <drive>

COMMAND SUMMARY

BASIC Commands

BAD
XLIST
SEARCH <string>
SREPLACE <string> <string>
GREPLACE <string> <string>
FDELETE <first> (<last>)
MERGE <filename>
BINSERT <filename> <after line>
BAPPEND <filename>
PARTSAVE <filename> <first> (<last>)
BLIST <filename>
MEMORY
FLUSH
SHRINK
BEXPAND
LVAR
NUMBER <first> <last> <start> <inc>
MLINES <first> <last> <after line>
CLINES <first> <last> <after line>
BMOVE <address>
BCHECK <filename>
FLIST
XON
XOFF

COMMAND SUMMARY

File Commands

RSAVE <rom> (<filename>)
RLOAD <filename> <rom>
KSAVE <filename>
KLOAD <filename>
FONTSAVE <filename>
FONTLOAD <filename>
RECALL (<number>)
FRECALL <filename>
SCRAMBLE <filename> <password> (<filename>)
UNSCRAMBLE <filename> <password> (<filename>)
ALTER <filename> <load> <exec>
CHECK <filename> <filename>
FILTER <filename> <filename> <first> <last> (<byte>)

Memory Commands

MDIS <address> (<rom>)
PDIS <address> (<rom>)
MEDIT <address> (<rom>)
SFIND <string> (<address>) (<address>)
BFIND <address> <address> <byte>...
MSHIFT <address> <address> <bytes> (<rom>)
MATCH <address> <address> <bytes> (<rom>)
MFILL <address> <address> <bytes>...
NROM (<rom>)
XVECTORS
XRAM (<number>)...
CRC <address> <address> (<rom>)

COPYRIGHT & CONDITIONS

Chalice software and Swindon ITEC LTD reserve the right to make improvements in this product at any time and without prior notice. Every effort has been made to ensure the accuracy of the material contained within this product, however experience shows that some errors always remain to be discovered. If you find any errors or have any suggestions on how to improve this product please contact:

Swindon ITEC LTD, 6 Oppenheimer Centre, Greenbridge Road,
Swindon SN3 3JD. Tel. (0793) 611808.

No part of this product may be reproduced by any means without the prior consent of the copyright holders.

DISCLAIMERS

The liability of Chalice software and Swindon ITEC LTD shall be limited to the cost of replacement of the product should it be found to be defective. In no event shall liability be accepted for indirect or consequential loss arising from this products performance.

Produced by:
Swindon ITEC Ltd
6 Oppenheimer Centre
Greenbridge Road
Swindon
SN3 3JD
England.
tel. : (0793) 611808