MAX

Multi-Axis eXerciser

OPERATION MANUAL
MAX001c

TABLE OF CONTENTS

GENERAL DESCRIPTION

ORMEC's Multi-Axis eXerciser (MAX) software package is a development tool for implementing and maintaining multi-axis motion control systems. It runs on the IBM Personal Computer (PC) and compatibles using the MS-DOS operating system, and aids system designers in developing Motion Programming Language (MPL) motion control routines and communicating with multi-axis systems.

MAX offers system designers three major functions which simplify and speed development of multi-axis motion control systems:

- Servomotors in a multi-axis motion control system can be "exercised" from the keyboard of a personal computer by selecting an axis of motion and **interactively** executing MPL commands

- MPL routines developed using WordStar* or another file editor can be **downloaded** from a disk file to the motion controller's program buffer; or MPL routines developed interactively can be **uploaded** to a disk file

- System designers can write **command files** combining MPL and MAX commands which will "program" a multi-axis motion control system from a disk file

Each of these functions is detailed in this document.

MAX communicates with ORMEC motion controllers through a serial channel. The user is able to specify which axis is active and characters typed at the console are sent to that axis only. The internal axis device drivers configured when the system is installed will use the serial channel for the axis selected.

MAX is invoked at the system command prompt by simply typing "MAX".

Example: A>**MAX<cr>**

(What the user types is in bold.)

* WordStar is a registered trademark of Micropro International

CREATING MULTI-AXIS MOTION INTERACTIVELY

The ability to interactively "exercise" a multi-axis servomotor system is a great aid to system development. When used interactively, MAX passes console communications directly to the currently selected system. The user communicates with several ORMEC motion controllers, one at a time, by individually selecting them and then typing the appropriate motion command information at the console of the host computer. This provides the system designer an integrated development environment for writing and testing motion control routines.

MAX sends information typed at the console directly to the active motion controller except when the user types the MAX command character (backslash "\"). When a backslash is encountered, it is intercepted by MAX, and the next character is interpreted as a MAX Command and the appropriate action taken.

When MAX is started at the console, it begins by presenting a menu of commands to the user. This menu of commands is available at any time by using the "Help" function.

The user can interrupt MAX at any time by typing an "escape" (ESC) character at the system console. For example, if MAX is asking for information (like a filename) because of a command request from the user, the user can hit the ESC key to abort the operation and return to normal pass-thru mode. If MAX is downloading an MPL file, hitting ESC will terminate the download and put the axis back into command mode from whatever mode it was in during the download.

2.1 GENERAL COMMANDS

\?
Help Menu;
This command produces a list of available commands at the system console.

\=<axis>
Select Axis;
To select an axis and make it active, the user enters the \= command. The next character entered specifies which axis to make active. The different axes are named using a single letter. The first axis is axis "A", the second is axis "B", etc. You can always tell what axis you are on by noting the axis name in the prompt (e.g. A}) or looking at the \= entry in the Help Menu. MAX allows the user to enter an axis ID as either an upper or lower case letter, but it always sends an an upper case axis ID to the PMC.

\X
Exit;
The Exit command returns to the host operating system.

2.2 MISCELLANEOUS COMMANDS

\I — Initialize;
The Initialize command restarts the MAX program. All pending characters from the axis buffers are cleared, the active axis is set to the letter 'O', and all toggles are reset. This is useful if the user wants to start again from the beginning.

\% — Set Baud Rate;
This command allows the user to change the baud rate of the current axis communication channel. A number of choices are displayed and by entering the letter corresponding to the desired serial communication speed the baud rate is immediately changed.

\B — Binary (set B7 of next byte);
The Binary command sets the top bit of the next byte sent to the active axis. The need for this command arises from the fact that most ASCII terminals do not allow the user to define the top bit and at times the user may want to communicate to the axis in binary mode. To insure compatibility with the majority of video display terminals, MAX will strip the top bit of each character sent from the console. For each byte the user wants to send to the axis with the top bit set, a \B command must be entered first.

\\ — Send a Backslash Character
This command sends the backslash character to the active axis.

2.3 PROGRAM BUFFER EDITING KEYS

MAX utilizes the IBM PC's ARROW keys, along with the INSERT and DELETE keys, to simplify program buffer editing. The chart below shows how each of these keys functions and what characters are being sent to the PMC.

Key	Function	Character to PMC
Up Arrow	Cursor left one character until at left margin. Then cursor moves up one line at a time.	Control-H
Left Arrow	Cursor left one character	Control-H
Right Arrow	Cursor right one character	Control-I or Tab
Down Arrow	Cursor moves down one line at a time.	LINEFEED

Insert	Allows text to be inserted at point of the cursor. A second insert must be entered to exit Insert command. Pressing ESCAPE cancels the insert command.	Control-V
Delete	Deletes unwanted characters from the cursor to the end of the line.	Control-K

UPLOADING/DOWNLOADING MPL FILES

A powerful feature of MAX is its ability to support uploading and downloading of MPL motion routines. All motion routine files should have the file extension ".MPL". An example of this filenaming convention is ROUTINE.MPL. The ability to upload and download information between an ORMEC motion controller's program buffer and a computer file gives the user a convenient method for storing and maintaining a tested set of MPL routines. MPL motion routines can be written in standard text files and then easily transferred to a selected axis for execution.

To support documentation of MPL files, inline comments are supported. Simply separate the comment from the MPL command with a "space" or "tab" character, and MAX will not send the comment when the file is downloaded. This allows the user to maintain one disk file for both MPL programs and complete line by line documentation of the application. An example is as follows:

```
 <tab>
 <tab>          Powerup routine
 <tab>
@@_Powerup      Label power-up routine "@" and add comments.
TV180           Set velocity loop gain to 180.
TCVC            Set the velocity loop compensator to "C".
TP30            Set the position loop gain to 30.
TF150           Set the feedforward gain to 150.
TCP0            Set position loop compensation to 0.
SM2             Set the PMC to the POSITION control mode.
D100            Delay 100 milliseconds.
N0-             Define the PMC's current physical position as 0.
E               Exit  the  powerup routine.
```

For an example of a typical application, and the kind of documentation recommended by ORMEC, see the back page of the Programmable Motion Controller data sheet.

3.1 PROGRAM MAINTENANCE COMMANDS

All program maintenance commands start with the characters \P. When the P is typed, MAX will prompt for the rest of the command by completing the word "Program:". The list of commands follows:

\P? Display Program Help Menu;
This command is like the main MAX \? command but gives a list of Program Maintenance Command Level.

\PF Program File Directory;
This command requests a file directory of .MPL files in the current directory of the host computer system.

\PD<filename> Program Download;
This command will take the contents of the specified .MPL file and overwrite the program buffer of the active motion controller. This operation can take many seconds, therefore activity dots are sent to the console periodically.

\PU<filename> Program Upload;
This command is just like the Program Download command except it is reversed in direction. The information is transferred from the active motion controller's program buffer to the host computer disk file. One note of caution: this command will not double check with the user that he is using the name of a current file on the disk. Instead, it will rename the old file to <filename>.BAK and write a new <filename>.MPL. This means that if the user performs this operation twice, he will destroy any existing file by the name of <filename>.MPL.

WRITING MAX COMMAND FILES

An advanced feature of MAX is its ability to operate from command file information similarly to the way it operates with information typed at the console. Users can "program" a multi-axis motion control system by creating a disk file containing Motion Programming Language (MPL) and MAX commands. The disk file contains information which is very similar to the information typed directly from the console interactively, allowing straightforward program development.

A MAX Command File contains the similar character sequences to those that are entered from the system console. This means that the file contains a combination of MPL commands to be sent to the

active axis and MAX commands. There are, however, a few minor differences. A \I command will abruptly stop the processing of that file and reinitialize MAX.

Another construct in the MAX command file deals with the ability to synchronize instructions with the active axis. MAX is able to wait for the active axis to return a prompt before proceeding. This allows MAX to wait for certain activity such as the completion of a motion or a time delay to occur before proceeding with more commands for that axis.

MAX is told to wait for a prompt before proceeding by placing a dollar sign ($) in the command file. When MAX finds the $, it will wait for the prompt from the active axis before proceeding. Therefore, most of the commands in the file should be preceded by a $ character. To let the user know how his MAX synchronizing is operating, MAX will replace each prompt character that it had been waiting for with a $ character when displayed at the system console.

With the $ synchronization method it may be necessary to tell MAX to continue processing a command file if it is waiting for a prompt which is not sent by the active axis. This happens when a dollar sign was placed in the command file at a point where no prompt will ever come from the axis without further instructions. To tell MAX to stop waiting and continue, the user enters an ESC character at the system console.

The following keystrokes may be entered during the execution of a MAX command file:

Q	Quit processing of the file immediately
R	Repeat this MAX file when the end is reached.
S	Stop processing at the end of this run.
ESC	continue processing when MAX is waiting for a prompt from the active axis. (A dollar sign ($) in the MAX file)

4.1 COMMANDS FOR MAX COMMAND FILES

\M<filename> Run a MAX file;
This command reads a MAX command file and prepares to process it. If a file had previously been read, MAX will show its name in parentheses when it asks for the file to read. At this point, the user may enter a new filename or by hitting <cr> the previous file will be processed again. After the file is read in properly, MAX will prompt the user to enter one of four characters:

I (Init w/cr) to send a <cr> to the active axis before processing the MAX command file,

Q (Quit) if the user decides not to process the file,

S (Start) to start processing the MAX command file without an initial <cr>, or

R (Repeat) to tell MAX to start processing the command file and when it gets to the end of the MAX file to repeat it again until the user breaks the execution with one of the other characters. Note that an initial <cr> is sent to the active axis before processing of the MAX command file is repeated.

\C Console Toggle (Console output active flag);
The Console Toggle command toggles the console output active flag. Entering a \C will instruct MAX to stop displaying all characters to the system console. This can be useful if the user does not want MAX to slow down for serial communication to the console or if the operation of a MAX Command file is to be hidden. By entering \C again, MAX will re-enable the console output.

\D Delay;
This command is used to suspend processing of MAX for a short period of time (usually one second). This feature is of particular use in MAX command files to wait for an event to occur.

\E Send an ESC Character
This command sends the ESC character (^[) to the active axis. This is useful since most file editors on the host computer operating system do not allow non-printing ASCII characters to be entered into the file.

\F File Directory List;
This command will provide a list of .MAX command files available on the current disk of the host computer system. This is useful in case the user forgot the name of the file.

ADVANCED METHODS FOR STARTING MAX

There are several more advanced methods documented below for starting MAX rather than invoking it by typing MAX at the system prompt level.

With this method of calling up the program, MAX will initialize itself to the startup configuration and look for a file on the system called STARTUP.MAX. If it finds STARTUP.MAX in the current directory, MAX will run this file as a MAX command file.

The purpose for this file is to allow the user to specify a different default configuration on startup by using MAX commands in the STARTUP.MAX file to change certain flags and modes of operation of MAX. A typical example of the contents of this file is:

```
\=a\%g\c
```

This command sequence selects axis 'A', sets the baud rate to 9600K baud, and turns on the console output. The last step is needed because MAX intentionally turns off console output if it processes a STARTUP.MAX file.

If STARTUP.MAX is not found, a Help Menu for the first command level is sent to the system console and MAX enters the communications mode for an undefined axis.

If MAX is invoked with a command trailer the information after the program name is used as a MAX command stream. Therefore, instead of using a STARTUP.MAX file for the initialization, a user could enter the following:

```
A>MAX \=a\%g\c
```

and MAX would perform the same steps as described above for a STARTUP.MAX file. This capability to specify a MAX command stream is especially useful when the user wants to run a different MAX command file right at startup. To run a MAX command file this way a user would enter the following:

```
A>MAX \mINIT\ns
```

where INIT.MAX is the file to run. The string \n is used to place a carriage return in the command stream (the \m command needs one to denote the end of the filename) and the following character 's' is also for the \m command to start the processing of the INIT.MAX file.