



## DATA FORMATS

MA520 receives data and commands from the host in 8-bit bytes. It interprets the data in the following format:

		BIT							
		7	6	5	4	3	2	1	0
Text Mode	0	ASCII CODE							
Graphics Mode	1	OPERATION						MODE	

A "0" in the most significant bit causes MA520 to interpret the rest of the byte as an ASCII character, which it displays on the screen at the alpha cursor location. If the rest of the byte is one of the 14 ASCII control characters, the MA520 recognizes it and takes the appropriate action.

A "1" in the most significant bit indicates that the next 5 bits are a graphics operation code. There are 20 operation codes; one governs the text display, the rest control graphics. The remaining two bits indicate a mode within an operation (for example, mode 0 of operation 3 turns a designated point on, mode 1 turns it off.) To command MA520 from high level software requires nothing more than subroutines that translate high level instructions to several bytes of MA520 operation code and coordinate data. For instance, to draw a line takes just 5 bytes from the host to the MA520. A complete list of operations, modes and the bytes they require appears on the page opposite.

## MEMORY SPACE

MA520 memory space is allocated as follows:

<b>0000-3FFF</b>	PROM
<b>4000-7FFF</b>	User
<b>8000-F800</b>	Screen Image
<b>F801-F93F</b>	ScreenWare
<b>F940-FD3F</b>	User
<b>FD40-FFFF</b>	ScreenWare

Memory address from 4000 to 7FFF point to one of two 16K byte banks of RAM buffer. A command in ScreenWare lets the user switch from one buffer to another. The first of the two 16K banks has allocated 1.5K bytes for a user-defined character font or macros. The balance of this bank and the entire other bank is for the user and optional software expansions.

## INTERCONNECTIONS/PORT ADDRESS

As shipped, the ports of MA520S (S-100 version) are mapped as F0H and F1H in your system's port address space. You can change the default addresses by modifying 2 port address jumpers. MA520M (Multibus version) is I/O or memory mapped (jumperable) to any 2-port boundary of 256 ports (8080/Z80), any 2-port boundary of 64K ports (8086) or any 2-byte boundary of 1 meg address space.

The lower-addressed port is an 8-bit data port for data to and from MA520. The higher-addressed port is a control port for handshaking communication. Writing a "1" to the least significant bit of the control port causes the MA520 to reset.

## INTERRUPTS

MA520 is shipped with three maskable interrupts enabled for data-to/from-host and lightpen-firing. Simple jumper connectors enable a maskable data-to-host interrupt and a non-maskable real time interrupt.

MA520 also generates signals that support interrupt-driven protocols in the host via vectored interrupt lines on the S-100 or Multibus.

## LIGHTPEN INTERFACE

MA520 has four pins for connecting a lightpen, as follows:

1. Lightpen strobe (rising edge — falling edge jumperable)
2. Ground
3. Lightpen enable (active high)
4. Power source (100 ma, +5V)

For a lightpen without an enable signal, pins 3 and 4 can be jumpered together permanently enabling the pen.

## ATTACHING A MONITOR

MA520 drives a composite video or direct drive monitor. Signals appear on 6 pins:

- |                                |                               |
|--------------------------------|-------------------------------|
| 1. RS-170 composite video      | } For Composite Video Monitor |
| 2. Ground                      |                               |
| 3. TTL video                   | } For Direct Drive Monitor    |
| 4. Ground                      |                               |
| 5. Horizontal sync (TTL level) |                               |
| 6. Vertical sync (TTL level)   |                               |

## MONITOR RECOMMENDATION

To take full advantage of the high-resolution signal MA520 produces, the video monitor should have a bandwidth of at least 15 MHz. Using a monitor with a long persistence P39 phosphor gives the best image with MA520 interlaced raster-scan signal. For lightpen operation, 7% infrared phosphor doping of the P39 is recommended.

# SCREENWARE MANAGES GRAPHICS AND TEXT DISPLAY

## GRAPHICS MODE

MA520 generates an image 512 pixels wide, 480 pixels high. Pixel coordinates run from 0-511, left to right, in the horizontal direction (X axis) and from 0-479, bottom-to-top, in the vertical (Y axis). Since it takes 9 bits to distinguish 512 ( $=2^9$ ) pixels, MA520 expects two bytes each for the X and the Y coordinates, with the high byte first. Graphics are clipped at screen edges.

MA520 automatically maintains 6 coordinate pairs: alpha cursor location (row, column), graphics cursor location (X,Y), the location of a second graphics cursor tied to the alpha cursor, tracking cross location, cross-hair location and location of the last lightpen firing.

## TEXT MODE

Graphics mode can put characters anywhere on the screen. In addition, MA520 provides a separate text cursor that writes up to 40 lines of text with 85 characters per line. The SPLITSCR command in ScreenWare can be used to limit the text cursor to the lower part of the screen. In text mode, lines are numbered from 0-39, top to bottom; characters from 0-84, left to right. The standard characters in ROM are 5 x 8 pixels in a 6 x 12 field. You can create a second character set using either the CHARACTER or MEMORY command. The ALPHA-MODE command sets figure/ground, overstrike, underline and complement options.

COMMAND NAME	MODE	NUMBER OF BYTES FROM HOST	FUNCTION
<b>GRAPHICS PRIMITIVES</b>			
<b>POINT RPOINT<sup>1</sup></b>	0	5	Turn point off
	1	5	Turn point on
	2	5	Complement point (turn off, if on; turn on, if off)
	3	5	Read point (1 byte to host)
<b>RVECTOR<sup>1</sup> RREGION<sup>1</sup></b>	0	5	Turn vector off
	1	5	Turn vector on
	2	5	Complement vector
<b>VECTOR REGION</b>	0	9	Turn rectangular region off
	1	9	Turn rectangular region on
	2	9	Complement rectangular region
<b>CIRCLE</b>	0	2	Turn circle off
	1	2	Turn circle on
	2	2	Complement circle
<b>FLOOD<sup>2</sup></b>	0	5	Flood bordered region with zeroes (can be any point inside region)
	1	5	Flood with ones
	2	5	Flood relative with zeroes
	3	5	Flood relative with ones

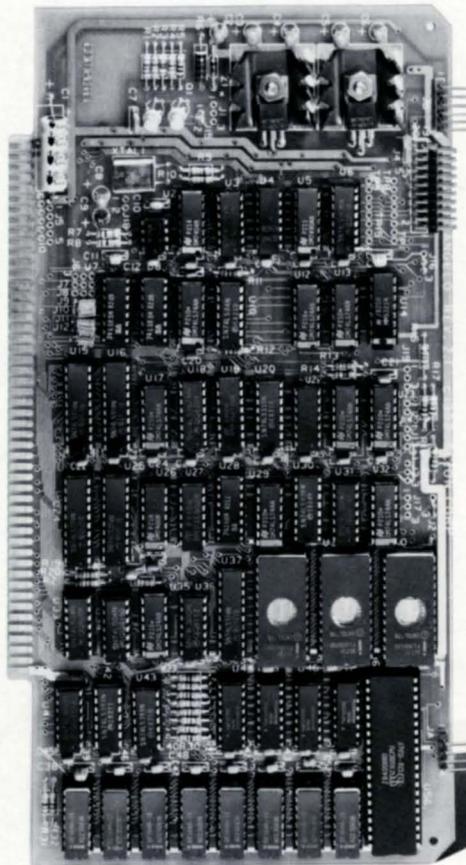
COMMAND NAME	MODE	NUMBER OF BYTES FROM HOST	FUNCTION
<b>TEXT AND CHARACTERS</b>			
<b>ALPHAMODE</b>	0	2	Establish alpha display functions
	1	3	Position alpha cursor
	2	1	Read alpha cursor (2 bytes to host)
	3	2	Set number of lines to scroll at screen end
<b>CHARACTER</b>	0	2	Write character at graphics cursor
	1	2	Set orientation, size and figure/ground of graphics character
	2 <sup>3</sup>	14	Define alternate character
	3	1	Load default character font to user memory (for modification)
<b>SPLITSCR</b>	0	2	Limit alpha screen to specified lines at lower end of screen
	1	9	Define special ASCII control codes
	2	1	Ignore or don't ignore linefeed
<b>SCREEN AND CURSOR CONTROLS</b>			
<b>GCURSOR</b>	0	5	Position graphics cursor
	1	1	Read graphics cursor (4 bytes to host)
	2	1	Tie graphics cursor to alpha cursor
	3	1	Tie graphics cursor to tracking cross
<b>RGRAPHIC</b>	0	5	Move graphics cursor relative to present position
<b>SCREEN</b>	0	1	Clear screen
	1	2	Set figure/ground
	2	1	Toggle figure/ground
	3	1	Read figure/ground (1 byte to host)
<b>LIGHTPEN</b>	0	1	Turn tracking cross off
	1	5	Turn tracking cross on
	2	1	Read tracking cross (1 or 5 bytes to host)
	3	1	Read lightpen (1 or 5 bytes to host)
<b>CROSSHAIRS</b>	0	1	Turn crosshairs off
	1	5	Draw crosshairs at specified location
	2	1	Read crosshairs (4 bytes to host)
	3	1	Draw crosshairs at graphics cursor location
<b>USER FUNCTIONS</b>			
<b>MACRO<sup>3</sup></b>	0	2	Start/Stop definition of a series of commands as a macro command (1 byte to host)
	1	2	Add next byte to macro (1 byte to host)
	2	2 or 4	Erase macro or clear facility (1 byte to host)
	3	2	Invoke macro command
<b>USER</b>	0	1	User defined
	1	1	User defined
	2	1	User defined
	3	1	User defined
<b>UTILITIES</b>			
<b>MEMORY</b>	0	1	Read out entire screen (left to right, top to bottom) (7800H to host)
	1	7801H	Load entire screen (left to right, top to bottom)
	2	5	Read out y bytes of MA520 memory (y bytes to host)
	3	3 + x	Load x bytes to MA520 memory
<b>UTILITY</b>	0	3	Set memory address of user command (see USER)
	1	6	Give user code control of MA520 CPU
	2	2 or 4	Enable or disable real-time interrupts
	3	1	Force cold start (no data saved)
<b>TEST</b>	0	2	Test EPROM by checksums (1 byte to host)
	1	1	Test RAM by storing and reading every byte value at every memory location (1 or 5 bytes to host)
	2	1	Alpha test by displaying entire default font
	3	4	Munching squares (moving display for adjusting monitor)

<sup>1</sup>RPOINT, RVECTOR and RREGION operate relative to the current position of the graphics cursor.

<sup>2</sup>FLOOD requires a region completely bordered with ones or zeros, and the interior must be all zeros or all ones.

<sup>3</sup>The user-defined character font and macros share 1.5K bytes of user memory space.

**SCION**  
if the image is important.



## SYSTEM OVERVIEW

MicroAngelo MA520 is a single board monochrome graphics computer with a Z80 microprocessor. It displays graphics with sharp, 512 x 480 pixel resolution. 30K bytes of RAM memory buffers the display and takes all the raster-scan refresh burden off the host system. The additional RAM memory you may use as you wish. ROM firmware contains ScreenWare, a graphics control package that manages the graphics and text on display and performs the hundreds of calculations needed to draw graphics elements. Text controls in ScreenWare recognize 128 ASCII text and control characters and produce an 85 character, 40 line text display. MA520 drives a TTL or composite video monitor. There is a lightpen interface on the board. S-100 and Multibus versions of MA520 are simple to integrate into any standard system.

