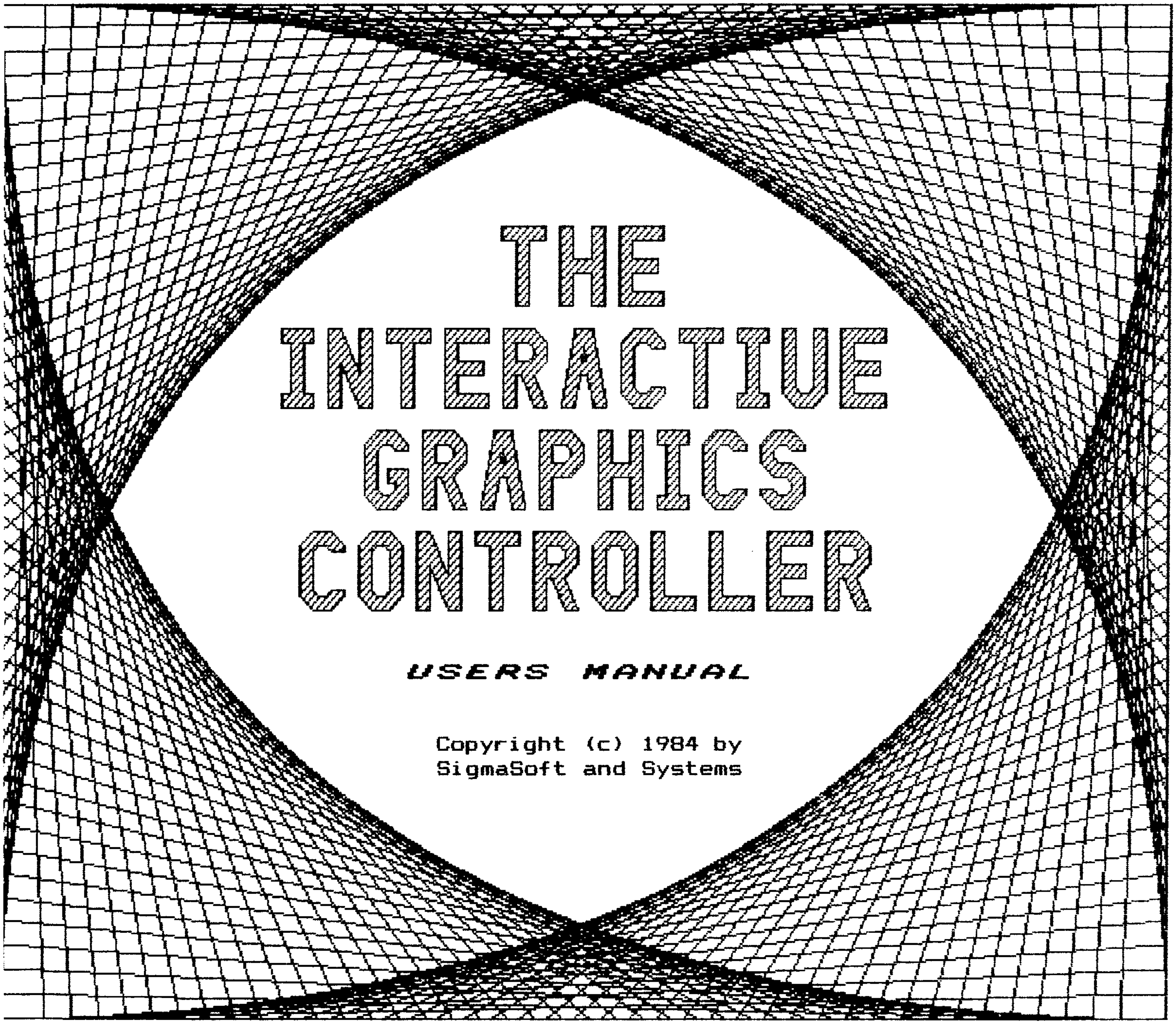


THE INTERACTIVE GRAPHICS CONTROLLER

USERS MANUAL

Table of Contents

In using Heath/Zenith H/Z89 or H/Z19 computer equipment, you are probably aware of the machines existing graphics capabilities. However, this is very different from the graphics capabilities of the same equipment once the Interactive Graphics Controller (IGC) has been installed. The existing graphics of this equipment is known as Character Graphics. Whereas, the IGC adds a far more powerful type of graphics known as High Resolution or Pixel Graphics.

Character graphics makes use of a set of graphic characters or symbols, much like the other alphabetic and numeric symbols which can be displayed on the screen. Each character is simply a different pattern of dots all based on a common 8 by 10 dot matrix.

Graphics on a standard H/Z89 or H/Z19 are composed of these same characters, by arranging them into differnt patterns. This severely limits the creativity of the programmer because of the poor amount of detail that is possible on a 80 column by 25 line screen. Moreover, the process of composing images out of this small set of characters is extremely tedious.

High resolution graphics are comprised of the same dots that are used to build characters, but there is an important difference. With the IGC device these dots or pixels are individually controllable. You are no longer limited to a predetermined set of characters, and the resolution of an image that can be drawn is greatly increased.

With Pixel Graphics the programmer no longer needs to be concerned about constructing the individual components of a line, box, or circle. Now, single commands draw entire geometric figures by simply specifying the positions for these objects.

The Interactive Graphics Controller/Pseudo Disk system greatly extends the utility of the H/Z89 and H/Z19 by incorporating the latest in semiconductor technology into these fine machines. The IGC not only provides the most powerful monochrome graphics available on any Heath/Zenith equipment, the large video memory area can also be used for many other purposes, such as a Pseudo Disk. In addition, the IGC package provides 2 Centronics Printer Interfaces, and 2 Graphics Input Device Ports for truly interactive graphics applications.

The IGC produces a high resolution video image of 640 by 250 pixels (640 by 500 in interlace mode) that can be displayed simultaneously with the existing video of the H/Z89 of H/Z19, and all of the existing functions of the H/Z89 or H/Z19 remain intact.

Three independent video memory pages exist, each capable of being selected as either the drawing page or the display page. This allows for several images to be drawn and stored at the same time for true video animation.

The Interactive Graphics Device Drivers (IG: and IGC:) are both powerful and easy to use. Images can be drawn using either absolute or relative coordinates, with a selectable origin position. Numerous styles can be selected for use with the various drawing commands, which can be performed from any language.

Single commands can be used to reposition, scale, and rotate entire images without tedious recalculations. Any page of the IGC video memory can be printed on a number of different printer types through either a parallel or serial interface.

Step #1, Getting Started

To begin the installation of the Interactive Graphics Controller position the H/Z89 or H/Z19 on a table with the cover removed. It may require a screwdriver to release the two side latches which fasten the cover closed. Turn the H/Z89 or H/Z19 so that the right side (from the front) is positioned comfortably in front of you.

Refer to Illustration A during the process of installing the IGC Power Supply Board. Please be sure to read each step completely before performing any part of that step.

Step #2, Disk Drive and Shield Removal

Remove the disk drive unit mounted in the front panel of the H/Z89 (if one is present) to provide easier access to the mounting holes for the IGC Power Supply Board installation. The drive is released by removing two screws from the top, a third screw near the bottom, and two cables that connect to the drive unit's circuit board in back. The drive unit can then be pulled out from the front of the H/Z89 cabinet.

Next, remove the drive unit shield. This sheild either partially or completely surrounded the drive unit before it was removed. The sheild may be held in place by 4 screws that fasten it to the front of the H/Z89 unit. Set the disk drive unit and shield away until needed.

Step #3, Filter Capacitor Removal

Locate the large filter capacitor which is mounted in the base of the H/Z89 or H/Z19 directly beside the power transformer. The top of this capacitor may be covered by a blue plastic insulator. The filter capacitor is mounted to the base with a capacitor clamp and has four connecting wires, colored blue and black.

Use a screwdriver to remove the two screws which fasten the capacitor clamp to the base, but do not remove the clamp or the wires from the capacitor.

Lift the capacitor (with clamp and wires attached) out over the side of the H/Z89 or H/Z19 base. Carefully, pull the filter capacitor out as far as possible to provide sufficient wire length to work with.

In some cases there may be an uninsulated ground strap connecting to the capacitor clamp and screwed to the base of the H/Z89 or H/Z19 cabinet. If it is present, disconnect it from the filter capacitor clamp.

Step #4, Filter Capacitor Remounting

Attach the filter capacitor to the IGC Power Supply Board with two #6 screws, two #6 lock washers, and two #6 nuts. Be sure that the filter capacitor is attached to the component (not foil) side of the IGC Power Supply Board and that the blue and black wires attached to the capacitor are routed away in the direction of the top of the board.

Step #5, IGC Power Supply Board Mounting

During the following step, check to be sure that no metallic objects are close enough that they might make contact with the foil side of the IGC Power Supply Board once it has been installed. There may be grounding straps and solder lugs attached to the filter capacitor clamp and/or the power transformer. These must either be removed or positioned away from the IGC Power Supply Board or they will create a severe short circuit hazard.

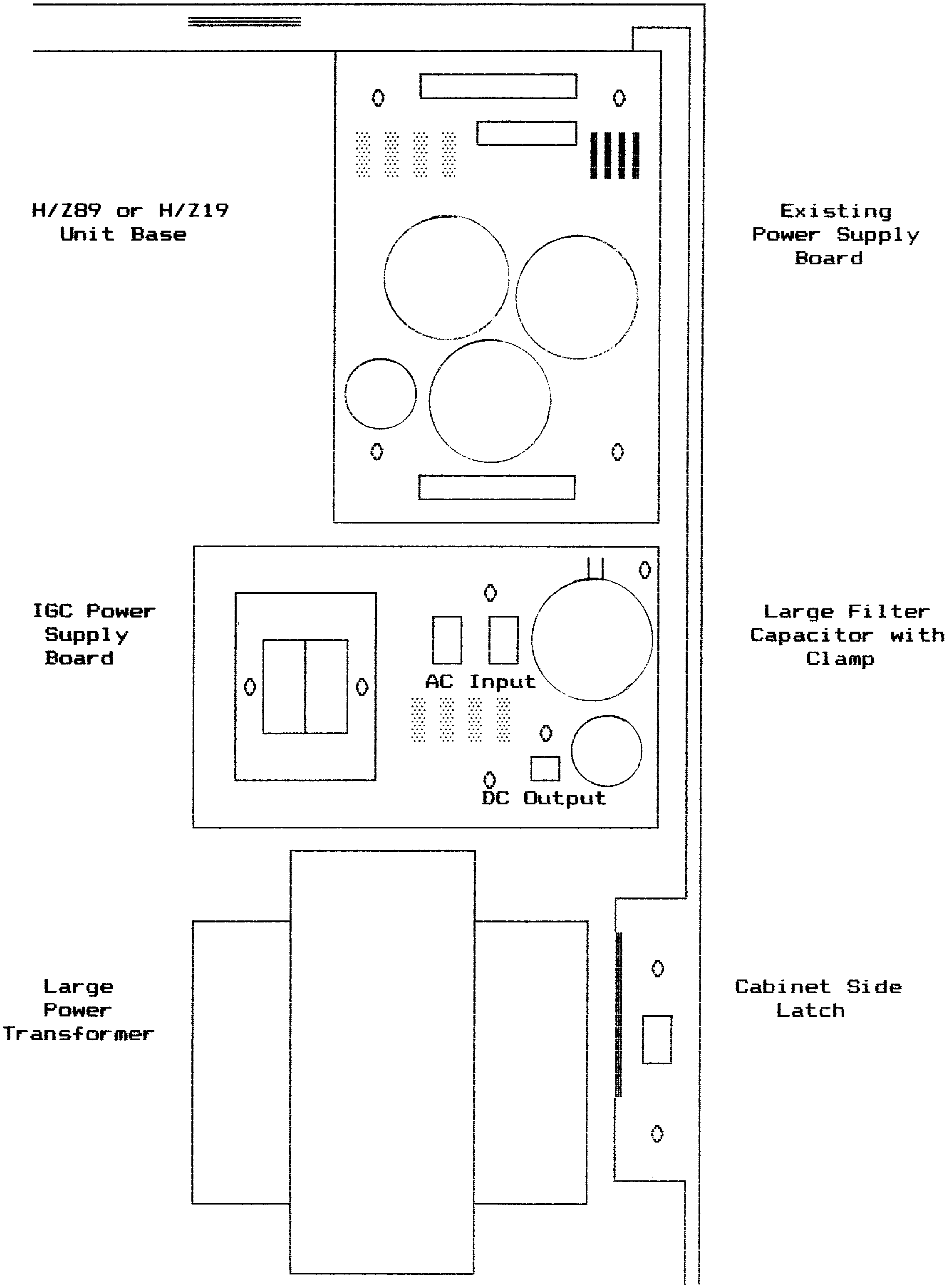
If a bare ground strap cable is present, it must be insulated. Fasten it to the base of the H/Z89 or H/Z19 unit with one of the adhesive insulating tapes provided. Position the cable so that it does not pass underneath the IGC Power Supply Board. Be sure to save the second piece of insulating tape, it will be needed later.

Slowly position the IGC Power Supply Board into the base of the H/Z89 or H/Z19 at the same position at which the filter capacitor was mounted. This step requires some care and patience, as the board must be worked through the many wires and cables in the way. The IGC Power Supply Board should be positioned so that its two mounting holes are directly above the mounting holes in the base at which the filter capacitor was attached, with the small transformer on the board toward the interior of the H/Z89 or H/Z19 unit.

Do not be afraid to pull the bulk of wires in this area up out of the way so that this board can be mounted. Feel free to cut any wire ties (DO NOT CUT ANY WIRES) needed to ease this operation. Replacement wire ties are provied for rebundling the wires after installation.

In some cases, the corner of the Video Board (the board mounted in the bottom of the H/Z89 or H/Z19 unit, under the CRT) heat sink may interfere with the mounting of the IGC Power Supply Board. If this is the case with your unit, it will be necesary to loosen the mounting screws of the Video Board so that the IGC Power Supply Board can be slipped under the corner of the Video Board heat sink. Once this is done, both boards can then be mounted firmly in place.

Illustration A



Once installed, there should be no wires beneath the IGC Power Supply Board. These wires should be routed over the top of the two large filter capacitors on this board.

Use two #6 screws and two #6 lock washers to fasten the IGC Power Supply Board to the H/Z89 or H/Z19 base using the same two mounting holes that the large filter capacitor clamp was originally mounted to.

Step #6, Power Supply Cables Installation for the H/Z89

If you are installing the IGC Power Supply Board into an H/Z19 and not an H/Z89, skip this step and proceed to Step #7.

Locate the 3 white power cables that have been provided. The longest of these cables has a 2 pin connector at each end, this is the IGC Power Supply Output Cable. The other two cables have 3 pin connectors, they are the AC Line Input Cables.

Begin by connecting either end of the Output Cable to the 2 pin output connector on the IGC Power Supply Board. The other end of this cable will be connected to the Graphics Controller, later.

WARNING! Take extreme care when connecting cables to the IGC Power Supply Board. Never join two connectors of an unequal number of pins. Never join two connectors unless they fit together easily without force. The single 2 pin connector on this board is the low voltage output, it must never be confused with the two 3 pin AC line connectors (high voltage).

Each of the 3 connectors on the IGC Power Supply Board has a locking ramp for polarization. Each time a cable is connected to one of these connectors be absolutely certain that the ramp on the connector of the cable is positioned toward the ramp on the connector of the circuit board that it joins with. The slots in the connector on the cable's end must always be positioned away from the ramp on the circuit board's connector.

Next, locate the AC line cable that connected to the fan before the H/Z89 cover was removed. This cable comes through an opening in the unit's base under the existing power supply board. Join the free end of this cable with the short (about 3 inches) AC Line Cable. Only the end of this cable without the locking ramp can be connected to the fan cable.

Now, connect the other end of the short AC Line Cable (the end with the locking ramp) to one of the two 3 pin AC Line Input connectors on the IGC Power Supply Board. These two 3 pin AC Line Input Connectors are interchangeable.

Connect the long AC Line Input Cable to the other 3 pin AC Line Input Connector on the IGC Power Supply Board, just as the short cable is connected. The free end of this long cable will now be your fan cable.

Step #7, Power Supply Cables Installation for the H/Z19

If you are installing the IGC Power Supply Board into an H/Z89 and not an H/Z19, skip this step and proceed to step #8.

Locate the 2 white power cables that have been provided. The longest of these cables has a 2 pin connector at each end, this is the IGC Power Supply Output Cable. The other cable has a single 3 pin connector at one end, this is the AC Line Input Cable.

Begin by connecting either end of the Output Cable to the 2 pin output connector on the IGC Power Supply Board. The other end of this cable will be connected to the Graphics Controller, later.

WARNING! Take extreme care when connecting cables to the IGC Power Supply Board. Never join two connectors of an unequal number of pins. Never join two connectors unless they fit together easily without force. The single 2 pin connector on this board is the low voltage output, it must never be confused with the two 3 pin AC line connectors (high voltage).

Each of the 3 connectors on the IGC Power Supply Board has a locking ramp for polarization. Each time a cable is connected to one of these connectors be absolutely certain that the ramp on the connector of the cable is positioned toward the ramp on the connector of the circuit board that it joins with. The slots in the connector on the cable's end must always be positioned away from the ramp on the circuit board's connector.

Remove the two screws that hold the chassis cover in place over the AC section of the H/Z19 power supply. This is the metal piece in which the unit's power switch and fuse are mounted. Gently pull the chassis cover outward to provide access to the solder lugs of the 115V/230V Switch (SW1). This switch is mounted on the bottom of the H/Z19 unit. Do not confuse it with the main power switch (SW3) or the low voltage switch (SW2).

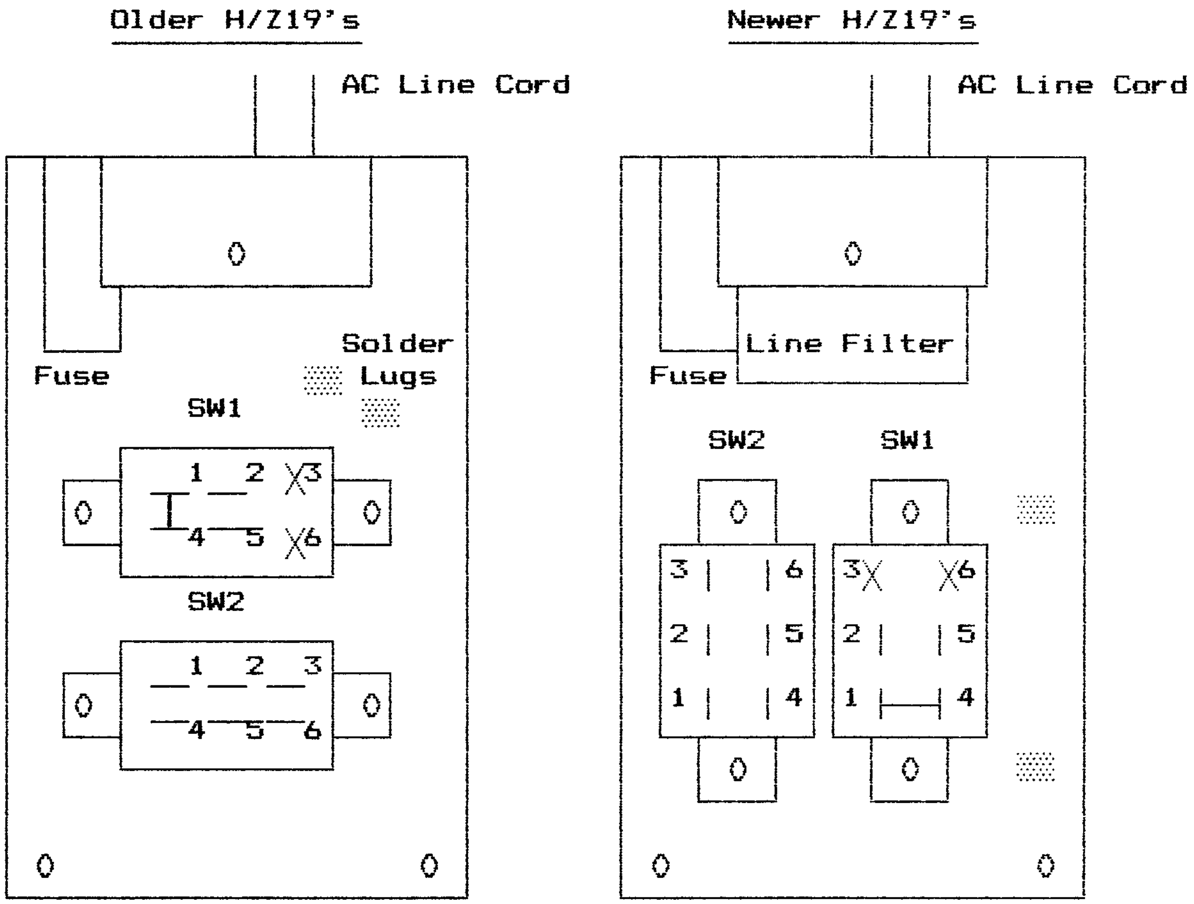
Connect the AC Line Input Cable to either of the 3 pin connectors on the IGC Power Supply Board. Route the bare end of this cable through the opening in the H/Z19 base into the AC power section. This opening is located underneath the H/Z19 Power Supply Board.

Note that in the following step the AC Line Input Cable will be soldered onto 2 of the solder lugs on switch SW1. In some cases this switch may be positioned horizontally above SW2, or it may be mounted vertically beside SW2. Refer to Illustration B to identify SW1 and the 2 solder lugs at which the AC Line Input Cable is to be attached. The correct solder lugs have been indicated with two X symbols in the illustration.

Solder either of the two bare ends of the AC Line Input cable onto lug number 3 on SW1. Solder the other bare conductor of this same cable onto lug number 6 of SW1.

Carefully inspect the solder connections you have just made to be sure that they are correct, and that all of the solder lugs are well insulated from one another. Then, reinstall the chassis cover of the AC section of the H/Z19 power supply.

Illustration B



Step #8, Power Supply Installation Tests

Reinstall the H/Z89 or H/Z19 cover so that the IGC Power Supply Board can be tested. Join the long AC Line Input Cable with the fan cable in the unit's cover, if present. Once again, inspect the AC lines and be sure their connection to the IGC Power Supply Board is correct. Then, close the cover of the H/Z89 or H/Z19 and plug in the unit's AC line cord.

Turn on the power switch to the H/Z89 or H/Z19. The unit should begin operating as usual. If not, turn the power switch off and check the unit's fuse. A blown fuse is an indication of an installation error.

If you have a voltmeter, test the output of the IGC Power Supply Board at the free end of the 2 conductor Output Cable for about 13 Volts DC. This measurement may deviate by as much as 3 Volts maximum. Do not proceed to install the Interactive Graphics Controller or the Universal Parallel Interface Board unless the IGC Power Supply Board operates properly.

Step #1, Interface Board Installation for an H8

If you are installing the Universal Parallel Interface Board into an H/Z89 and not an H8 computer, skip this step and proceed to Step #2.

The installation of the H8 version of the Interface Board is very straightforward. Simply connect the board to any available bus slot inside the H8 mainframe. Refer to the Configuration section for information on setting the various jumpers on this board. If you have any special accessory cables to install, such as a Centronics printer cable, proceed to the accessories installation section. Otherwise, proceed directly to the Graphics Controller installation section.

Step #2, CPU Logic Board Removal

Start by removing the CPU Logic Board from the H/Z89 unit. The CPU Logic Board is the large board mounted vertically directly behind the CRT of the H/Z89. The smaller boards which are attached to the CPU Logic Board may remain attached.

Several cables must be detached from the CPU Logic Board and the smaller boards to facilitate their removal. Be sure to take note of how these cables are connected so that you can reconnect them properly once this installation proceedure is complete.

Step #3, Mounting Hardware Removal

In some cases there may be a Support Bracket that connects the CPU Logic Board to the front of the H/Z89 cabinet, just above the CRT. Remove this bracket if it is present, it will no longer be needed once the Interactive Graphics Controller has been installed.

Locate the Accessory Mounting Brackets that connect to the top of the CPU Logic Board to support the smaller accessory boards. There may be just one located on the right side, or a second one on the left side. Remove any aluminum or rubber spacers that may exist on the tops of these mounting brackets, these spacers will no longer be needed.

If an Accessory Mounting Bracket is present on the left side of the CPU Logic Board, remove it. This bracket will be reinstalled later. Do not remove the mounting bracket on the right side.

Step #4, Interface Board Installation

Plug the Universal Parallel Interface Board onto one of the 3 bus slots on the left side of the H/Z89 CPU Logic Board. Any of the 3 left side slots can be used. They are numbered P5Ø1-P5Ø7, P5Ø2-P5Ø8, and P5Ø3-P5Ø9. This board can not be connected to any of the 3 bus slots on the right side of the CPU Logic Board.

Reinstall the left Accessory Mounting Bracket to the CPU Logic Board to support the Interface Board.

Step #5, CPU Control Cable Installation

Locate and remove the Integrated Circuit U553 from its socket on the CPU Logic Board. It's the first IC to the left of the DIP switch (SW5Ø1), near the bottom right corner of the CPU Logic Board. Check to be sure that the IC chip you have removed has 14 pins and bears the part number 74LS32. The part number may not match exactly, but it should be close.

Locate the CPU Control Cable. This cable has a single 8 pin connector at one end and a black 14 pin DIP plug at the other. Note that one corner of the DIP plug is notched to indicate pin 1.

Plug the black DIP plug of the CPU Control Cable into the U553 socket. Be sure to position pin 1 of the DIP plug so that it connects to pin 1 of the U553 socket. Pin 1 should be labeled on the CPU Logic Board beside the U553 socket. Once connected, the small wire attached to the DIP Plug should exit from the top. The IC chip removed from U553 will no longer be needed.

Connect the free end of the CPU Control Cable to the 8 pin connector at the top right corner of the Interface Board. This connector must be positioned so that the 3 conductor cable enters the 8 pin connector shell on the right side. The 8 slots cut in the body of the connector shell must be positioned up and not down.

This completes the installation of the Universal Parallel Interface Board. Before reinstalling the CPU Logic Board, you may proceed to install the Interactive Graphics Controller. However, if you have any special accessory cables to install, such as a Centronics Printer Cable proceed to the accessories installation section first, and install these cables before you begin the installation of the Graphics Controller.

For the following steps refer to Illustration C. Note that for these steps you may wish to remove the Terminal Logic Board from the H/Z89 or H/Z19 unit to aid the installation process, but this is not required. The Terminal Logic Board is the second large board mounted vertically behind the CPU Logic Board, which should have already been removed.

Note that in the following steps references will be made to the "CPU Logic Board" eventhough this board may not be present if you are installing the Graphics Controller into an H/Z19 unit. Even if the CPU Logic Board is not present, it is still important to keep in mind that it would mount just in front of the Terminal Logic Board.

Step #1, Character ROM installation

Begin by locating the character generator ROM on the Terminal Logic Board and then remove it. This is a large 24 pin Integrated Circuit positioned in the top left corner of the Board. The character ROM is labeled either as U473 or U420 on the circuit board.

Plug the character ROM you have just removed into the empty character ROM socket on the Interactive Graphics Controller. This socket is located immediately beside the Terminal Interface Connector which is labeled as "Terminal" on this board.

Note that this ROM must not be installed onto the IGC if you have purchased an Alternate Character Set ROM. Such a part replaces the ROM from the Terminal Logic Board completely. Refer to the Accessories Installation section for details.

Be certain to position pin 1 of the Character ROM toward the top (the direction of the Power Connector) of the IGC board, just as the other chips on this board are positioned.

Step #2, Terminal Interface Cable Installation

Plug the 24 pin DIP plug of the Terminal Interface Cable into the now vacant IC socket at either U473 or U420 on the Terminal Logic Board so that the cable exits from the top of the DIP plug and extends out over the Terminal Logic Board's heat sink.

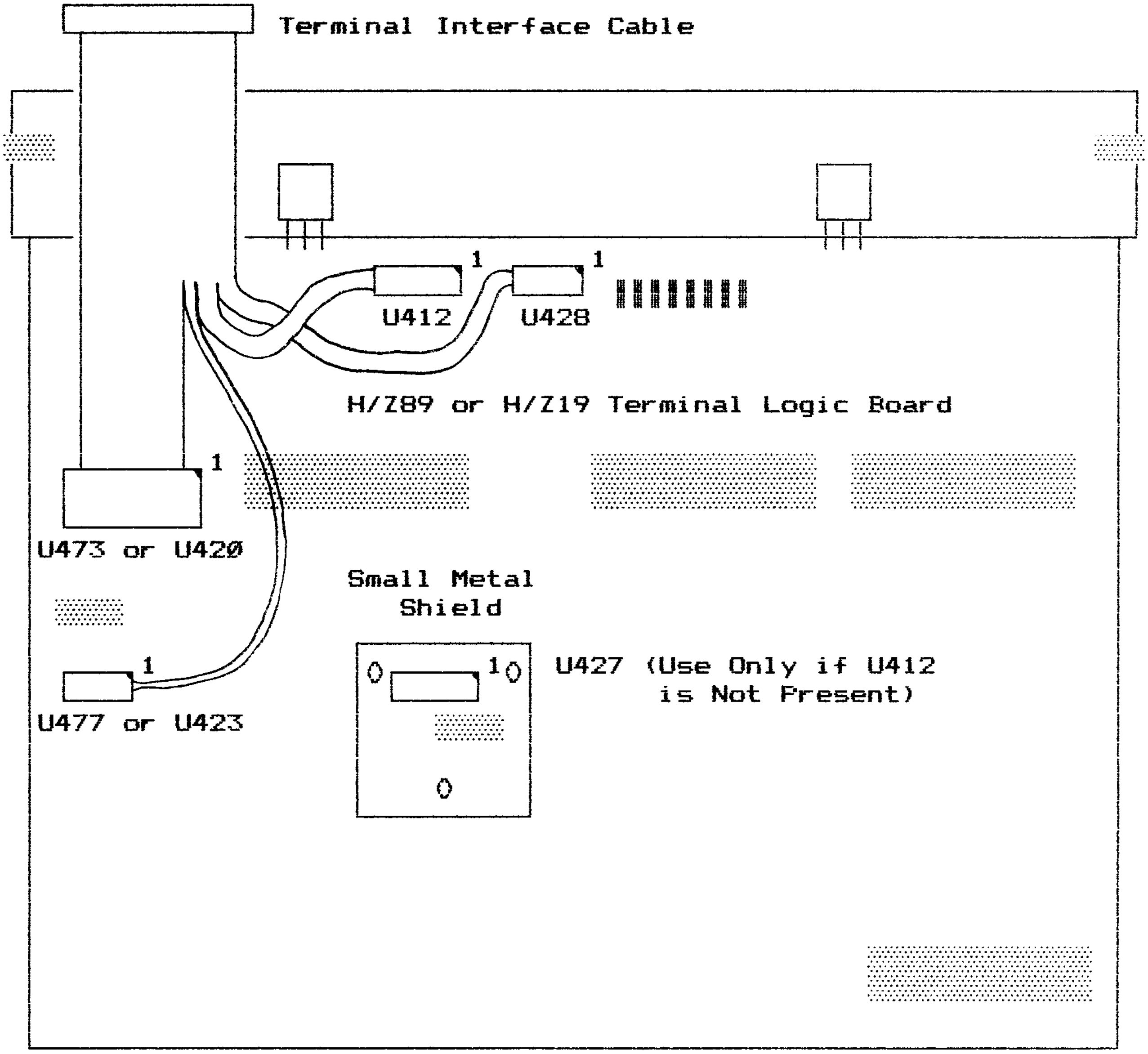
The Terminal Interface Cable is about 10 inches in length and has 4 DIP plug connectors at one end and a single 34 pin socket connector at the other. This cable is for connection between the Terminal Logic Board and the Interactive Graphics Controller.

Locate and remove a 14 pin IC chip at either U477 or U423 on the Terminal Logic Board. This IC chip is the second one below the Character ROM socket on this board. Check to be sure that the part you remove bears the part number 74LS08. This IC chip will no longer be needed.

Now, plug the 14 pin DIP Plug of the Terminal Interface Cable that has a 2 conductor wire into this empty socket. This is the same DIP plug that connects to the Terminal Interface Cable closest to the 24 pin DIP plug.

Note that one corner of the DIP plug is notched to indicate pin 1. Be sure to position pin 1 of the DIP plug so that it connects to pin 1 of the IC socket. Pin 1 should be labeled on the Terminal Logic Board beside the IC socket. Once connected, the small wire attached to the DIP Plug should exit to the right.

Illustration C



Next, locate and remove a 14 pin IC chip at U428 on the Terminal Logic Board. This IC chip is located at the very top of this board. Check to be sure that the part you remove bears the part number 74LS74. This IC chip will no longer be needed.

Plug the 14 pin DIP Plug of the Terminal Interface Cable that has a 4 conductor wire into this empty socket. Again, be sure to position the notched corner of the DIP plug with the pin 1 designation on the circuit board. The attached wire should exit to the left of this DIP plug.

Finally, locate and remove a 16 pin IC chip at either U412 or U427 on the Terminal Logic Board. This device may be located in one of two different places.

With older model H/Z89 and H/Z19s, U412 is located to the immediate left of U428 which is the socket that you just installed a DIP plug into. If this is the case with your Terminal Logic Board, then use this socket, U412.

If U412 is not present beside U428 you must use U427 instead, which is located beneath the small metal shield in the center of the Terminal Logic Board, on the component side. Note that there may be a U412 located elsewhere on the board, but it can not be used.

The shield is attached by 3 screws. If necesary, remove the shield and set it aside. It will no longer be needed.

The part removed from either U412 or U427 must have 16 pins and bear the part number 74LS161. This IC chip will no longer be needed.

Plug the 16 pin DIP Plug of the Terminal Interface Cable that has a 4 conductor wire into the empty socket at either U412 or U427. Again, be sure to position the notched corner of the DIP plug with the pin 1 designation on the circuit board. The attached wire should exit to the left of this DIP plug.

Step #3, CPU Interface Cable Installation

Locate the CPU Interface Cable and connect it to the Universal Parallel Interface Board which has already been installed. Some models of the interface board have two 34 pin header connectors in which case the lower one must be used.

For an H/Z89 unit, this cable is about 12 inches in length and has a 34 pin socket connector at each end. This cable must be connected so that it extends into the interior of the H/Z89, and not in the direction of the left side of the cabinet.

With an H/Z19-H8 system, the CPU interface cable is about 6 feet in length and has a 34 pin socket connector at each end. This cable must be routed out of the back of the H8 mainframe and into the back of the H/Z19 cabinet. Bring the cable up along the back of the Terminal Logic Board to the Graphics Controller.

Step #4, CPU Logic Board Reinstallation

Reinstall the CPU Logic Board if you removed it earlier. Also, reinstall the Terminal Logic Board if it was removed. During this process, take great care that the DIP plugs installed onto the Terminal Logic Board earlier are not knocked loose. Also, check to be sure that all of the cables that originally connected to these two boards are properly reconnected.

The free end of the Terminal Interface Cable must extend upward between the CPU (if present) and Terminal Logic Boards for connection to the IGC.

Do not reinsert the 2 screws that fasten the CPU Logic board into place. The Graphics Controller Mounting Brackets will be connected here.

Step #5, Graphics Controller Installation

Press the remaining strip of adhesive insulating tape that was provided into place on top of the CRT. This insulator must cover the center portion of the top of the metal band that extends around the glass CRT tube.

Connect the free end of the CPU Interface Cable just installed, to the CPU Interface Connector on the Interactive Graphics Controller. This is a 34 pin header connector and is labeled "CPU" on the circuit board. For an H/Z89 this cable should be connected so that the ribbon cable exits from the bottom of the 34 pin connector socket and be routed underneath the Graphics Controller. With an H/Z19 the ribbon cable should exit to the top and go down the back of the Terminal Logic Board.

Connect the free end of the Terminal Interface Cable to the Terminal Interface Connector on the Graphics Controller This is a 34 pin header connector and is labeled "Terminal" on the circuit board.

Lower the Graphics Controller down into place above the CRT with the CPU and Terminal Interface Connectors positioned in front of the CPU and/or Terminal Logic Boards. The components on the IGC should be on the top side with the left and right IGC Mounting Brackets extending around the sides of the CPU Logic Board position.

Step #6, Final Installation

Check to be sure that the bottom of the IGC is completely insulated from all metallic objects underneath the board. The IGC should be spaced about 1/2 of an inch above the Left and Right Accessory Mounting Brackets (if present), and the plastic insulator installed earlier should protect it from contacting the metal band around the CRT.

Locate the 2 small aluminum shoulder spacers that have been provided. Note that you may have received some spares, but only 2 will be needed. Push one of the spacers into the hole provided in the Left IGC Mounting Bracket so that the spacer extends inward (toward the IGC circuit board). In the same manner, push the other spacer into the hole of the Right IGC Mounting Bracket so that it extends inward. You may need to use pliers to squeeze these spacers into place.

Simultaneously, hold both of the spacers in place and gently pull the Left and Right IGC Mounting Brackets apart so that they can be pushed around the Vertical Mounting Brackets that support the CPU and/or Terminal Logic Boards. Align the spacers in the IGC Mounting Brackets with the 2 screw holes in the Vertical Mounting Brackets that were originally intended to mount the CPU Logic Board.

Fasten the IGC into place with two 5/8 inch #6 screws by inserting them through the IGC Mounting Bracket Holes, through the shoulder spacers, and into the bushings of the CPU Logic Board. If you do not have the CPU Logic Board (H/Z19), also use two #6 lock washers and two #6 nuts to complete this mounting.

Connect the free end of the IGC Power Supply Output Cable to the Graphics Controller at the 2 pin Power Connector.

WARNING! Take extreme care that this cable is connected properly. The locking ramp of the connector on the free end of the cable must join with the locking ramp of the Power Connector on the IGC. The 2 slots in the connector at the end of the cable must face downward against the circuit board.

This completes the installation of the Interactive Graphics Controller.

Step #7, Initial Tests

Connect the AC Line Cord to the H/Z89 or H/Z19 and turn the power switch on. The unit should operate as usual. If not, power the unit down immediately and re-check your installation. Also, refer to the Trouble Shooting Guide for assistance.

If the unit does operate properly, boot a system disk and run the IGCTEST utility provided on the IGC Software Distribution Diskettes.

Centronics Printer Cable

The Universal Parallel Interface Board features 2 Centronics Parallel Ports, each of which can be used to drive a separate printer provided that they have a 36 pin Centronics type input.

To install a Centronics cable for use with the Universal Parallel Interface Board, the Interactive Graphics Controller (if present) must be hinged up into the vertical position.

The two piece Centronics cable features an internal section, for connection to the H/Z89 backplate, and a longer external section for connection to the printer.

Begin by connecting the 10 and 12 pin connectors at one end of the internal cable to one set of the 10 and 12 pin connectors on the Interface Board. The 10 and 12 pin connector set closest to the CPU Logic Board should be used first, if they are not already being used.

Route the free end of the internal cable down and under the CPU and Terminal Logic Boards and to the backplate. Use the supplied mounting hardware to fasten the free end of the internal cable to the inside of the backplate. Any available DB-25 mounting hole can be used.

The external portion of the cable can then simply be connected between the outside of the backplate and the Centronics interface of the printer.

Atari Adapter Cable

The Atari Adapter Cable can be connected to either of the 2 Input Device Interfaces on the Interactive Graphics Controller. The slotted side of the 12 pin connector should be positioned downwards when connected. The free end of the cable can then be connected to any standard digital Atari or Wico trackball or joystick.

The standard Heath/Zenith backplate does not provide a DB-9 mounting hole, so the cable will have to be simply routed out of an available opening. Note that replacement backplates are available which do provide DB-9 mounting holes.

Alternate Character Set ROM

To install an Alternate Character Set Generation ROM, simply remove any ROM that may already be present on the Interactive Graphics Controller and replace it with the new ROM. The character ROM sockets are the 2 large 24 pin IC sockets next to the Terminal Interface Connector in the bottom left corner of the board.

The socket closest to the Terminal Interface is only for use with 2k ROMs that contain a single character set, such as the standard Heath character ROM. The other 24 pin IC socket (further from the Terminal Interface Connector) should be used for 4k (2732) ROMs that contain 2 character sets, such as an alternate character set ROM. Both sockets should never be used simultaneously.

Be certain to position pin 1 of the Character ROM toward the top (the direction of the Power Connector) of the IGC board, just as the other chips on this board are positioned.

128k Memory Expansion

The IGC features the capability to use 2 sets of 64k Dynamic RAMs for a total of 128k bytes of memory by a process known as piggybacking. 128k bytes of piggybacked memory is normally provided by SigmaSoft installed as an option. This information is provided only for those who have a good knowledge of such memory devices. SigmaSoft recommends that this option be factory installed for testing reasons.

For a total of 128k of memory, 8 sets of 2 4164-150ns DRAMs must be soldered together pin for pin with the exception of pin 15 (CAS). Pin 15 on each of the devices on the top row should be bent up and connected with a jumper to the individual holes provided beside each memory socket.

A 74LS00 must be installed beside jumper J5 and a 74LS153 must be present beside jumper J1. The value of the 33pf silver mica capacitor located beside the 74LS14 device should be reduced to 10pf. Also, the jumper J5 must be placed in the piggyback position. Refer to the Configuration section.

256k Memory Expansion Kit Installation

The 256k Memory Expansion Kit contains eight 256k memory chips, a 74LS14, a 74LS153, and a silver mica capacitor. Begin the installation of this kit by removing the Interactive Graphics Controller from the H/Z89 or H/Z19 unit by unpluging it's three cables and unscrewing the two #6 mouting screws that fasten it to the CPU board's mounting brackets. The graphics board has two long steel brackets that are fastened to each side and should be left intact.

Place the graphics board on a work bench and position it so that these steel brackets are horizontal and all of the interface connectors that run along one side are to your left. At the far side of the board there should now be a large black heat sink and the board label which reads "The Interactive Graphics Controller". This is considered to be the top of the board.

Locate the eight 16 pin memory chip sockets which run along the bottom of the board (edge nearest you). These sockets will either contain eight 64k memory chips (64k boards) or sixteen 64k chips piggybacked together (128k boards). 64k memory chips will have a part number similar to 4164-15 or 4564-15 etc.

Remove all of the memory chips from these eight sockets. If you have a 128k board, you must first remove the small jumper wire that connects each pair of memory chips to the printed circuit board. The best way to do this is to push the wire off of the bent up pin with the tip of a hot soldering iron, then remove the chips and unsolder the jumper wires from the board last.

Both the 64k memory chips you are removing and the 256k memory chips you will be installing are static sensistive devices. These devices are protected from damage by static when they are plugged into the board but must be placed in conductive foam or aluminum foil when they are out of the circuit board.

Install the eight 256k memory chips provided into the emptied memory sockets. These components will have a part number similar to 41256-15. As you install each of these parts, carefully bend the pins against a hard metal surface to adjust the pin spacing to plug into the sockets easily. Each chip has a notch or dot at one end which must point towards the top of the board.

Locate jumper J5 in the lower right corner of the graphics controller just above the memory chip sockets. It is labeled on the printed circuit board foil. Remove the chip with a part number of 74LSØØ from the 14 pin socket to the immediate left of this jumper. This socket will already be empty if you have a 64k board. This part will no longer be needed.

Remove the plug jumper from J5 and reposition it upward if it is not already in that position. Refer to the Configuration section.

Locate jumper J1 in the upper left corner of the graphics controller. It is labeled on the printed circuit board foil. The 16 pin socket to the immediate left of this jumper must have a chip installed with the part number 74LS153. If you have a 64k board this socket will be empty and you should install the 74LS153 provided. Be sure to position the notched or dotted end of this chip towards the top of the board.

Locate the 74LS14 near the center of the board. There are two brownish red colored capacitors located to the immediate right of this component. Replace this 74LS14 with the new one provided. Be carefull not to confuse the old and new parts as they both have the same part number. Remember to position the notched or dotted end of the chip towards the top of the board.

The final step of the 256k memory kit installation is to replace one of the brownish red colored capacitors located beside the 74LS14 with the new one provided. As summarized in the chart below, the 1st of these capacitors (the one closest to the 74LS14) should be labeled with the number 680 (below the CM05 number). The value on the 2nd capacitor (on the right) will either be 330 or 100 depending on the size of the installed memory chips. This is the capacitor which must be replaced.

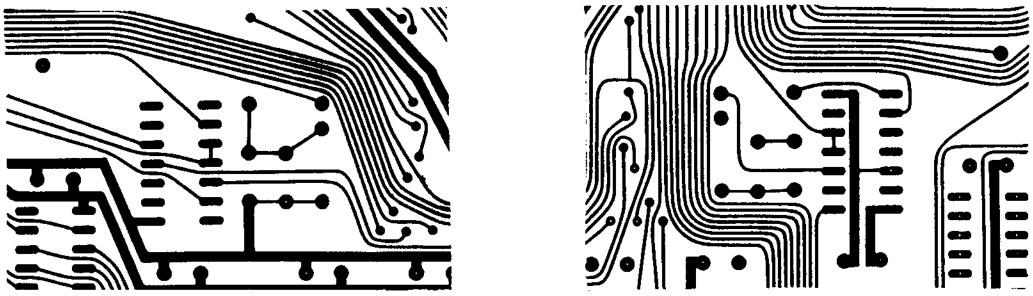
IGC Board Timing Capacitor Values

	64K Boards	128k Boards	256k Boards
1st Capacitor	680 (68pf)	680 (68pf)	680 (68pf)
2nd Capacitor	330 (33pf)	100 (10pf)	101 (100pf)

Take extreme care when removing this capacitor. The leads of this component fit very tightly into the holes of the printed circuit board and the foil may be damaged when the part is removed.

Use the enclosed copper desoldering wick to remove the solder from the holes so that the new capacitor can be inserted. If a foil is damaged refer to the pictorial below to repair it using the provided Kynar wire.

Printed Circuit Board Foil Around the 74LS14 Device



Component side

Solder Side

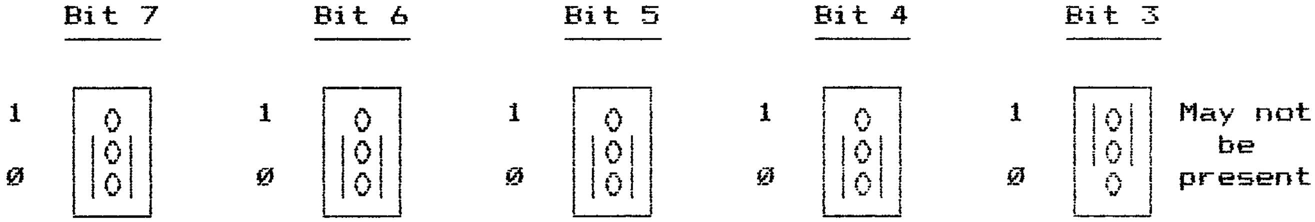
This completes the installation of the 256k memory expansion kit. Reinstall the graphics controller and run the IGCTEST utility for all four memory banks to test the new memory chips. If the test utility reports any errors, carefully re-inspect your work.

Port Address Selection Jumpers

The set of jumpers in the lower right corner of the Universal Parallel Interface Board are the Port Address Selection Jumpers. Some models of the interface board have four jumpers here and some have five (with a sixth jumper to the far left). The right most jumper (Bit 3) is not present on the boards with just four jumpers and should be ignored in the diagram below.

The positions of these jumpers determine the port address zone of the Universal Parallel I/O Board and all devices interfaced through it, including the Interactive Graphics Controller. The standard port address of 8 decimal should be selected as shown below.

Port Address Selection Jumpers



Each of the five jumpers correspond to an address bit (A3 through A7), which together form a binary number which gives the first port address of the selected eight port address zone. The first three bits from the right (AØ through A2) determine the local ports (one of eight) and so are considered to be zero when calculating the zone address. Remember that the calculated port address is only the first address of eight.

Unless your needs require a special port address, the port address setting illustrated above is recommended. If you must change the port address setting there are two important guidelines you must follow. The Universal Parallel Board must not be configured to operate at a port address that is currently being used by any other device connected to the H/Z89 computer. Also, there may be software changes that have to be made when the port address setting is modified.

The Enable Jumper

The Enable Jumper is located to the immediate right of the large 24 pin IC on the Universal Parallel Interface Board. This jumper must be in the on position for the interface to operate. When the Enable Jumper is placed in the off position, the port decoding circuitry of the interface is disabled. This effectively disconnects all devices from the H/Z89 computer that are interfaced through this board. For operation of the IGC system, be sure that this jumper is in the upward position as shown. This jumper is not present on some models.

The IGC contains five jumpers which allow the user some flexibility in configuring the operation of the device for his specific needs. However, these jumpers can interfere with the operation of the IGC if not set properly. Refer to the chart below for a summary of the functions of these jumpers. The positions of the jumpers shown in this chart are the most common settings.

Interlaced Pixel Video (J1 and J4)

Interlaced video is a mode that doubles the pixel resolution of the video image displayed on the CRT screen of the H/Z89 or H/Z19. The trade-off in using interlaced video is that the refresh rate of the screen is reduced, and this can result in a noticable flicker in the image. The severity of the flicker depends on the type of CRT screen installed in your machine. Amber CRTs are the best suited for interlaced video. If your unit has a white or green CRT screen you may not wish to use interlaced video. However, some people find the flicker less objectionable than others.

In order to use interlaced video, you must have installed a Super19, or Ultra ROM onto your Terminal Logic Board. Also, the IG: or IGC: device driver must be set for the proper interlace mode, and for the particular terminal code ROM (Super19/Ultra ROM) that you have installed.

Two different modes of interlaced video are supported by the IGC, page interlace, and bank interlace. If your IGC has only 64k bytes of memory, you can not use bank interlace mode. However, 64k byte systems can use page interlace mode instead. 128k and 256k byte systems should use bank interlace mode only. Both page and bank interlace modes must never be selected at the same time.

Interlaced Character Video (J2)

Character interlaced video increases the matrix resolution of the H/Z19 or H/Z89's entire character set from 8 x 1Ø to 8 x 2Ø. This mode can not be used without a special character ROM installed, that has been designed to produce these characters.

5ØHz/6ØHz Refresh Rate (J3)

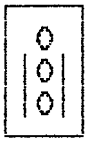
The 5ØHz frame rate setting should only be selected if your AC line frequency is 5Ø Hertz (European countries). Otherwise, 6ØHz frame rate should be set. Also, the setting of this jumper must correspond to the 5ØHz/6ØHz selection DIP switch on the Terminal Logic Board. The combination use of 5ØHz and interlaced video is not recommended due to the increased flicker caused by the reduced frame rate.

Piggybacked Memory (J5)

The 128k byte piggyback jumper allows two sets of the less expensive 64k dynamic memories to be used as an alternative to the 256k dynamic memories. This jumper should only be set to the 128k piggyback position if this option has been installed.

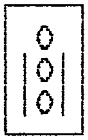
IGC Configuration Jumper Settings

J1



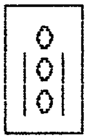
Up	Bank Interlaced Pixel Video On
Down	Bank Interlaced Pixel Video Off

J2



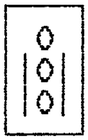
Up	Interlaced Character Video On
Down	Interlaced Character Video Off

J3



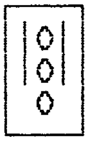
Up	50Hz Frame Rate for European Operation
Down	60Hz Frame Rate for American Operation

J4



Up	Page Interlaced Pixel Video On
Down	Page Interlaced Pixel Video Off

J5



Up	64k or 256k Bytes Memory
Down	128k Bytes Piggybacked Memory

IG: and IGC:

The Interactive Graphics Device Driver

HDOS and CP/M Versions 2.0

Written by Darron J. Shaffer
and Clay D. Montgomery

SigmaSoft and Systems
4488 Spring Valley, #107
Dallas, TX 75234

Introduction

The Interactive Graphics device driver is a high level graphics command processor designed especially for use with the Interactive Graphics Controller. IG: and IGC: (the HDOS and CP/M versions respectively) are actually a complete programming language that combines power and ease of use to a unique degree.

These device drivers are used to create graphics images on the IGC. They were written in Assembly Language to provide very fast drawing speeds, even to applications written in slower interpretive or compiled languages.

Getting Started

The device drivers are provided on the IGC software distribution diskettes. One of these disks contains the HDOS version and another contains the CP/M version of the same software. Begin by copying the contents of the one you wish to use onto a system disk. Be sure to store these distribution diskettes in a safe place once they have been copied onto working system disks.

The next step is configuration. The graphics driver features a number of set options for flexibility. These set options must be configured properly before use.

Performing a Set Operation

The Set utilities (Set.ABS under HDOS and Set.COM under CP/M) allow the user to configure various options available for a device driver. The Set utilities modify the driver by making patches in the code of the program and then writting the patched driver back to the device driver disk file, so that the change need not be made again. An example of the syntax under HDOS is:

```
>Set IG: Help
```

Under CP/M type:

```
A>Set IGC: Help
```

The values specified for a set operation may be given in several number bases by supplying the proper radix where decimal is the default. Each time a set operation is performed the set status table for all of the options will be displayed to verify that the desired change has been made.

The Help Option

The Help option will display the set status of all of the options to the system console without making any set changes to the driver. All of the values displayed will be in decimal, regardless of the number base they were originally set in.

The Inter Option

The Inter set option defines the video interlace mode for all of the drawing commands of the device driver. Only one of three different values may be specified. A value of Ø for non-interlace, 1 for page interlace, and 2 for bank interlace.

The interlace mode set here must correspond to the jumper settings on the IGC mainboard and to the terminal initialization sequence defined with the Term set option described below. Note that either a Super19 or Ultra ROM is required to use an interlace mode.

The Printer Option

The Printer set option defines the printer type that is to be used by the Print command of the device driver. Refer to the chart below to determine the proper value for your printer type.

Summary of Printer Type Values

Value	Printer Type
Ø	Epson/Star Gemini/Panasonic/Riteman/Mannesman Tally
1	Integral Data Systems/Paper Tiger/Micro Prism
2	C. ITOH/Prowriter
3	Micro Peripherals Inc./99G/Bell and Howell/S Printer
4	Okidata/Microline/Okigraph

The Port Option

The Port set option defines the I/O port address of the Interactive Graphics Controller. The value set here must correspond to the port address selection jumpers on the interface board. The standard setting is 8.

The Term Option

The Term set option determines the 8 byte terminal initialization sequence which is output to the H/Z89 or H/Z19 console device when the device driver is loaded. The values which should be set here depend on the desired interlace mode (refer to the Inter set option) as well as the type of terminal code ROM installed on your Terminal Logic Board. Below is a chart which lists the values needed for the various ROMs available. Unused values should be set to zero.

Values Required for a Term Set Option

Terminal Code ROM	Non-Interlace	Interlace (Page or Bank)
Standard Heath ROM	27,12Ø,49	n/a
Watzman ROM	27,12Ø,49	n/a
Super19 ROM	27,12Ø,49	27,12Ø,49,27,1Ø4,57
Ultra ROM	Ø	27,1Ø4,52

Set Option Summary for IG: and IGC:

Help	Print Set Option Summary With Status
Inter x	Non, Page, or Bank Interlace (Ø, 1, or 2)
Printer x	Printer Type (Epson, IDS, CITOH, MPI, Okidata)
Port x	IGC Port Address (8 Standard)
Term x,x,x,...	8 Byte Terminal Initialization Sequence

Loading the Device Driver

Once the device driver you are going to use has been configured properly, it is ready to be loaded into the operating system. An example of this under HDOS is as follows:

```
>LOAD IG:
```

Under CP/M type:

```
A>LOADD IGC:
```

If you get an error, it will most likely be because either the IG.DVD (HDOS) or IGC.DVD (CP/M) file is not present on the system disk. Error messages should be self explanatory.

Note that you may be prompted about space being available above the BIOS with CP/M. If you are unfamiliar with this, simply hit return. Refer to the section on the Loadd.COM utility for a complete explanation of this.

With a successful load, you are ready to draw. Under HDOS the IGC is now the device IG:, and under CP/M the IGC is the device PUN:. Remember that these device names can not be used to communicate with the IGC unless the IG: or IGC: device driver has been loaded without errors.

Executing Graphics Command Files

A graphics command file is an ASCII text file which contains a list of commands for the IG: device driver. These files can be identified by their .IGC extensions. If a graphics command file is complete it can simply by copied to the driver via PIP for execution. A number of these files are provided on the IGC software distribution diskettes. From HDOS type:

```
>PIP IG:=DEMO.IGC
```

From CP/M type:

```
A>PIP PUN:=DEMO.IGC
```

A series of circles should begin drawing on the screen. If nothing happens, then the IG: or IGC: driver is probably not loaded properly. If the image of the circles flickers excessively, or if they are not centered on the screen, then check the settings of the Term and Inter set options of the device driver and the configuration jumpers on the IGC circuit board.

Note that the circles may be slightly elongated or flattened. There are several adjustments on the original video board (underneath the CRT) that will correct this. Refer to your H/Z89 or H/Z19 Operation Manual.

Editing Graphics Commands

There are numerous ways to communicate to the IG: or IGC: driver. For experimentation, commands can be entered directly using the PIP utility. Under HDOS type:

```
>PIP IG:=TT:
Reset
Clear
Mode +Pixel
Circle Ø,Ø 1Ø 2Ø 3Ø 4Ø 5Ø 6Ø 7Ø 8Ø 9Ø 1ØØ
(Control D)
```

Under CP/M type:

```
A>PIP PUN:=CON:
Reset
Clear
Mode +Pixel
Circle Ø,Ø 1Ø 2Ø 3Ø 4Ø 5Ø 6Ø 7Ø 8Ø 9Ø 1ØØ
(Control Z)
```

If you wish to save your commands, use a text editor to create an IGC command file or modify an existing one. Try adding commands to the Demo.IGC file, and then copying your new version to the graphics driver with PIP as shown above.

Using Graphics from a Programming Language

Some graphics projects may be to complex to enter all of the commands from a text editor. In these cases you can either write a program in the language of your choice to create the IGC command file on disk, or write a program that talks to the graphics driver directly. The facility to communicate with the IG: and IGC: drivers from programming languages is provided in two ways.

Some languages allow the programmer to output data to a device name. Under HDOS this would be done using the device name IG:. Under CP/M this would be done using the device name of PUN: (after IGC: has been loaded).

Another method is provided which is well suited for use with any programming language. Commands to the IG: or IGC: device driver are simply output to the system console preceeded by a single ASCII control A character. These commands will be intercepted by the driver and executed, but will not be displayed on the screen. An example of this in BASIC would be:

```
1Ø PRINT CHR$(1);"Line -1ØØ,-1ØØ 1ØØ,1ØØ"
```

Remember that the graphics driver must first be loaded under CP/M before any communication with the driver is possible. Under HDOS the driver must be opened for write (which will load the driver automatically) before use.

The following are some examples of how to communicate to the IG: and IGC: drivers from several different BASIC languages. These languages differ primarily in how input/output is setup. The syntax for the individual Print commands is well standardized. Only BASIC will be used for examples for the sake of simplicity. However, it should be obvious how to implement the same instructions in any other language with which you are familiar.

Extended Benton Harbor BASIC under HDOS

```
10 OPEN "IG:" FOR WRITE AS FILE #1
20 PRINT CHR$(1);"Mode +Pixel"
30 CLOSE #1
```

With MicroSoft BASIC under HDOS (IG: must be loaded)

```
10 OPEN "O",1,"IG:"
20 PRINT CHR$(1);"Mode +Pixel"
30 CLOSE #1
```

With MicroSoft BASIC under CP/M (IGC: must be loaded)

```
10 PRINT CHR$(1);"Mode +Pixel"
```

The IG:/IGC: Driver Syntax

The syntax for commands to the IG: and IGC: drivers is very flexible, but certain guidelines must be observed. Please refer to the demo files provided on the IGC software distribution diskettes as examples of proper syntax.

Each line must either begin with a command, be completely blank, or begin with the comment symbol (apostrophe). All space and tab characters are ignored except when used as delimiters (to separate number values). Any command name can be abbreiviated to the minimum number of letters required to distiguish it from any other command name.

Space, tab, or commas can be used to delimit number values. Floating point values can be specified, but the decimal portion will be ignored.

The apostrophe character can appear anywhere in a line to designate a comment and the colon character can be used to concatenate any number of commands onto a single line. However, no single command can exceed 80 characters in length. Also, upper and lower case alphabetic characters are interchangeable.

The IG:/IGC: Coordinate System

The IG: and IGC: device drivers make use of the Cartesian coordinate system to specify individual pixel positions on the CRT screen. The location of the origin (coordinate Ø,Ø) can be defined to be any of three different positions by the Origin command.

Both absolute and relative coordinates are supported. With absolute coordinates, the first value in a coordinate pair (X) specifys a horizontal position and the second value (Y) specifys a vertical position. Any signed integer is valid for an X or Y coordinate so long as a boundary is not violated.

Two special symbols are employed for relative coordinates. The equal sign (=) can replace any X and Y coordinate pair to repeat the last coordinate given to the driver. The pound sign (#) can proceed any X and Y coordinate pair to indicate a position based on a pixel count from the current coordinate (=). Note that by definition the coordinate pairs = and #Ø,Ø are equivalent.

As an example, consider that a dot is placed at the absolute coordinate of 1ØØ,1ØØ with the bottom origin setting selected. The pixel immediately to the right of this one could be specified as either 1Ø1,1ØØ (absolute) or #1,Ø (relative). Note also that in either case the position of the symbol = would now be 1Ø1,1ØØ.

Similarly, to light the pixel at 1Ø1,98 (two pixels down from the last), a Dot command could be given the position as 1Ø1,98 (absolute) or #Ø,-2 (relative). The equal sign would now be positioned at 1Ø1,98.

The addressable pixel boundaries depend upon the current origin setting. However, the number of addressable pixels is always 64Ø horizontal by 5ØØ vertical. Note that in non-interlace mode the vertical pixel resolution of the hardware is limited to 25Ø, but the driver automatically scales this count to a constant 5ØØ for compatibility between non-interlace and interlace software.

The IG:/IGC: Command Set

The next section is a complete description of all of the commands of the IG: and IGC: device drivers. A symbolic summary of the syntax for each command is given followed by an explanation and programming examples.

In these symbolic syntax summaries a number of special symbols are used to represent certain syntactical possibilities and limitations. Optional parameters are enclosed within braces and the broken bar symbol is used to indicate a possible substitution.

Arc {{#}x1, y1}|{=}, r, a1, a2

The Arc command draws one or a series of any portion of a circle or ellipse. A coordinate pair must be supplied to determine the location of the arc to be drawn. This must be followed by a radius specification and two angle specifications to indicate the portion of the circle to be drawn.

All angles must be specified in positive integer degrees (Ø through 36Ø). Additional radius specifications may also be given after the second angle value.

An arc will be drawn in a counter-clockwise direction from the first angle supplied to the second angle using the supplied coordinate pair as an axis for each radius value given. All drawing modes and line styles are honored.

The current scale and rotation factors are also honored and can be used to draw virtually any curve desired. Also note that arcs can be drawn which partially extend past drawing boundaries.

Example 1: Draw one 9Ø degree arc with dashed line style.

```
Editor: Origin Center
        Mode -Erase
        Style 3
        Scale 1,1
        Arc Ø,Ø 2ØØ Ø 9Ø

BASIC:  1Ø X=Ø:Y=Ø:R=2ØØ:A=9Ø
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style 3"
        5Ø PRINT CHR$(1);"Scale 1,1"
        6Ø PRINT CHR$(1);"Arc",X,Y,R,Ø,A
```

Example 2: Draw 5 nested partial ellipses.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Scale 2,1
        Arc Ø,Ø 1ØØ Ø 27Ø 11Ø 12Ø 13Ø 14Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø"
        4Ø PRINT CHR$(1);"Scale 2,1"
        5Ø PRINT CHR$(1);"Arc Ø,Ø 1ØØ Ø 27Ø";
        6Ø PRINT "11Ø 12Ø 13Ø 14Ø"
```

Box {{#}x1, y1}|{=}, {#}x2, y2

The Box command can be used to draw one or a series of any square or rectangular polygons in the current drawing page. The current Line style will be used to draw each of the boxes. The currently selected drawing mode also affects the Box command.

Each box is specified by a set of four values. The first two values are the X and Y coordinates of the top left corner of the box and the second two values are the X and Y coordinates of the lower right corner of the box to be drawn. This format is similar to the Erase command.

Example 1: Draw a large box in center of page.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Box -2ØØ,-2ØØ 2ØØ,2ØØ

BASIC:  1Ø X1=-2ØØ:Y1=2ØØ:X2=2ØØ:Y2=-2ØØ
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style Ø"
        5Ø PRINT CHR$(1);"Box",X1,Y1,X2,Y2
```

Example 1: Erase a large box in center of page.

```
Editor: Origin Center
        Mode +Erase
        Style Ø
        Box -2ØØ,-2ØØ 2ØØ,2ØØ

BASIC:  1Ø X1=-2ØØ:Y1=2ØØ:X2=2ØØ:Y2=-2ØØ
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode +Erase"
        4Ø PRINT CHR$(1);"Style Ø"
        5Ø PRINT CHR$(1);"Box",X1,Y1,X2,Y2
```

Example 3: Draw a dashed box relative to origin.

```
Editor: Origin Center
        Mode -Erase
        Style 3
        Go Ø Ø
        Box = #1ØØ,1ØØ

BASIC:  1Ø S=1ØØ:X=Ø:Y=Ø
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style 3"
        5Ø PRINT CHR$(1);"Go",X,Y
        6Ø PRINT CHR$(1);"Box = #",S,S
```

Circle {{#}x1, y1}|{=}, r

The Circle command draws one or a series of circles or ellipses. This command must be supplied with a coordinate pair to determine where the center of the circle will be positioned, followed by one or a series of radius specifications. A circle will be drawn at the coordinate specified in the currently selected drawing page (refer to the Draw command) for each radius value given. All drawing modes and line styles are honored.

The current scale and rotation factors are also honored and can be used to draw virtually any ellipse desired. Also note that circles can be drawn which partially extend past drawing boundaries.

Example 1: Draw one large circle in center of page.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Circle Ø,Ø 2ØØ

BASIC:  1Ø X=Ø:Y=Ø:R=2ØØ
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style Ø"
        5Ø PRINT CHR$(1);"Circle",X,Y,R
```

Example 2: Draw 5 nested circles with dashed line style.

```
Editor: Origin Center
        Mode -Erase
        Style 3
        Circle Ø,Ø 1Ø 2Ø 3Ø 4Ø 5Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style 3"
        4Ø PRINT CHR$(1);"Circle Ø,Ø 1Ø 2Ø 3Ø 4Ø 5Ø"
```

Example 3: Draw one tilted ellipse in center of page.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Scale 2,1
        Rotate 45
        Circle Ø,Ø 12Ø

BASIC:  1Ø D=45:R=12Ø
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style Ø"
        5Ø PRINT CHR$(1);"Scale 2,1"
        6Ø PRINT CHR$(1);"Rotate",D
        7Ø PRINT CHR$(1);"Circle Ø,Ø",R
```

Clear

The Clear command erases all pages of the video memory using the currently selected clear style (refer to the Style command). The current style defines the new background pattern. Although the clear styles are much more limited than the paint styles that can be used with the Erase command, the Clear command is much faster. A Clear command is performed in about 60 milliseconds.

Example 1: Clear all pages to black.

```
Editor: Style 0,0,0
        Clear

BASIC:  10 PRINT CHR$(1);"Style 0,0,0"
        20 PRINT CHR$(1);"Clear"
```

Example 2: Clear all pages to a gray background.

```
Editor: Style 0,0,2
        Clear

BASIC:  10 PRINT CHR$(1);"Style 0,0,2"
        20 PRINT CHR$(1);"Clear"
```

Cursor {x, y}¦{On}¦{Off}

The Cursor command draws and moves a cross-hair graphics cursor to the coordinate specified in the current drawing page. The graphics cursor is always drawn in toggle drawing mode regardless of the true current drawing mode. This allows the cursor to be moved on top of another image without altering it. The cursor command also ignores the current line style setting, although various cursor styles can be selected with the Style command.

The Cursor command can be extremely helpful in developing interactive graphics software. However, using this command requires several important considerations to prevent the drawn image from being altered when the cursor is moved.

The cursor must be turned off before a drawing operation is performed that might overlap any portion of the cursor, and before selecting another drawing page if the cursor is going to be used in the new page. Note that the cursor will automatically be turned off by either a Reset or a Clear command.

Example 1: Turn on the graphics cursor.

```
Editor: Cursor On

BASIC:  1Ø PRINT CHR$(1);"Cursor On"
```

Example 2: Move the graphics cursor to the origin.

```
Editor: Cursor Ø,Ø

BASIC:  1Ø PRINT CHR$(1);"Cursor Ø,Ø"
```

Example 3: Move the graphics cursor relative.

```
Editor: Cursor #1ØØ,2ØØ

BASIC:  1Ø PRINT CHR$(1);"Cursor #1ØØ,2ØØ"
```

Display {p} {,s}

The Display command allows any of three video pages to be selected for display on the screen and (optionally) from which scan line of that page the image will be displayed. A separate image can be drawn in each of the three pages (refer to the Draw command). Note that using the scan line feature will cause portions of the next page to be moved into the display area. Refer to the demonstration programs at the end of this section for examples of how the Display and Draw commands can be used for animation.

In either non-interlace or bank interlace modes, the valid page numbers are 0, 1, and 2. However, the use of page interlace reduces the page count to 1 which is specified as 0 (refer to the Inter set option). Valid scan line numbers are 0 through 65535.

Example 1: Display first page.

```
Editor: Display 0

BASIC:  PRINT CHR$(1);"Display 0"
```

Example 2: Display second page from third scan line.

```
Editor: Display 1 2

BASIC:  10 P=1:S=2
        20 PRINT CHR$(1);"Display",P,S
```

Example 3: Display from fourth scan line of current page.

```
Editor: Display ,3

BASIC:  10 S=3
        20 PRINT CHR$(1);"Display ,",S
```

Dot {{#}x1, y1}|{=}

The Dot command draws a single pixel dot, or a series of dots, based on the supplied coordinates. A dot will be drawn for each pair of coordinates given. The position of the coordinates is determined by the current origin position (refer to the Origin command) and dots will only be drawn in the currently selected drawing page (refer to the Draw command). Also note that the Erase and Toggle drawing modes affect the Dot command (refer to the Mode command). The Dot command is not affected by any of the drawing styles.

Example 1: Draw a dot at the origin.

```
Editor: Mode -Erase
        Dot Ø,Ø

BASIC:  1Ø X=Ø:Y=Ø
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Dot",X,Y
```

Example 2: Draw a four dot sequence from the origin.

```
Editor: Go Ø Ø
        Mode -Erase
        Dot = #1,1 #1,1 #1,1

BASIC:  1Ø PRINT CHR$(1);"Go Ø,Ø"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Dot = #1,1 #1,1 #1,1"
```

Draw {p} {,s}

The Draw command allows any of three video pages to be selected for drawing and (optionally) from which scan line of that page the image will be drawn. A separate image can be drawn in each of the three pages independent of the page that is being displayed (refer to the Display command).

In either non-interlace or bank interlace modes, the valid page numbers are Ø, 1, and 2. However, the use of page interlace reduces the page count to 1 which is specified as Ø (refer to the Inter set option). Valid scan line numbers are Ø through 65535.

Example 1: Select first page for drawing.

```
        Editor: Draw Ø

        BASIC:  PRINT CHR$(1);"Draw Ø"
```

Example 2: Select second page, third scan line for drawing.

```
        Editor: Draw 1 2

        BASIC:  1Ø P=1:S=2
                2Ø PRINT CHR$(1);"Draw",P,S
```

Erase {{#}x1, y1}|{=}, {#}x2, y2

The Erase command can be used to erase, paint, or invert any rectangular polygon in the current drawing page. The specified area will be erased using the current painting style and drawing mode.

Each area to be erased is specified by a set of four values. The first two values are the X and Y coordinates of the top left corner of the erase area and the second two values are the X and Y coordinates of the lower right corner of the area to be erased. This format is similar to the Box command.

Because of its much faster drawing speed, the Erase command should be used instead of the Paint command when the polygon involved is rectangular in shape.

Example 1: Erase entire page to black.

```
Editor: Origin Bottom
        Mode +Erase
        Style Ø,Ø
        Erase Ø,Ø 639,499

BASIC:  1Ø PRINT CHR$(1);"Origin Bottom"
        2Ø PRINT CHR$(1);"Mode +Erase"
        3Ø PRINT CHR$(1);"Style Ø,Ø"
        4Ø PRINT CHR$(1);"Erase Ø,Ø 639,499"
```

Example 2: Paint square area with dot pattern.

```
Editor: Origin Center
        Mode -Erase
        Style Ø,7
        Erase -2ØØ,-2ØØ 2ØØ,2ØØ

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø,7"
        4Ø PRINT CHR$(1);"Erase -2ØØ,-2ØØ 2ØØ,2ØØ"
```

Example 3: Invert lower half of page.

```
Editor: Origin Bottom
        Mode +Toggle
        Style Ø,Ø
        Erase Ø,Ø 639,25Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Bottom"
        2Ø PRINT CHR$(1);"Mode +Toggle"
        3Ø PRINT CHR$(1);"Style Ø,Ø"
        4Ø PRINT CHR$(1);"Erase Ø,Ø 639,25Ø"
```

Get {{#}x1, y1}|{=}

The Get command returns the decimal value of the binary number formed by the 16 pixels beginning from the specified coordinate of the current drawing page. This command can be used from an application program to input any portion of the binary video image for direct manipulation, storage in a disk file, etc. Some knowledge of graphics hardware is recommended to use this command, refer to the Advanced Programmer's Guide of this text.

Note that many BASIC interpreters consider the most significant bit of a sixteen bit word to be a sign bit. Therefore, the Get command will return a value preceded with a negative symbol (-) to indicate that the high bit is set.

The syntax given in the following examples may need to be altered to prevent the BASIC interpreter you are using from printing the question mark (?) symbol when the INPUT command is used.

Example 1: Get video image word at origin.

```
BASIC:  10 PRINT CHR$(1);"Get 0,0"
        20 INPUT A
        30 PRINT A;" is the video image word"
```

Example 2: Get video image word at a relative coordinate.

```
BASIC:  10 X=10:Y=20
        20 PRINT CHR$(1);"Get #",X,Y
        30 INPUT A
        40 PRINT A;" is the video image word"
```

Go {{#}x, y}|{=}

The Go command allows the current or relative coordinate (=) to be moved in the current drawing page without plotting any pixels. An initial Go command can be used to position a relative drawing on the screen.

Example 1: Place the relative coordinate at page center.

```
        Editor: Origin Center
                Go Ø,Ø

        BASIC:  1Ø X=Ø:Y=Ø
                2Ø PRINT CHR$(1);"Origin Center"
                3Ø PRINT CHR$(1);"Go",X,Y
```

Example 2: Move the relative coordinate 1ØØ pixels upward.

```
        Editor: Origin Center
                Go #Ø,1ØØ

        BASIC:  1Ø X=Ø:Y=1ØØ
                2Ø PRINT CHR$(1);"Origin Center"
                3Ø PRINT CHR$(1);"Go #",X,Y
```

Input {Left}|{Right}

The Input command returns both the directional and fire button status from either the left or the right input device (trackball or joystick). The device to be polled must be indicated after the Input command by specifying either Left or Right. The format for the values returned is listed in the chart below.

This command is only for use in application programs and can not be used in IGC command files. Also, the syntax given in the following examples may need to be altered to prevent the BASIC interpreter you are using from printing the question mark symbol (?) when the INPUT command is used.

Input Device Format

Fire Button Not Pressed	Fire Button Pressed
Ø - No Motion	1Ø - No Motion
1 - Up	11 - Up
2 - Up and Right	12 - Up and Right
3 - Right	13 - Right
4 - Down and Right	14 - Down and Right
5 - Down	15 - Down
6 - Down and Left	16 - Down and Left
7 - Left	17 - Left
8 - Up and Left	18 - Up and Left

Example 1: Monitor the status of the right input device.

```
BASIC:    1Ø PRINT CHR$(1);"Input Right"
          2Ø INPUT A
          3Ø IF A<1Ø THEN GOTO 6Ø
          4Ø PRINT "Fire Button is Pressed"
          5Ø A=A-1Ø
          6Ø ON A GOTO 8Ø 9Ø 1ØØ 11Ø 12Ø 13Ø 14Ø 15Ø
          7Ø GOTO 1Ø
          8Ø PRINT "Up":GOTO 1Ø
          9Ø PRINT "Up and Right":GOTO 1Ø
         1ØØ PRINT "Right":GOTO 1Ø
         11Ø PRINT "Down and Right":GOTO 1Ø
         12Ø PRINT "Down":GOTO 1Ø
         13Ø PRINT "Down and Left":GOTO 1Ø
         14Ø PRINT "Left":GOTO 1Ø
         15Ø PRINT "Up and Left":GOTO 1Ø
```

Line {{#}x1, y1}|{=}, {#}x2, y2

The Line command draws one, or a series of lines based on the supplied coordinates. A line will be drawn for every two pairs of coordinates given. The position of the coordinates is determined by the current origin setting (refer to the Origin command), and lines will only be drawn in the currently selected drawing page (refer to the Draw command). Also note that the Erase and Toggle drawing modes (refer to the Mode command) and the Line styles (Refer to the Style command) affect the Line Command.

With the Line command, the current and relative coordinate symbols (= and #) are extremely important. Images drawn with Line commands using relative coordinates can easily be moved, enlarged, and turned with the Go, Scale, and Rotate commands respectively.

Example 1: Draw a line with absolute coordinates.

```
Editor: Origin Bottom
        Mode -Erase
        Style 0
        Line 0,0 639,499

BASIC:  10 PRINT CHR$(1);"Origin Bottom"
        20 PRINT CHR$(1);"Mode -Erase"
        30 PRINT CHR$(1);"Style 0"
        40 PRINT CHR$(1);"Line 0,0 639,499"
```

Example 2: Draw a right triangle with dotted lines.

```
Editor: Origin Center
        Mode -Erase
        Style 1
        Line 0,0 0,50 0,50 50,0 50,0 0,0

BASIC:  10 PRINT CHR$(1);"Origin Center"
        20 PRINT CHR$(1);"Mode -Erase"
        30 PRINT CHR$(1);"Style 1"
        40 PRINT CHR$(1);"Line 0,0 0,50 0,50 50,0";
        50 PRINT "50,0 0,0"
```

Example 3: Draw a right triangle with relative coordinates.

```
Editor: Origin Center
        Mode -Erase
        Style 0
        Go 0 0
        Line = #0,50 = #50,-50 = #-50,0

BASIC:  10 PRINT CHR$(1);"Origin Center"
        20 PRINT CHR$(1);"Mode -Erase"
        30 PRINT CHR$(1);"Style 0"
        40 PRINT CHR$(1);"Go 0 0"
        50 PRINT CHR$(1);"Line = #0,50 = #50,-50";
        60 PRINT "= #-50,0"
```

Mode {{-}|{=}|{+}} {Pixel} {Character} {Inverse} {Alternate} {{Erase}|{Toggle}}

The Mode command is used to configure all of the various operating modes of the graphics driver. By using the plus and minus signs any combination of modes may be turned on or off without affecting the status of other modes. The equal sign allows a new mode status to be given which completely replaces the old configuration. The plus sign is the default.

The Pixel Video Mode

The Pixel video mode turns on the high resolution graphics of the IGC to the screen. Note that if the currently displayed page is blank, then this mode can be on without being detectable on the screen. This mode defaults to off.

The Character Video Mode

The Character video mode turns on the normal ASCII characters of the H/Z89 or H/Z19 screen. If this mode is turned off any characters on the screen will become invisible eventhough they may still be present. This mode defaults to on.

The Inverse Video Mode

The Inverse video mode causes the entire screen (both pixel and character video) to be displayed inverted. All black pixels will be displayed as white and all white pixels will be displayed as black. Note that this inversion only affects the display. The contents of the video memory is not affected. This mode defaults to off.

The Alternate Character Set Mode

The Alternate character set mode turns off the normal ASCII characters of the H/Z89 or H/Z19 and replaces them with an alternate set of characters. This mode can only be used if you have installed an alternate character set generation ROM, and the particular type of character ROM installed will determine the type of characters you have as an alternate set. This mode defaults to off.

The Erase Drawing Mode

The Erase drawing mode causes most of the drawing commands to plot black pixels instead of white for the purpose of erasing items which have been drawn previously. Drawing in Erase mode on an erased background will have no effect. Note that selecting the Erase drawing mode automatically deselects the Toggle drawing mode. This mode defaults to off.

The Toggle Drawing Mode

The Toggle drawing mode causes most of the drawing commands to plot images by toggling the state of overwritten pixels. White pixels will become black, and black pixels will become white, when overwritten. This allows images to be drawn which will always be visible regardless of the background. Note that selecting the Toggle drawing mode automatically deselects the Erase drawing mode. This mode defaults to off.

Example 1: Turn on high resolution pixel video.

```
Editor: Mode +Pixel

BASIC:  10 PRINT CHR$(1);"Mode +Pixel"
```

Example 2: Turn off pixel video and turn on video invert.

```
Editor: Mode -Pixel +Invert

BASIC:  10 PRINT CHR$(1);"Mode -Pixel +Invert"
```

Example 3: Turn off character video and video invert.

```
Editor: Mode -Character -Invert

BASIC:  10 PRINT CHR$(1);"Mode -Character -Invert"
```

Example 4: Turn both video modes on, all other modes off.

```
Editor: Mode =Pixel Character

BASIC:  10 PRINT CHR$(1);"Mode =Pixel Character"
```

Example 5: Deselect Erase drawing mode to default.

```
Editor: Mode -Erase

BASIC:  10 PRINT CHR$(1);"Mode -Erase"
```

Example 6: Return all modes to set defaults.

```
Editor: Mode

BASIC:  10 PRINT CHR$(1);"Mode"
```

Origin {Top}|{Center}|{Bottom}

The Origin Command allows the user to select one of four different implementations of the cartesian coordinate system. These origin modes differ in where the coordinate Ø,Ø is located on the screen, and in the size and direction of the Y coordinate. The Center origin mode is the default.

The Center Origin Mode

The Center origin mode defines the drawing page size to be equal to the screen size with the Ø,Ø coordinate in the screen's center. This logically divides the screen/page into four quadrants, -X +Y, +X +Y, -X -Y, and +X -Y. This origin mode is recommended for most applications since it makes it very easy to draw objects which are centered on the screen. The drawing boundaries in this mode are -319, -249 through 32Ø, 25Ø.

The Bottom Origin Mode

The Bottom origin mode defines the drawing page size to be equal to the screen size with the Ø,Ø coordinate at the bottom left corner of the screen/page. This mode has the advantage that it eliminates negative absolute coordinates. The screen/page is effectively one positive quadrant (+X, +Y). The drawing boundaries in this mode are Ø, Ø through 639, 499.

The Top Origin Mode

The Top origin mode defines the drawing page size to be equal to the screen size with the Ø,Ø coordinate at the top left corner of the screen/page. This mode also eliminates negative absolute coordinates, but the Y direction is reversed (Y = Ø is at the top of the screen/page). This origin mode is provided for compatibility with graphics software developed on other systems which only support a Top origin mode. The drawing boundaries in this mode are Ø, Ø through 639, 499.

The Large Origin Mode

The Large origin mode defines the drawing page size to be three times the screen size with the Ø,Ø coordinate at the top left corner of the first screen. This mode is similar to the Top origin mode except that all drawing commands now have a much larger boundary area (Ø, Ø through 639, 1535).

Note that this mode limits the usable drawing pages to Ø only, and that this mode can not be used in conjunction with page mode interlace.

Example 1: Set page Center origin mode.

```
Editor: Origin Center

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
```

Paint {{#}x1, y1}|{=}

The Paint command fills any outlined polygon area with the currently selected Paint style (see the Style command). Any number of X and Y coordinate pairs can be specified. However, caution must be taken that each coordinate is completely surrounded by a solid border (such as a triangle or circle). The Paint command is affected by the Draw and Style commands. It is not affected by the Mode command.

Either a white or black area can be painted so long as it is thoroughly "sealed off" by the opposite color. The state of the pixel at the coordinate specified to the Paint command determines the color of the area to be painted. Any pixels of the opposite color will then be seen as a border. Note that this can be used to invert any of the painting style patterns as shown in example 3 below.

The Paint command handles complex shapes with "islands" by filling around the interior polygons without leaving any portion unpainted. However, this function is extremely memory intensive and there are limits to the complexity of the shapes that can be painted. Also note that complex shapes with "islands" will paint more effectively with more solid painting styles than the more sparse styles.

Example 1: Draw and paint a large solid circle.

```
Editor: Origin Center
        Mode -Erase
        Style Ø,Ø
        Circle Ø,Ø 2ØØ
        Paint Ø,Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø,Ø"
        4Ø PRINT CHR$(1);"Circle Ø,Ø 2ØØ"
        5Ø PRINT CHR$(1);"Paint Ø,Ø"
```

Example 2: Draw a circular polygon with a striped pattern.

```
Editor: Origin Center
        Mode -Erase
        Style Ø,2
        Go Ø,Ø
        Circle = 1ØØ
        Paint =
        Mode +Erase
        Circle = 1ØØ

BASIC:  1Ø X=Ø:Y=Ø:R=1ØØ
        2Ø PRINT CHR$(1);"Origin Center"
        3Ø PRINT CHR$(1);"Mode -Erase"
        4Ø PRINT CHR$(1);"Style Ø,2"
        5Ø PRINT CHR$(1);"Go",X,Y
        6Ø PRINT CHR$(1);"Circle =",R
        7Ø PRINT CHR$(1);"Paint ="
        8Ø PRINT CHR$(1);"Mode +Erase"
        9Ø PRINT CHR$(1);"Circle =",R
```

Example 3: Draw a circular polygon with an inverted style.

```
Editor: Origin Center
        Mode -Erase
        Style Ø,Ø
        Circle Ø,Ø 1ØØ
        Paint =
        Style Ø,1
        Paint =

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø,Ø"
        4Ø PRINT CHR$(1);"Circle Ø,Ø 1ØØ"
        5Ø PRINT CHR$(1);"Paint ="
        6Ø PRINT CHR$(1);"Style Ø,1"
        7Ø PRINT CHR$(1);"Paint ="
```

Pan {Left}|{Right}|{Up}|{Down} c {,s}

The Pan command will scroll the entire graphics screen display in any of four directions a specified number of pixels and at a specified speed. The first parameter required is a direction specification which can be either Left, Right, Up or Down. The image on the screen will move in the opposite direction of this specification.

Vertical pans will be performed on an even field scan line basis across all 3 display pages (one in page interlace mode). Horizontal pans will be performed with an increment of eight pixels at a time and the left and right screen borders will wrap around.

The supplied pixel count must be a positive integer and should not exceed the top of the first display page or the bottom of the last. If these boundaries are violated the excess will simply be ignored and an error will be reported if the trace mode is enabled (refer to the Trace command).

The final parameter is an optional speed specification. This value must be a positive integer between Ø (fastest) and 255 (slowest). If no speed is specified a value of zero will be assumed.

Example 1: Pan Down and back up.

```
Editor: Display Ø
        Pan Down 512,1Ø
        Pan Up 512,1Ø

BASIC:  1Ø C=512:S=1Ø
        2Ø PRINT CHR$(1);"Display Ø"
        3Ø PRINT CHR$(1);"Pan Down",C,S
        4Ø PRINT CHR$(1);"Pan Up",C,S
```

Example 2: Pan Right and wrap screen around 1Ø times fast.

```
Editor: Display Ø
        Pan Right 64ØØ

BASIC:  1Ø C=64ØØ
        2Ø PRINT CHR$(1);"Display Ø"
        3Ø PRINT CHR$(1);"Pan Right",C
```

Pattern {bØ},{b1},{b2},{b3},{b4},{b5},{b6},{b7}

The Pattern command will create a user defined painting style that can be used by the Paint and Erase commands which consists of any 8 x 8 bit (pixel) matrix. This new style must then be selected by the Style command as paint style number 5Ø.

The eight by eight bit matrix must be supplied as 8 decimal values between Ø and 255. The first value will define the top of the matrix and the last value will define the bottom of the matrix with the least significant bits to the left on the screen (refer to diagram below).

The best way to create a paint style pattern is to draw a diagram as shown below and fill in the desired dot pattern. This diagram can then be converted to the 8 decimal parameters needed by adding the bit values of the pixels that are on.

New Painting Style Pattern Calculation Example

Decimal Bit Values

	1	2	4	8	16	32	64	128	
BYTE Ø	0	0	0						VALUE = 7
BYTE 1		0	0	0					VALUE = 14
BYTE 2			0	0	0				VALUE = 28
BYTE 3				0	0	0			VALUE = 56
BYTE 4				0	0	0			VALUE = 56
BYTE 5			0	0	0				VALUE = 28
BYTE 6		0	0	0					VALUE = 14
BYTE 7	0	0	0						VALUE = 7

Example 1: Define a new painting style pattern.

```
Editor: Style Ø,5Ø
        Pattern 7,14,28,56,56,28,14,7

BASIC:  1Ø PRINT CHR$(1);"Style Ø,5Ø"
        2Ø PRINT CHR$(1);"Pattern 7,14,28,56";
        3Ø PRINT "56,28,14,7"
```

Print {p},{s},{c}

The Print command duplicates the drawn graphics in any page to a dot matrix print device. The Print command can be followed by a page specification (0 through 2), or a scan line specification (0 through 65535), or a count of how many scan lines to print (0 through 750), or any combination of these three parameters. If none of these values are given, all of the currently selected display page will be printed.

The print command is very flexible in the different types of printers and printer interfaces that it supports. However, the IG:/IGC: driver must be configured for the proper hardware before this command will operate (refer to the Printer set option).

Before a Print command can be performed, the SP:/SPC: device driver must be loaded into memory. This is because the Print command transmits its data to the print spooler driver for output to allow for graphics spooling (refer to the section on the SP:/SPC: device driver).

Example 1: Copy all of the current page to the printer.

```
Editor: Print

BASIC:  10 PRINT CHR$(1);"Print"
```

Example 2: Copy top half of page 2 to the printer.

```
Editor: Print 2,0,125

BASIC:  10 P=2
        20 PRINT CHR$(1);"Print",P,0,125
```

Example 3: Copy bottom half of current page to the printer.

```
Editor: Print ,125,125

BASIC:  10 S=125
        20 PRINT CHR$(1);"Print ,",S,125
```

Put {{#}x1, y1}|{=}, n

The Put command writes the binary bit pattern image of the decimal value specified into the current drawing page at the coordinate specified. This command can be used to draw an image directly from an array of video data. Some knowledge of graphics hardware is recommended to use this command, refer to to the Advanced Programmer's Guide section of this text.

Note that many BASIC interpreters consider the most significant bit of a sixteen bit word to be a sign bit. Therefore, it may be necesary to specify a value preceded with a negative symbol (-) to use the high bit.

Example 1: Put a video image word at origin.

```
Editor: Put 0,0 43690

BASIC:  10 X=0:Y=0:I=43690
        20 PRINT CHR$(1);"Put",X,Y,I
```

Example 2: Put a video image word at a relative coordinate.

```
Editor: Put #10,20 21845

BASIC:  10 X=10:Y=20:I=21845
        20 PRINT CHR$(1);"Put #";X,Y,I
```

Rotate d

The Rotate command turns images drawn with relative coordinates. A rotation factor must be specified in positive integer degrees between Ø and 36Ø (defaults to zero). All relative coordinates and angles will be rotated in a counter clockwise direction by this degree specification. Absolute coordinates are not affected.

Example 1: Draw a triangle and rotate it 45 degrees.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Line Ø,Ø #Ø,5Ø = #5Ø,-5Ø = #-5Ø,Ø
        Rotate 45
        Line Ø,Ø #Ø,5Ø = #5Ø,-5Ø = #-5Ø,Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø"
        4Ø PRINT CHR$(1);"Line Ø,Ø #Ø,5Ø = #5Ø,-5Ø";
        5Ø PRINT "= #-5Ø,Ø"
        6Ø PRINT CHR$(1);"Rotate 45"
        7Ø PRINT CHR$(1);"Line Ø,Ø #Ø,5Ø = #5Ø,-5Ø";
        8Ø PRINT "= #-5Ø,Ø"
```

Example 2: Draw an ellipse and rotate it 315 degrees.

```
Editor: Origin Center
        Mode -Erase
        Style Ø
        Scale 2,1
        Circle Ø,Ø 12Ø
        Rotate 315
        Circle Ø,Ø 12Ø

BASIC:  1Ø PRINT CHR$(1);"Origin Center"
        2Ø PRINT CHR$(1);"Mode -Erase"
        3Ø PRINT CHR$(1);"Style Ø"
        4Ø PRINT CHR$(1);"Scale 2,1"
        5Ø PRINT CHR$(1);"Circle Ø,Ø 12Ø"
        6Ø PRINT CHR$(1);"Rotate 315"
        7Ø PRINT CHR$(1);"Circle Ø,Ø 12Ø"
```

Reset

The Reset command causes the graphics driver to initialize all modes and settings to their default values. This command is simply a convenient way to begin a graphics program. Well designed programs should always begin with a Reset command. Below is a list of commands that are equivalent to a Reset command. Refer to the text regarding each of these commands for details on their effects. All other commands are unaffected.

```
Mode
Origin Center
Draw Ø
Display Ø
Style Ø,Ø,Ø,Ø
Scale 1,1
Rotate Ø
Cursor Off
Go Ø,Ø
```

Example 1: Reset graphics driver to defaults.

```
        Editor: Reset

        BASIC:  1Ø PRINT CHR$(1);"Reset"
```

Scale x, y

The Scale command enlarges and/or flips images drawn with relative coordinates. A scaling factor (multiplier) for both X and Y must be specified which can be any integers between -255 to +255, except for zero (defaults to 1, 1). All relative coordinates and radii will be multiplied by these factors. Absolute coordinates are not affected.

Negative scaling factors will cause coordinate directions to reverse, which is usefull in flipping drawings.

Example 1: Draw three triangles scaled to different sizes.

```
Editor: Origin Bottom
        Mode -Erase
        Style 0
        Go 0,0
        Scale 1,3
        Line = #0,50 = #50,-50 = #-50,0
        Go 50,50
        Scale 2,2
        Line = #0,50 = #50,-50 = #-50,0
        Go 100,100
        Scale 3,1
        Line = #0,50 = #50,-50 = #-50,0

BASIC:   10 X=1:Y=3
         20 PRINT CHR$(1);"Origin Bottom"
         30 PRINT CHR$(1);"Mode -Erase"
         40 PRINT CHR$(1);"Style 0"
         50 PRINT CHR$(1);"Go 0,0"
         60 FOR A=1 TO 3
         70 PRINT CHR$(1);"Scale",X,Y
         80 PRINT CHR$(1);"Line = #0,50 = #50,-50";
         90 PRINT "= #-50,0"
        100 PRINT CHR$(1);"Go #50,50"
        110 X=X+1:Y=Y-1
        120 NEXT A
```

Example 2: Scale and flip triangle.

```
Editor: Origin Center
        Mode -Erase
        Style 0
        Go 0,0
        Scale -4,-4
        Line = #0,50 = #50,-50 = #-50,0

BASIC:  10 PRINT CHR$(1);"Origin Center"
        20 PRINT CHR$(1);"Mode -Erase"
        30 PRINT CHR$(1);"Style 0"
        40 PRINT CHR$(1);"Go 0,0"
        50 PRINT CHR$(1);"Scale -4,-4"
        60 PRINT CHR$(1);"Line = #0,50 = #50,-50";
        70 PRINT "= #-50,0"
```

Style {l},{p},{cl},{cu}

The Style command selects the various patterns to be used by most of the drawing commands. A maximum of 4 values can be specified in a Style command.

Line Styles

The first value indicates the desired Line style. Any of 12 different Line styles can be specified (Ø through 11) which consist of various combinations of dots and dashes (see chart below). The line style is used by the Line, Box, Circle, and Arc commands.

Painting Styles

The second value indicates the desired Paint style. Any of 5Ø different predefined Paint styles can be specified here (Ø through 49). In addition, style 5Ø can also be specified which is a user definable style (refer to the Pattern command). The painting style is used by the Paint and Erase commands.

Clear Styles

The third value indicates the desired Clear style. Any of 4 different Clear styles can be specified (Ø through 3). These styles consist of levels of gray which are well suited for background shading. This style is used only by the Clear command.

Cursor Styles

The forth value indicates the desired graphics cursor style. Any of 4 different cursor styles can be specified (Ø through 3) which consist of various sizes of crosshairs. The cursor style is used only by the Cursor command.

Style Definitions

Line Styles

#0	Solid
#1	Fine Dots
#2	Fine Dashes
#3	Sparse Dashes
#4	Coarse Dashes
#5	Sparse Small Dots
#6	Extra Small Dashes
#7	Small Dashes
#8	Medium Dashes
#9	Large Dashes
#10	Extra Large Dashes
#11	Very Large Dashes

Clear Styles

#0	Solid Black
#1	Fine Vertical Stripes
#2	Gray
#3	Solid White

Cursor Styles

#0	Small
#1	Medium
#2	Large
#3	Full Screen

Painting Styles

#0	Solid	#1	Sparse Diagonal Stripes
#2	Diagonal Stripes	#3	Sparse Diagonal Stripes
#4	Diagonal Stripes	#5	Crossed Diagonal Stripe
#6	Dense Crossed Stripes	#7	Sparse Diagonal Dots
#8	Diagonal Dots	#9	Large Dots
#10	Dense Dots	#11	Dot Array
#12	Sparse Vertical Stripes	#13	Vertical Stripes
#14	Dense Vertical Stripes	#15	Sparse Horizontal Stripes
#16	Horizontal Stripes	#17	Crossed Sparse Stripes
#18	Crossed Stripes	#19	Sparse Vertical Zig Zag
#20	Vertical Zig Zag	#21	Sparse Horizontal Zig Zag
#22	Horizontal Zig Zag	#23	Crossed Sparse Zig Zag
#24	Quilt	#25	Large Random Dots
#26	Brick Wall	#27	Diagonal Brick Wall
#28	Tire Tread	#29	Truck Bumper
#30	Diagonal Squares with Gray	#31	Cheverons
#32	Waffles	#33	Air Filter
#34	Fish Scales	#35	Large Boxes
#36	Wallpaper	#37	Diagonal Weave
#38	Diagonal Checker Board	#39	Checker Board
#40	Honey Comb	#41	Double Weave
#42	Large Circles	#43	Large Solid Triangles
#44	Molecules	#45	Weave
#46	Recursive Squares	#47	Tees
#48	Stars	#49	Macrame

Example 1: Set solid line style.

```
Editor: Style 0

BASIC:  10 S=0
        20 PRINT CHR$(1);"Style",S
```

Example 3: Change Clear style only.

```
Editor: Style ,,2

BASIC:  10 PRINT CHR$(1);"Style ,,2"
```

Test {{#}x1, y1}¦{=}

The Test command returns a value of either Ø or 1 to indicate the status of the pixel at the specified X and Y coordinate position. A value of Ø will be returned if the pixel is on (white) and a value of 1 will be returned if the pixel is off (black). This command is only for use in an application program and can not be used in IGC command files. Note that the pixel tested will be in the currently selected drawing page.

The syntax given in the following examples may need to be altered to prevent the BASIC interpreter you are using from printing the question mark (?) symbol when the INPUT command is used.

Example 1: Test state of pixel at origin.

```
BASIC:   1Ø PRINT CHR$(1);"Test Ø,Ø"
         2Ø INPUT A
         3Ø IF A=Ø THEN PRINT "Pixel is not on"
```

Example 2: Test state of pixel at relative coordinate.

```
BASIC:   1Ø X=1Ø:Y=2Ø
         2Ø PRINT CHR$(1);"Test #",X,Y
         3Ø INPUT A
         4Ø IF A=Ø THEN PRINT "Pixel is not on"
```

Text {"string"} {,n}

The Text command allows any string of ASCII characters to be output to the system console. Any combination of printable characters can be specified by enclosing them in quotation marks except for the quotation mark character. All other characters can be specified by giving their decimal values with each character separated by a comma.

This command is very usefull in creating IGC command files which take full advantage of the capabilities of both the IGC and the H/Z19 or H/Z89 console. This command is not recommended for use in application programs since most programming languages can output characters directly to the console. The programming examples given below in BASIC do not demonstrate the use of the Text command, but rather a more direct alternative.

Example 1: Erase characters and enable the 25th line.

```
Editor: Text 27,"E",27,"x1",10

BASIC:  10 PRINT CHR$(27);"E";CHR$(27);"x1"
```

Example 2: Print a Hello message in inverse video.

```
Editor: Text 27,"pHello",27,"q",10

BASIC:  10 PRINT CHR$(27);"pHello";CHR$(27);"q"
```

Trace {On}|{Off}

The Trace command is a special debugging feature to aid in locating errors in IGC graphics programs. The syntax for this command simply consist of either an On or Off specification. This mode is not disabled by a Reset command.

When the trace mode is turned on all commands received by the graphics driver will be echoed to the system console as they are executed. Furthermore, commands which contain an error will be displayed with a question mark (?) and a bell character. There are three types of errors that can cause the error symbol to be displayed as outlined below.

Syntax Error

The driver can not identify the command given. Check your spelling and be sure you are following the correct format for the command as outlined in this manual.

Value Error

One of the values supplied in this command exceeds the acceptable limits for the current mode configurations. Check the stated ranges for the parameters involved.

Boundary Error

One of the coordinate pairs supplied in this command is not within the current drawing page boundaries. This error is very common when working with the Scale and Rotate commands. Try smaller scaling factors and moving the drawing in the page.

```
Example 1: Enable trace mode.

        Editor: Trace On

        BASIC:  10 PRINT CHR$(1);"Trace On"

Example 2: Disable trace mode.

        Editor: Trace Off

        BASIC:  10 PRINT CHR$(1);"Trace Off"
```

Wait s

The Wait command allows pauses to be inserted into an IGC graphics file. A count of the number of seconds to wait must be specified with a positive integer between Ø and 255.

The timing for this command is done using the standard Heath 2 millisecond interrupt driven clock. If this hardware does not exist in your system, or if you are running special software which affects this interrupt, the Wait command may not function properly.

```
Example 1: Wait for 1 second.

        Editor: Wait 1

        BASIC:  1Ø S=1
                2Ø PRINT CHR$(1);"Wait",S

Example 2: Wait for 1 minute.

        Editor: Wait 6Ø

        BASIC:  1Ø S=6Ø
                2Ø PRINT CHR$(1);"Wait",S
```

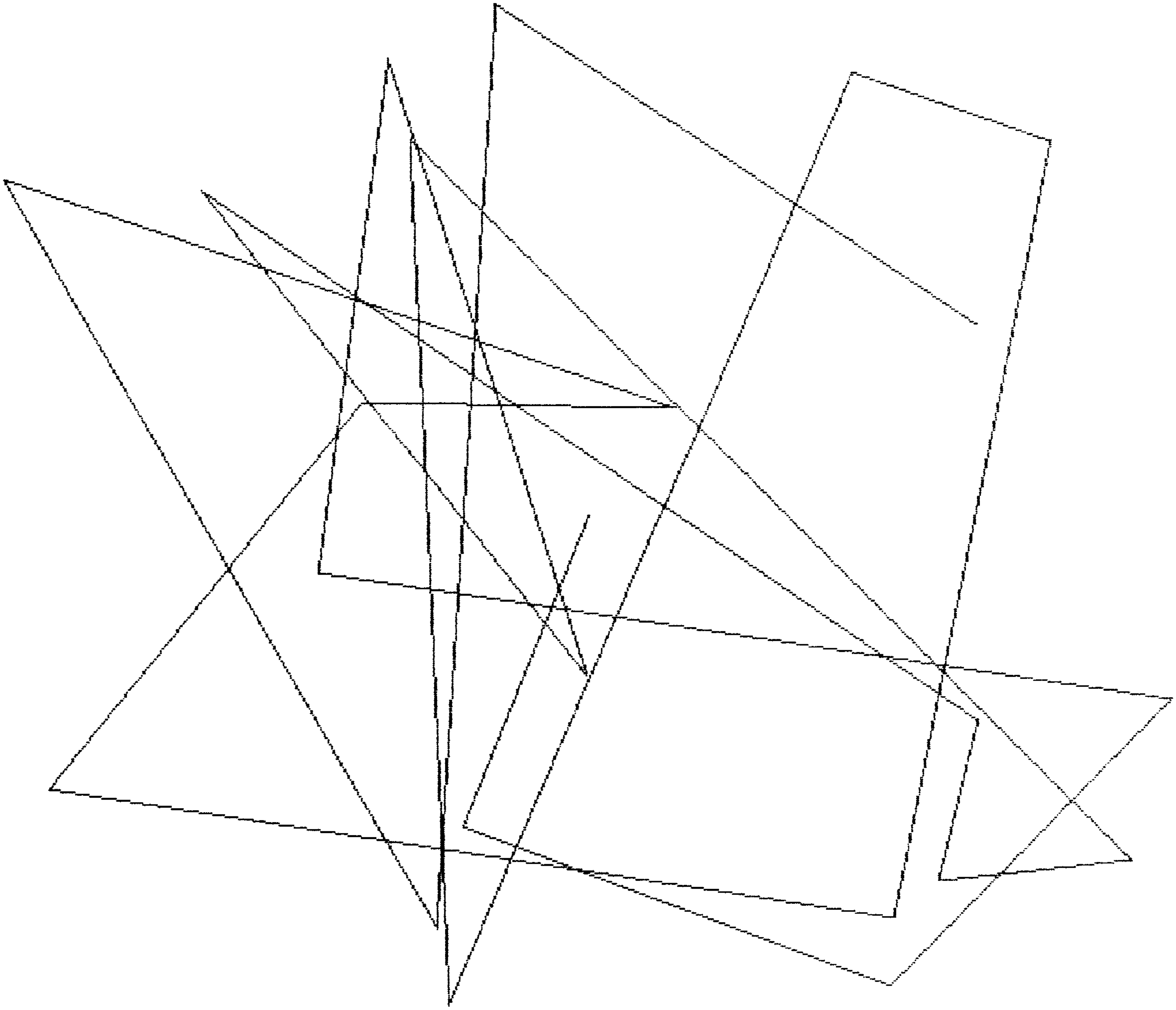
Lines.BAS

```
00040 PRINT CHR$(27);"E";CHR$(27);"x1";CHR$(27);"x5"
00050 A$=CHR$(1)
00060 PRINT A$;"Reset:Clear:Origin Bottom:Mode +Pixel"
00070 FOR S=0 TO 11
00080 FOR A=1 TO 100
00090 X=RND(1)*640
00100 Y=RND(1)*500
00110 PRINT A$;"Style",S
00120 PRINT A$;"Line =",X,Y
00130 NEXT A
00140 PRINT A$;"Clear"
00150 NEXT S
00170 END
```

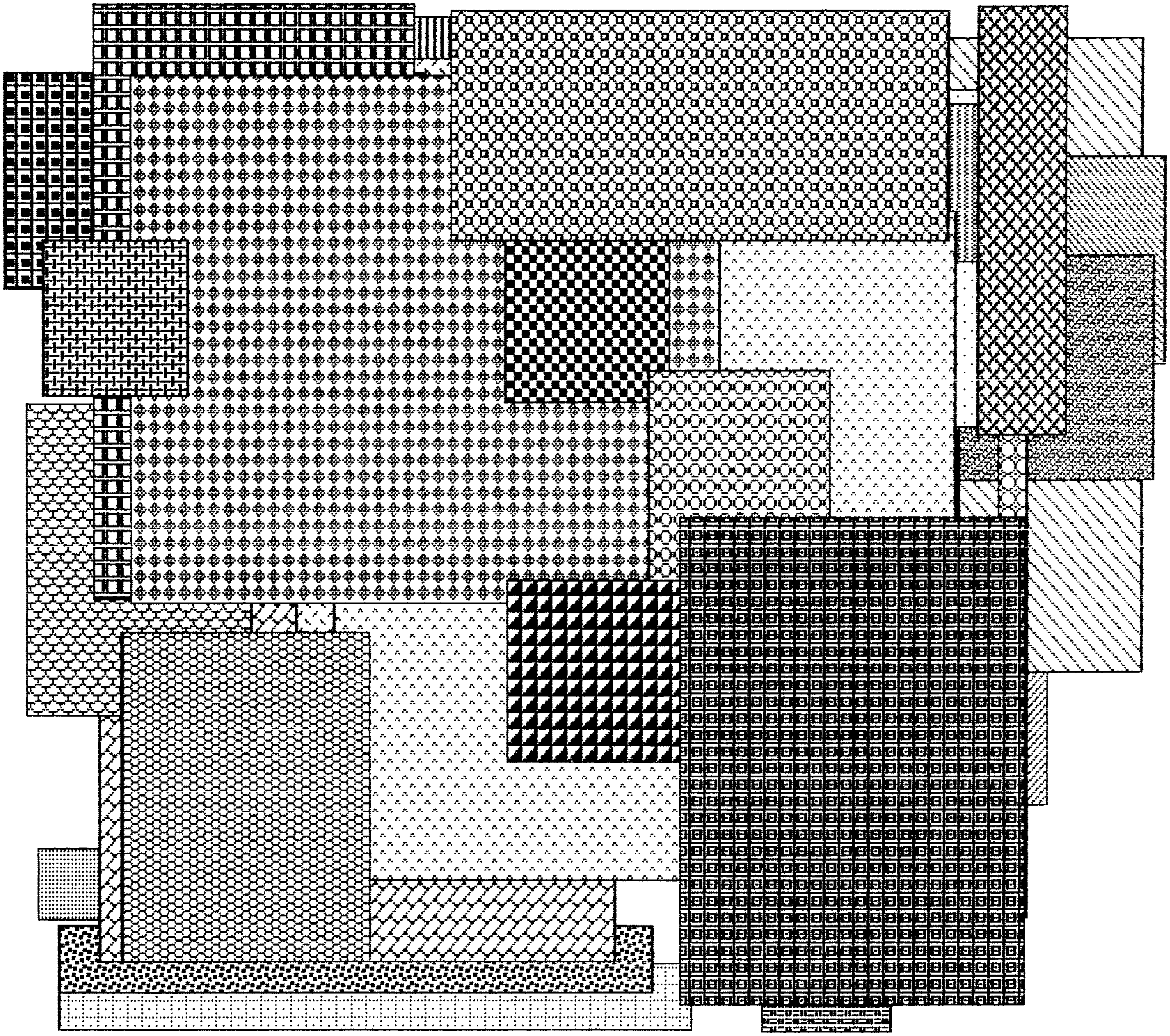
Boxes.BAS

```
00020 REM This program can not be used in page interlace mode.
00040 PRINT CHR$(27);"E";CHR$(27);"x1";CHR$(27);"x5"
00050 A$=CHR$(1)
00060 PRINT A$;"Reset:Clear:Origin Bottom:Mode +Pixel"
00070 P=1:P1=0
00080 FOR D1=450 TO 50 STEP -2
00090 S=S+1:IF S=50 THEN S=1
00100 X1=INT(RND(1)*640)
00110 X2=INT(RND(1)*640)
00120 Y1=INT(RND(1)*500)
00130 Y2=INT(RND(1)*500)
00140 IF ABS(X2-X1)>D1 THEN GOTO 100
00150 IF ABS(Y2-Y1)>D1 THEN GOTO 100
00160 GOSUB 210
00170 GOSUB 210
00180 NEXT D1
00200 END
00210 P=P+1:IF P=2 THEN P=0
00220 P1=P1+1:IF P1=2 THEN P1=0
00230 PRINT A$;"Display",P
00240 PRINT A$;"Draw",P1
00250 PRINT A$;"Mode +Erase"
00260 PRINT A$;"Style 0 0"
00270 PRINT A$;"Erase",X1,Y1,X2,Y2
00280 PRINT A$;"Box",X1,Y1,X2,Y2
00290 PRINT A$;"Style 0",S
00300 PRINT A$;"Mode -Erase"
00310 PRINT A$;"Erase",X1,Y1,X2,Y2
00320 PRINT A$;"Box",X1,Y1,X2,Y2
00330 RETURN
```

Lines.BAS



Boxes.BAS



Sine.BAS

```
00040 PRINT CHR$(27);"E";CHR$(27);"x1";CHR$(27);"x5"
00050 A$=CHR$(1)
00060 PRINT A$;"Reset:Clear:Mode +Pixel"
00070 FOR S=245 TO 5 STEP -5
00080 PRINT A$;"Go -300 0"
00090 P=P+.1
00100 FOR X=-300 TO 300 STEP 6
00110 Y=INT(COS(X+P)*S)
00120 PRINT A$;"Line =",X,Y
00130 NEXT X
00140 NEXT S
00160 END
```

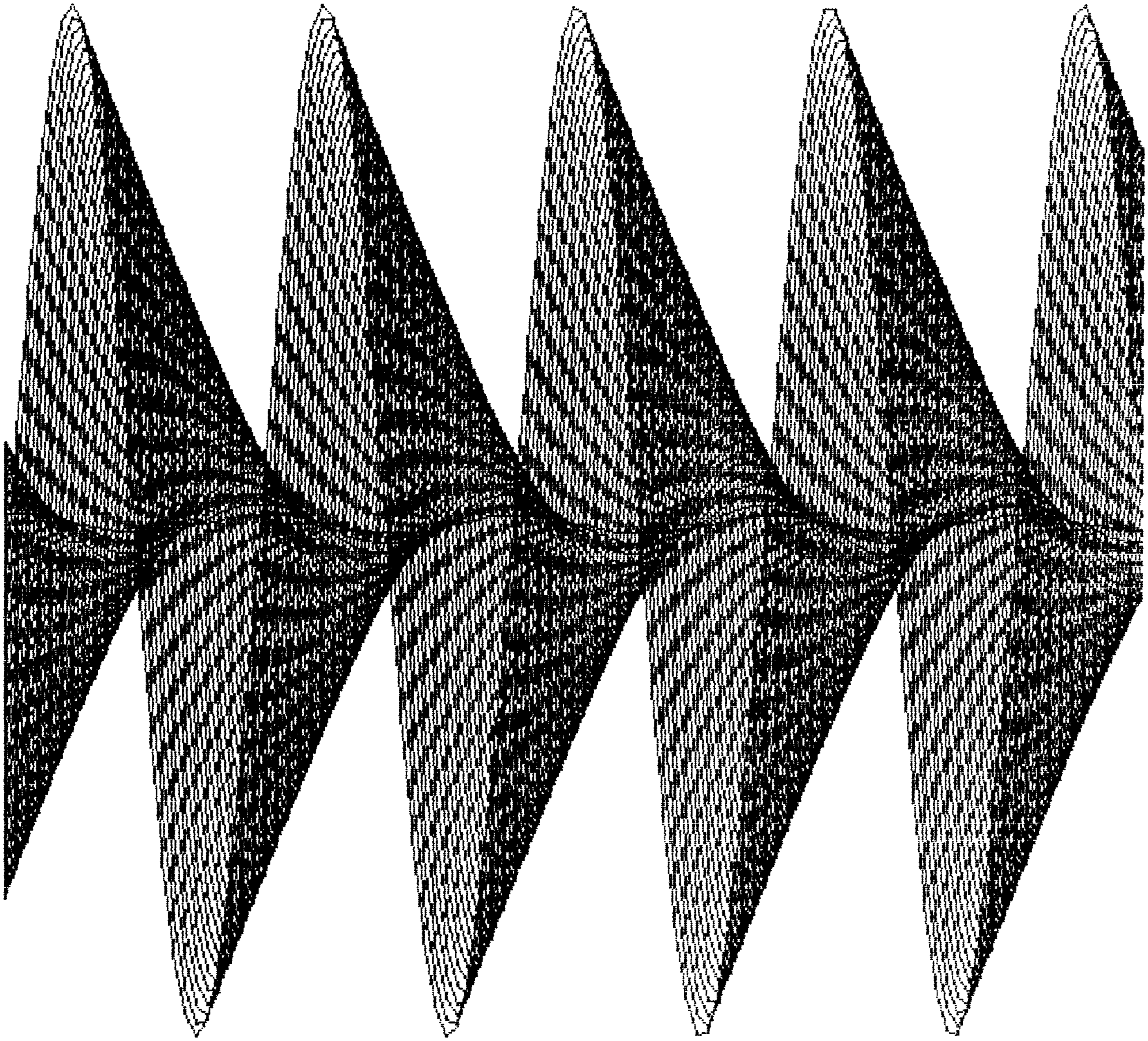
Circle1.BAS

```
00040 PRINT CHR$(27);"E";CHR$(27);"x1";CHR$(27);"x5"
00050 A$=CHR$(1)
00060 PRINT A$;"Reset:Clear:Mode +Pixel"
00070 FOR R=240 TO 5 STEP -5
00080 PRINT A$;"Circle",X1,Y2,R
00090 PRINT A$;"Circle",X2,Y2,R
00100 PRINT A$;"Circle",X1,Y1,R
00110 PRINT A$;"Circle",X2,Y1,R
00120 X1=X1-5:X2=X2+5
00130 Y1=Y1-5:Y2=Y2+5
00140 NEXT R
00160 END
```

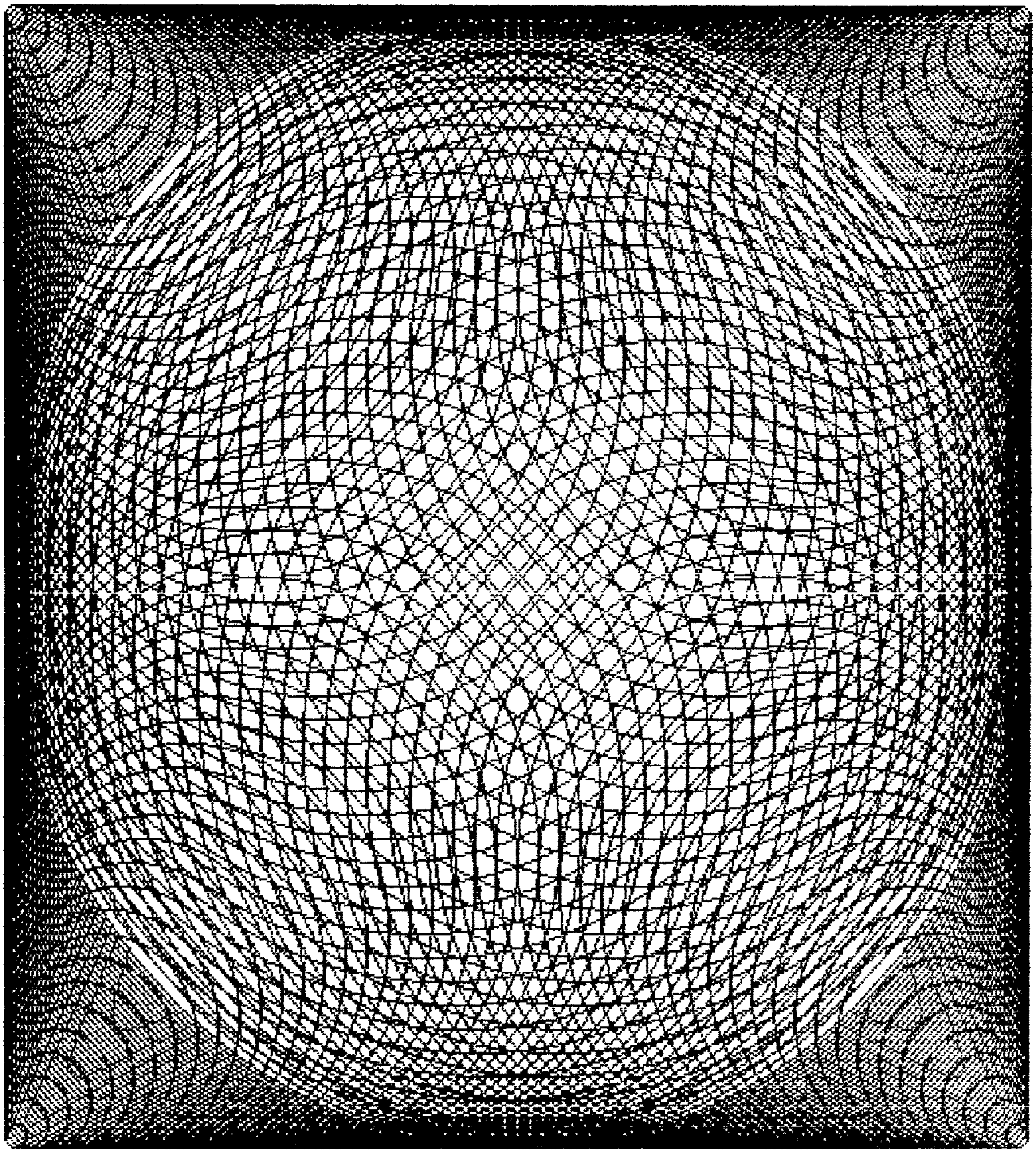
Circle2.BAS

```
00020 REM This program can not be used in page interlace mode.
00040 PRINT CHR$(27);"E";CHR$(27);"x1";CHR$(27);"x5"
00050 A$=CHR$(1)
00060 PRINT A$;"Reset:Clear:Mode +Pixel"
00070 R=10:G=15:P1=0:P2=1
00080 FOR L=1 TO 8
00090 FOR D=10 TO 360 STEP 10
00100 PRINT A$;"Display",P1
00110 PRINT A$;"Draw",P2
00120 P1=P1+1:IF P1=3 THEN P1=0
00130 P2=P2+1:IF P2=3 THEN P2=0
00140 PRINT A$;"Rotate",D
00150 PRINT A$;"Go 0 0"
00160 PRINT A$;"Circle #170 0",R
00170 NEXT D
00180 R=R+G
00190 G=G-2
00200 NEXT L
00220 END
```

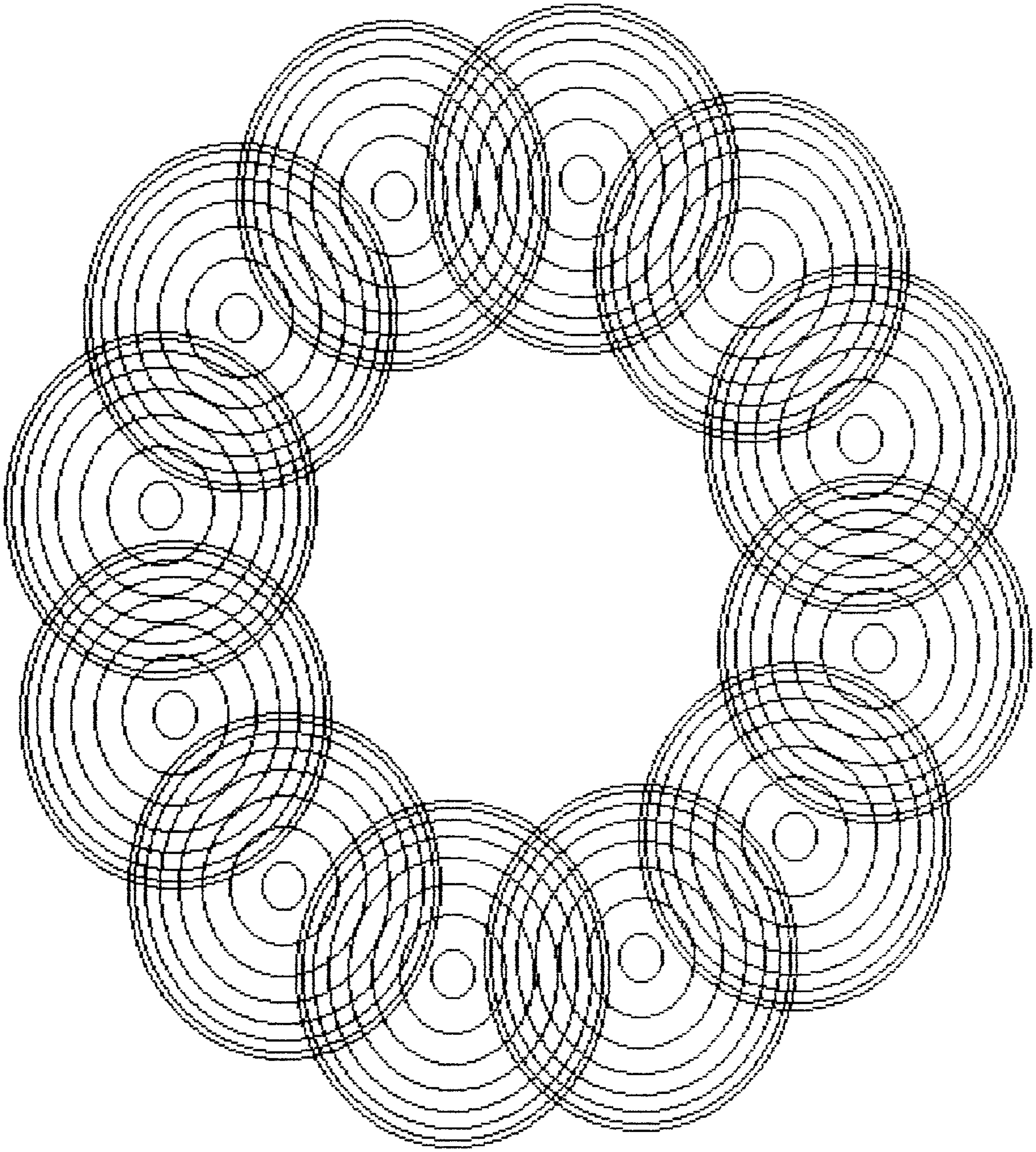
Sine.BAS



Circle1.BAS



Circle2.BAS



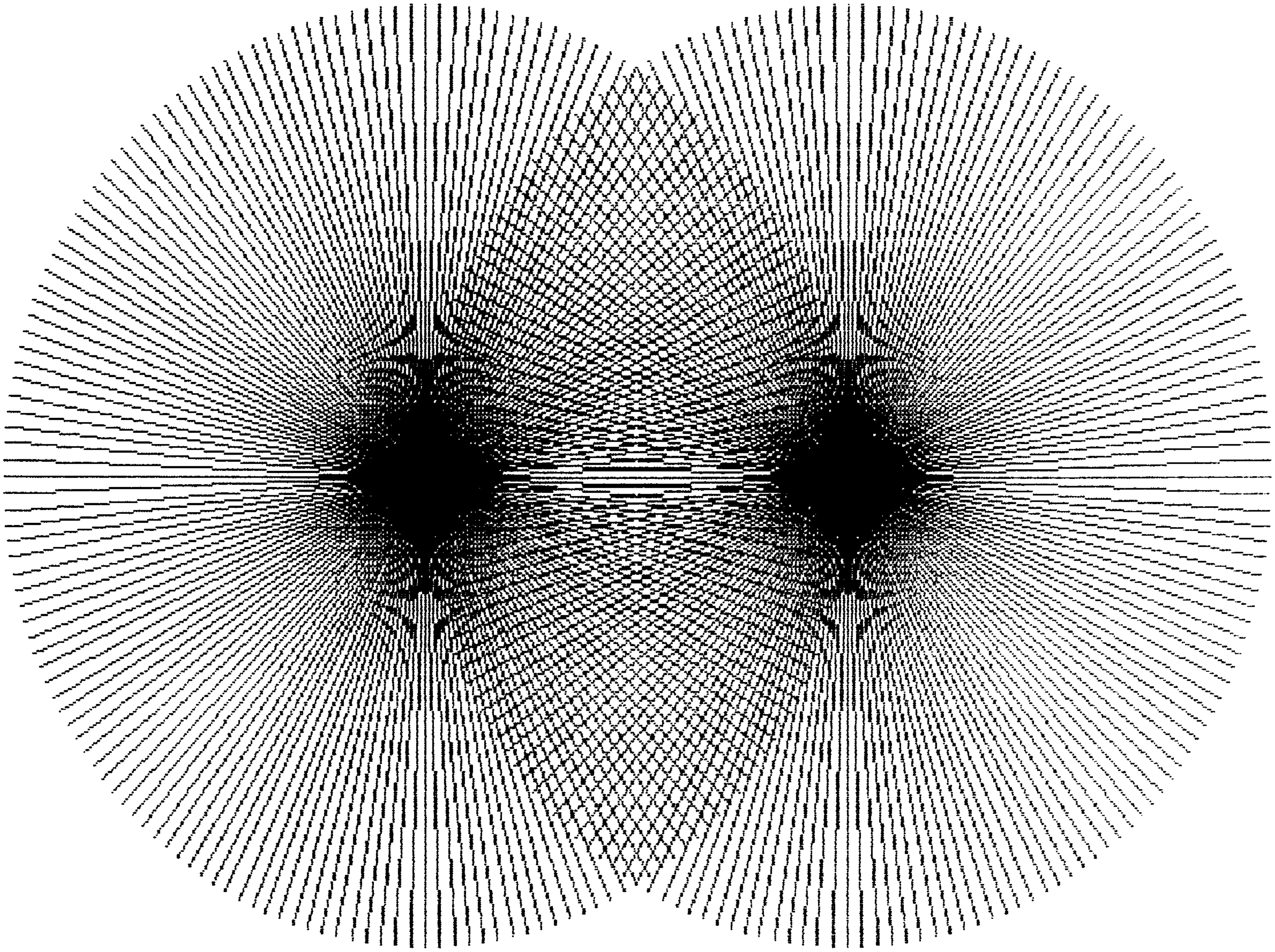
Radii.BAS

```
00030 PRINT CHR$(27);"E"
00040 A$=CHR$(1)
00050 PRINT A$;"Reset:Clear:Mode +Pixel"
00060 PRINT A$;"Origin Bottom"
00070 FOR A=2 TO 360 STEP 2
00080 PRINT A$;"Rotate",A
00090 PRINT A$;"Line 424 250 #210 0"
00100 PRINT A$;"Line 214 250 #-210 0"
00110 NEXT A
00130 END
```

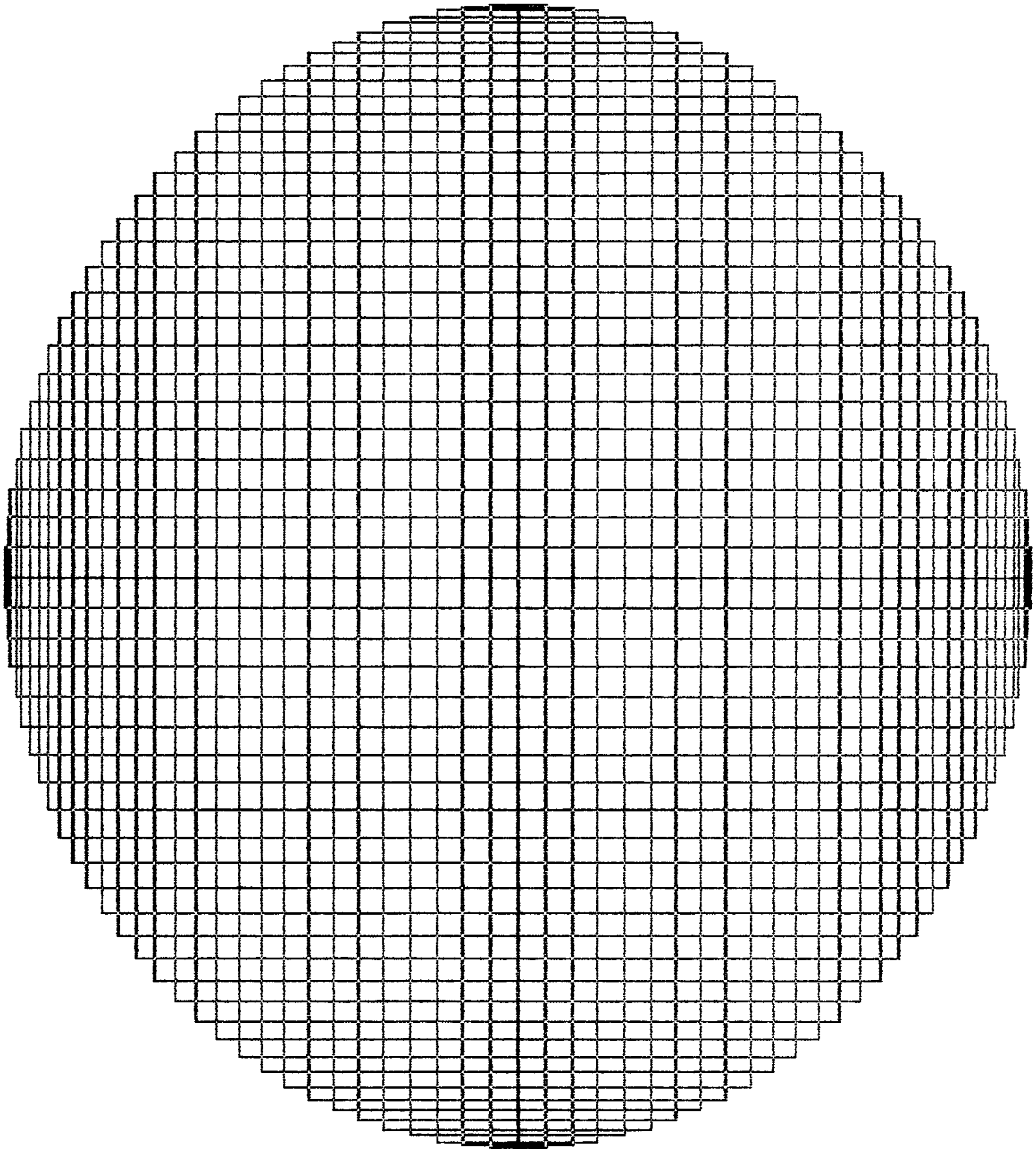
Rotate.BAS

```
00030 PRINT CHR$(27);"E"
00040 A$=CHR$(1)
00050 PRINT A$;"Reset:Clear:Mode +Pixel +Toggle"
00060 FOR A=3 TO 360 STEP 3
00070 PRINT A$;"Rotate",A
00080 PRINT A$;"Box 0 0 #160 160"
00090 NEXT A
00110 END
```

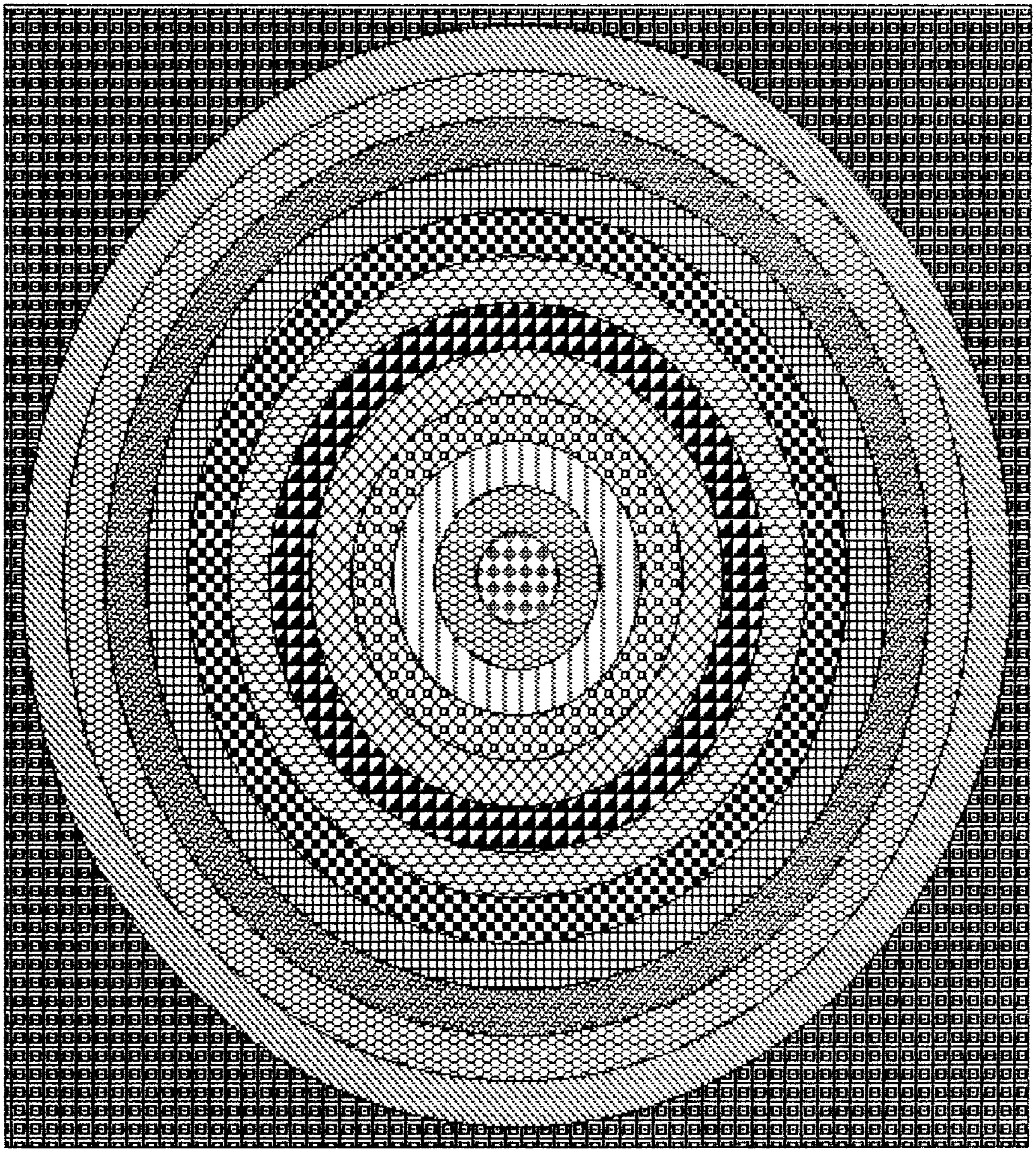
Radii.BAS



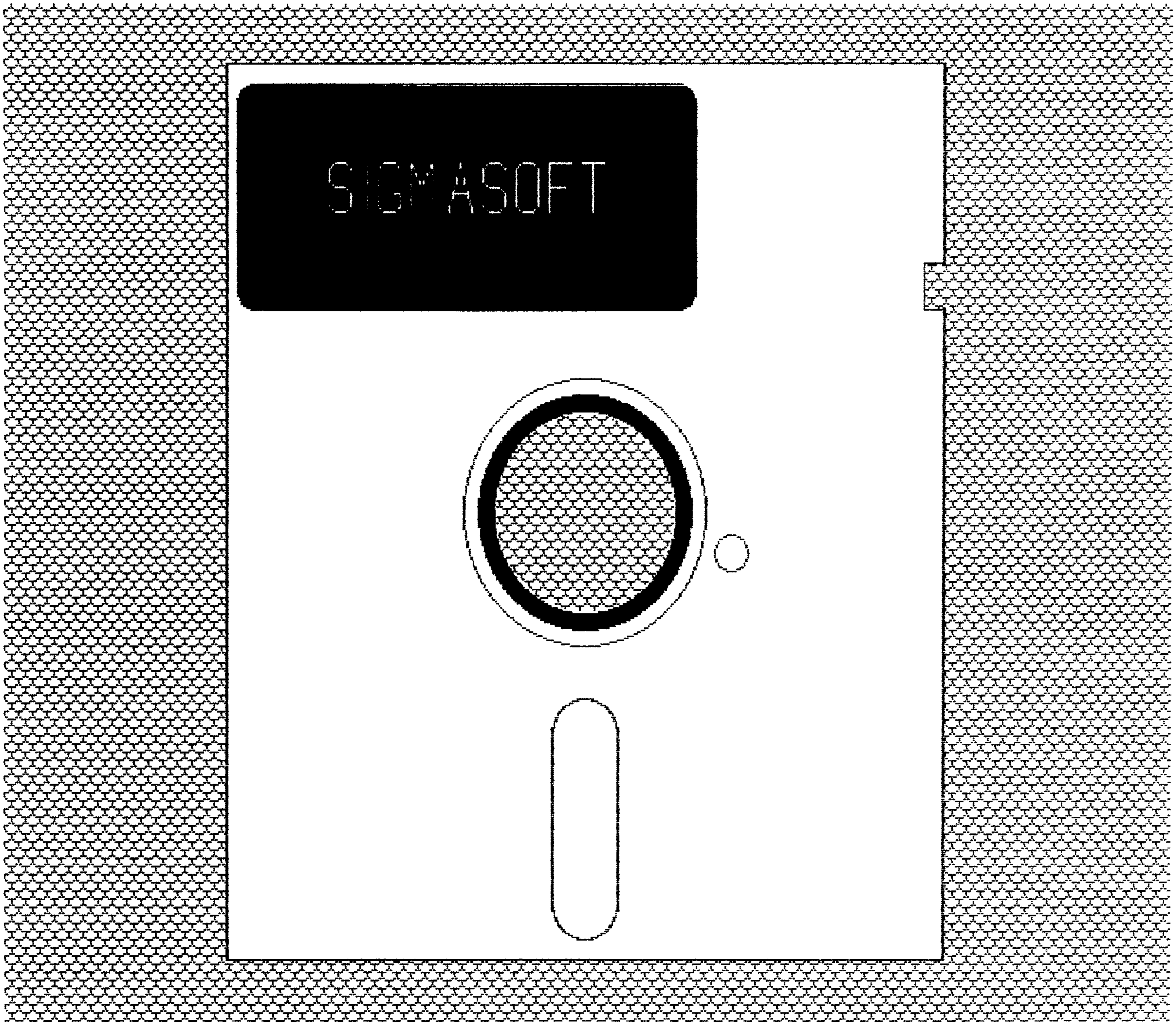
Rotate.BAS



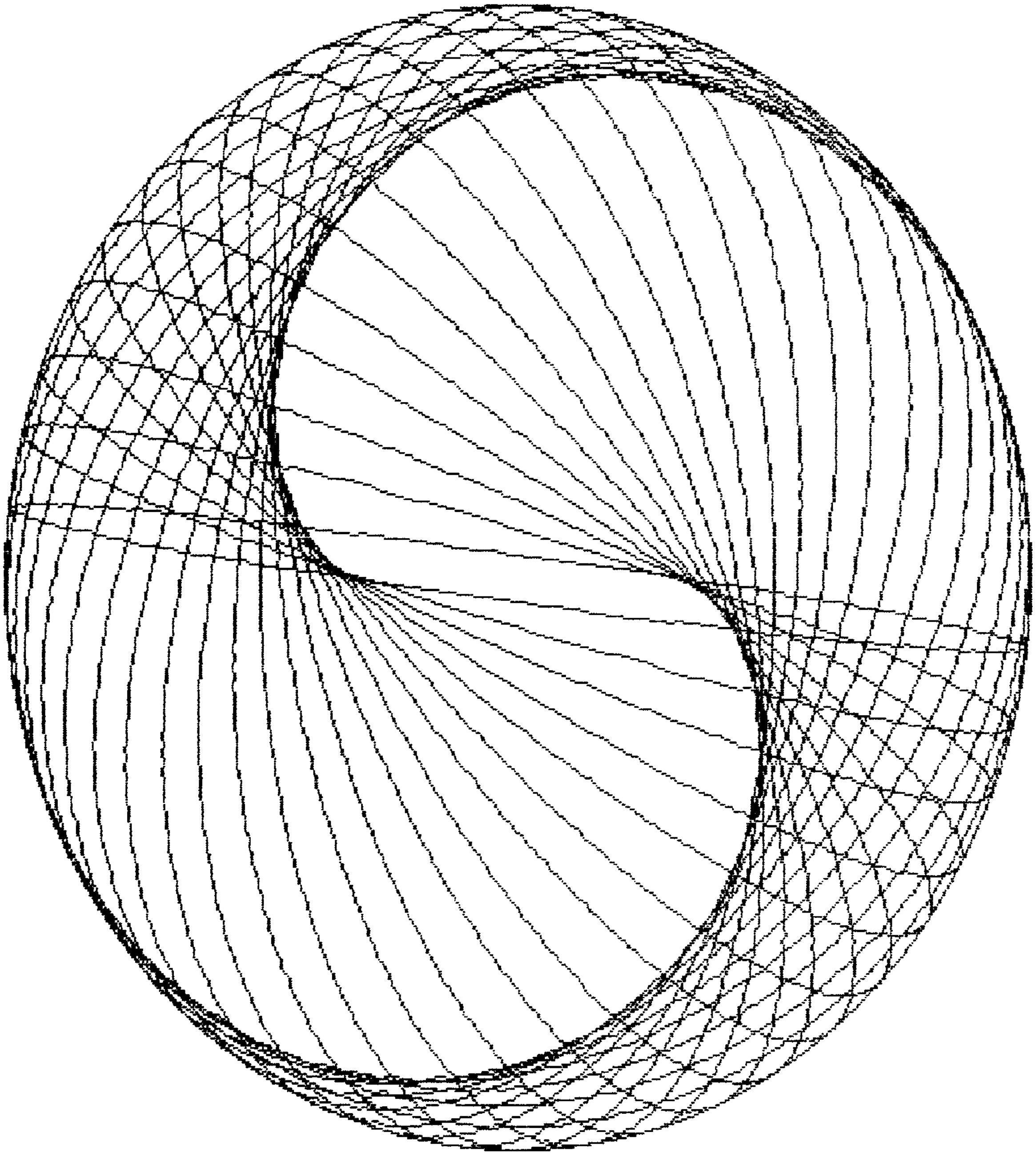
Demo.IGC



Demo.IGC



Demo.IGC



Letter1.IGC

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890/:.,%

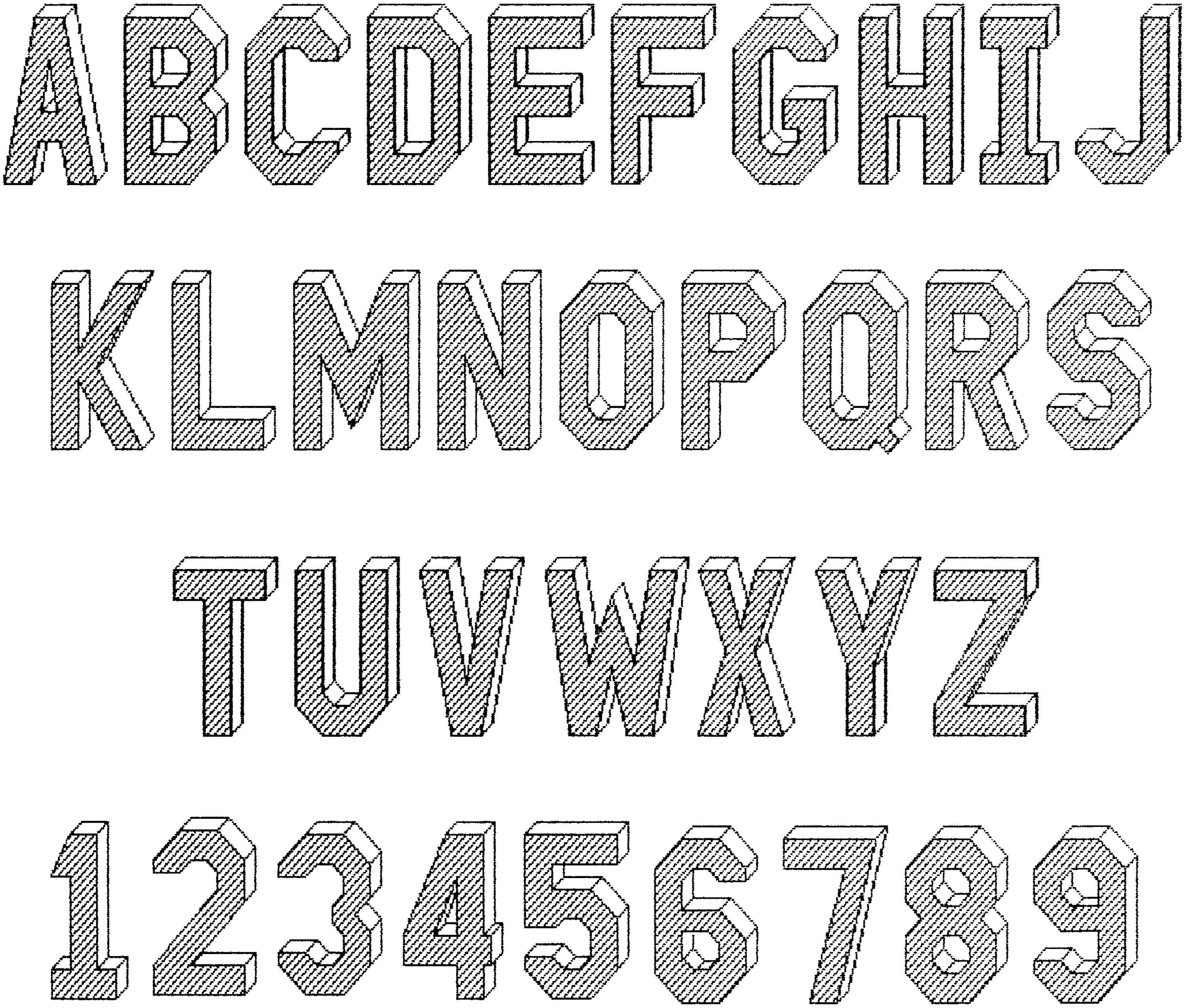
Letter2.IGC

ABCDEFGHIJ

KLMNOPQRST

UVWXYZ

Letter3.IGC



PD: and PDC:

The Pseudo Disk Device Driver

HDOS and CP/M Versions 2.Ø

Written by Darron J. Shaffer
and Clay D. Montgomery

Copyright (c) 1985 by

SigmaSoft and Systems
4488 Spring Valley, #1Ø7
Dallas, TX 75234

Introduction

The Pseudo Disk Device Driver is a floppy disk drive emulator designed especially for use with the Interactive Graphics Controller. PD: and PDC: (the HDOS and CP/M versions respectively) use the video memory available on the IGC device to store disk files in the standard format of the operating system involved.

Files can be read and written to/from the Pseudo Disk in exactly the same way other drive units in the system are used but in a fraction of the time required by mechanical drives.

Getting Started

The device drivers are provided on the IGC distribution software diskettes. One of these disks contains the HDOS version and another contains the CP/M version of the same software. Begin by copying the contents of the one you wish to use onto a system disk. Be sure to store these distribution diskettes in a safe place once they have been copied onto working system disks.

Performing a Set Operation

The Set utilities (Set.ABS under HDOS and Set.COM under CP/M) allow the user to configure various options available for a device driver. The Set utilities modify the driver by making patches in the code of the program and then writting the patched driver back to the device driver disk file, so that the change need not be made again. The PD: and PDC: device drivers support several of these options and they must be set properly for the driver to operate. An example of the syntax under HDOS would be:

```
>Set PD: Help
```

Under CP/M type:

```
A>Set PDC: Help
```

The values specified for a set operation may be given in several number bases by supplying the proper radix where decimal is the default. Each time a set operation is performed the set status table for all of the options will be displayed to verify that the desired change has been made.

The Help Option

The Help option will display the set status of all of the options to the system console without making any set changes to the driver. All of the values displayed will be in decimal, regardless of the number base they were originally set in.

The Bank and NoBank Options

These options allow the partitioning of the available IGC memory between the Pseudo Disk and other drivers you may wish to run simultaneously.

The maximum amount of memory that the IGC system supports is 256k bytes. This address range is divided into four banks, each of which has 64k bytes of capacity. However, you can only use these banks if they exist on your IGC board. Refer to the chart below for a summary of the available banks on your particular model.

Take care that you only enable banks which do actually exist on your IGC board. If non-existant banks are enabled, or if banks are enabled that are also used by other device drivers, data stored on the Pseudo Disk will be lost. Note that at least one bank must be enabled for the Pseudo Disk driver. Bank Ø is the default.

The IG: and IGC: graphics drivers always use bank Ø, and will also use bank 2 if bank interlace mode is selected (refer to the Inter set option of IG: and IGC:). These banks can be used by the Pseudo Disk if the graphics driver will not be used.

If you are uncertain about the memory size of your IGC board, run the IGCTEST utility and use the Control-B function to skip to the memory banking test. This test will display the memory size for you.

Summary of Available Memory Banks

Total IGC Memory Size	Available Memory Banks
64k	Bank Ø only.
128k	Either bank Ø, 2, or both.
256k	Any combination of banks Ø, 1, 2, or 3.

The Port Option

The Port set option defines the I/O port address of the Interactive Graphics Controller/Pseudo Disk. The port address value set here must correspond to the port address selected on the interface board. The standard port address for the IGC device is 8.

Set Option Summary for PD: and PDC:

Help	Print Set Option Summary With Status
BankØ	Enable Bank Ø for use by Pseudo Disk
NoBankØ	Disable Bank Ø for use by Pseudo Disk
Bank1	Enable Bank 1 for use by Pseudo Disk
NoBank1	Disable Bank 1 for use by Pseudo Disk
Bank2	Enable Bank 2 for use by Pseudo Disk
NoBank2	Disable Bank 2 for use by Pseudo Disk
Bank3	Enable Bank 3 for use by Pseudo Disk
NoBank3	Disable Bank 3 for use by Pseudo Disk
Port x	IGC Port Address (8 Standard)

Loading the Device Driver

Under CP/M the PDC: driver must be loaded into the system using the Loadd.COM utility provided. The PD: driver under HDOS will load automatically when mounted. For HDOS type:

```
>MOUNT PD:
```

Under CP/M type:

```
A>LOADD PDC:
```

If you get an error, it will most likely be because either the PD.DVD (HDOS) or PDC.DVD (CP/M) file is not present on the system disk. Error messages should be self explanatory.

Note that you may be prompted about space being available above the BIOS with CP/M. If you are unfamiliar with this, simply hit return. Refer to the section on the Loadd.COM utility for a complete explanation of this.

Initialization

When the Pseudo Disk driver is loaded, it will ask if you wish to initialize PD: (format unit M: under CP/M). If this is the first time the Pseudo Disk driver has been loaded since the computer was powered up, then you must respond with a 'Yes' to erase the memory area and create a directory on the Pseudo Disk for file storage.

Once initialized, the Pseudo Disk is ready to store files. Files which are written to the Pseudo Disk will remain intact even during cold boots, so long as power to the computer is not turned off.

When the system is rebooted, simply repeat the proceedure for loading the PD: or PDC: driver. If you wish for the files on the Pseudo Disk to remain intact, be sure to respond 'No' to the initialization prompt.

Copying files to and from the Pseudo Disk

Under HDOS the PDØ: device functions in just the same way as the SYØ: and the DKØ: devices. With CP/M the PDC: device becomes M: similar to the units A: and B:. For HDOS type:

```
>CAT PD:
```

Under CP/M type:

```
A>DIR M:
```

Any copy utility such as PIP can be used to transfer files to and from the Pseudo Disk. An example of this under HDOS is as follows:

```
>PIP PD:=PIP.ABS
```

For CP/M type:

```
A>PIP M:=PIP.COM
```

Under HDOS the PD: driver can be Sysgened as a bootable device using the Sysgen.ABS and Boot.ABS utilities provided with HDOS. Simply mount the PD: device and run Sysgen specifying PD: as the destination device. The Boot.ABS utility will then reboot the system from PD:. Type:

```
>BOOT PD:
```

SP: and SPC:

Spooling Printer Device Driver

HDOS and CP/M Versions 1.0

Written by Clay D. Montgomery
and Darron J. Shaffer

Copyright (c) 1985 by

SigmaSoft and Systems
4488 Spring Valley, #107
Dallas, TX 75234

Introduction

The Spooling Printer device driver is a general purpose printer driver designed especially for use with the Interactive Graphics Controller. The IGC memory can be utilized to store the spooled data or the spooling can be disabled if the IGC memory is not available.

This driver is compatible with virtually any type of printer and printer interface. Versions are provided for HDOS, CP/M, parallel, and serial interfaces.

Character Set

SP:/SPC: supports the entire ASCII character set. All printable and non-printable control characters will be passed to the printer with the parity bit intact. This is to prevent interference with any of the special features of a printer that the user may wish to access by passing control codes.

Getting Started

The device drivers are provided on the IGC software distribution diskettes. One of these disks contains the HDOS version and another contains the CP/M version of the same software. Begin by copying the contents of the one you wish to use onto a system disk. Be sure to store these distribution diskettes in a safe place once they have been copied onto working system disks.

Two versions of the spooling driver are provided on each diskette. One version is for use with a parallel printer interface and the other is for a serial printer interface. Refer to the chart below to determine the file name for the version of the driver that matches your hardware system. Then, rename this file to eliminate the numeral. For example, if you wish to use a parallel interface under HDOS, copy the file named SP1.DVD to a file named SP.DVD on your HDOS system disk.

The working copy of this driver must be named SP.DVD on your HDOS system disk and SPC.DVD on your CP/M system disk. The other versions of the drivers will not be needed unless you ever change your printer interface type.

File Names for the Different Versions of SP:/SPC:

Interface	HDOS Name	CP/M Name
Parallel	SP1.DVD	SPC1.DVD
Serial	SP2.DVD	SPC2.DVD

Once you have selected the proper version of SP: or SPC: for your system, the next step is configuration. These drivers have a number of important set options that must be configured properly before use.

Performing a Set Operation

The Set utilities (Set.ABS under HDOS and Set.COM under CP/M) allow the user to configure various options available for a device driver. The Set utilities modify the driver by making patches in the code of the program and then writting the patched driver back to the device driver disk file, so that the change need not be made again. The SP: and SPC: device drivers support many of these set options. An example of the syntax under HDOS would be:

```
>Set SP: Help
```

Under CP/M type:

```
A>Set SPC: Help
```

Although most of these set options are optional, several control the manner in which the driver communicates with the printer. These options must be set properly for the driver to operate.

The values specified for a set operation may be given in several number bases by supplying the proper radix where decimal is the default. Each time a set operation is performed the set status table for all of the options will be displayed to verify that the desired change has been made.

The Help Option

The Help option will display the set status of all of the options to the system console without making any set changes to the driver. All of the values displayed will be in decimal, regardless of the number base they were originally set in.

The Form and NoForm Options

The Form and NoForm set options set and reset the form flag. If this flag is set the printer will advance to the top of the next unused form when the driver is closed after a print operation.

Skip and NoSkip

The Skip and NoSkip set options set and reset the skip flag. If this flag is set, the driver will skip all redundant form feeds. If nothing has yet been printed on a form, then the driver will ignore any form feeds that might occur, which would normally cause forms to be skipped blank. This feature can be very useful while printing a file that has already been paged with form feed characters.

The PPI and NoPPI Options

The PPI and NoPPI set options set and reset the PPI (8255 type Programmable Peripheral Interface) flag. This option is not available with serial versions of the SP: and SPC: drivers.

This flag should only be set if you are using the Heath/Zenith Multifunction I/O Card (Z-89-11) or a similar board to interface your printer. This flag must not be set (NoPPI) if you are using one of the Centronics interfaces on the SigmaSoft Universal Parallel Interface Board.

The CTS and NoCTS Options

The CTS and NoCTS set options set and reset the CTS (Clear To Send) when low flag. This option is not available with parallel versions of the SP: and SPC: drivers.

This flag should only be set if you are using the non-standard H/Z14 printer cable to connect the serial I/O board to your printer. This flag must not be set (NoCTS) if you are using the standard DSR (Data Set Ready) high serial cable.

The Bank and NoBank Options

These options allow the partitioning of the available IGC memory between the Spooler and other drivers you may wish to run simultaneously.

The maximum amount of memory that the IGC system supports is 256k bytes. This address range is divided into four banks, each of which has 64k bytes of capacity. However, you can only use these banks if they exist on your IGC board. Refer to the chart below for a summary of the available banks on your particular model.

Take care that you only enable banks which do actually exist on your IGC board. If non-existant banks are enabled, or if banks are enabled that are also used by other device drivers, data stored in the Spooler will be lost. Note that this driver can be used without any IGC memory banks if you so desire.

By disabling all four banks for this driver it will assume non-spooling mode in which it functions as a regular printer driver. This mode must be used to print graphics with a 64k IGC board.

The IG: and IGC: graphics drivers always use bank Ø, and will also use bank 2 if bank interlace mode is selected (refer to the Inter set option of IG: and IGC:). These banks can be used by the Spooler if the graphics driver will not be used.

If you are uncertain about the memory size of your IGC board, run the IGCTEST utility and use the Control-B function to skip to the memory banking test. This test will display the memory size for you.

Summary of Available Memory Banks

Total IGC Memory Size	Available Memory Banks
64k	Bank Ø only.
128k	Either bank Ø, 2, or both.
256k	Any combination of banks Ø, 1, 2, or 3.

Page x

The Page set option determines the count of lines per page that will be printed before skiping the form perforation. Also, note that zero can be set as the lines per page count to effectively disable paging.

The Ports Option

The Ports set option defines the I/O port addresses of both the IGC and printer devices. The first value is the port address of the Interactive Graphics Controller and the second value is the port address of the printer. The port address values set here must correspond to the port address values of the particular interface boards being used. The standard port address for the IGC device is 8. The standard port address of the first Centronics interface on the Universal Parallel I/O Board is 15. The standard serial printer port address of the H/Z89 is 224 decimal (340 octal).

The Ready Option

The Ready set option is for use in configuring the handshaking of the driver with the printer through a Centronics parallel interface. This option is not available with serial versions of the SP: and SPC: drivers.

When using the SigmaSoft Universal Parallel Interface with a standard Centronics cable, these values should be set to 30, 24. If you are instead using an 8255 type Programmable Peripheral Interface (such as the Heath/Zenith Multifunction I/O Card or a similar board) to interface your printer, these values should be 128, 128.

The first value is used as a Boolean And mask to selectively clear bits that are to be ignored in the input status byte. The second value is then used as the binary compare pattern to determine if the printer is ready to receive the next data byte.

A full Centronics interface will have four or five bits which collectively indicate the printer status (Busy, error, paper out, etc.). This system allows the user to configure the device handshaking to monitor any or all of the condition bits of the input status byte. Refer to the chart below for an example of calculating the Ready set values for a particular system.

Ready Set Option Values Calculation Example

Bit	Value	Ready Pattern	Indicates
0	1	-	Data Acknowledge
1	2	0	Printer Not Busy
2	4	0	Paper Not Empty
3	8	1	Device Selected
4	16	1	No Printer Error
5	32	-	Undefined
6	64	-	Undefined
7	128	-	Undefined

Bits 1,2,3, and 4 should be monitored, so And mask would be 00011110 binary (30 decimal) to mask all other bits of status byte clear. The logical And result is then compared to the ready value (second set option value) which would be 00011000 binary (24 decimal).

The Baud Option

The Baud set option is for use in configuring the serial data baud rate between the serial interface and the printer. This option is not available with parallel versions of the SP: and SPC: drivers.

This baud rate should correspond to the baud rate selected on the internal switches of the printer you are using. Note that baud rates above 1200 may be unreliable on some printers when printing high resolution graphics, due to limitations of their serial interface boards. For this reason SigmaSoft strongly recommends the use of a parallel printer interface if you are going to be printing graphics.

The Init Option

The Init set option determines the 16 byte printer initialization sequence which is output to the printer device each time a print operation is performed.

This data should contain any of the control characters or escape sequences that are required by your printer to initialize. This option can also be used to configure the printer for any special default modes that are desired. Refer to the operations manual for your printer to determine what modes are available. The following chart list the recommended values for this option for several different printer types. All unused values of this sequence should be set to zero.

Recommended Values for the Init Option

Values	Printer Type
27,64	Epson/Star Gemini/Panasonic/Riteman/Mannesman Tally
17,2,5,6,29	Integral Data Systems/Paper Tiger/Micro Prism
17,27,62	C. ITOH/Prowriter
17,29	Okidata/Microline/Okigraph

Set Option Summary for SP: and SPC: (Parallel Printer Interface)

Help	Print Set Option Summary With Status
Form	Form Feed Printer After a Print Command
NoForm	No Form Feed After a Print Command
Skip	Skip redundant form feeds.
NoSkip	Pass all form feeds.
PPI	Printer Interface is 8255 Type PPI
NoPPI	Printer Interface is SigmaSoft Type Centronics
BankØ	Enable Bank Ø for use by Spooler
NoBankØ	Disable Bank Ø for use by Spooler
Bank1	Enable Bank 1 for use by Spooler
NoBank1	Disable Bank 1 for use by Spooler
Bank2	Enable Bank 2 for use by Spooler
NoBank2	Disable Bank 2 for use by Spooler
Bank3	Enable Bank 3 for use by Spooler
NoBank3	Disable Bank 3 for use by Spooler
Page x	Count of lines per page.
Ports x,x	IGC and Printer Port Addresses (8, 15 Standard)
Ready x,x	Printer Device Handshaking (3Ø, 24 Standard)
Init x,x,x,...	16 Byte Printer Initialization Sequence

Set Option Summary for SP: and SPC: (Serial Printer Interface)

Help	Print Set Option Summary With Status
Form	Form Feed Printer After a Print Command
NoForm	No Form Feed After a Print Command
Skip	Skip redundant form feeds.
NoSkip	Pass all form feeds.
CTS	Printer handshaking is Clear To Send when Low.
NoCTS	Printer handshaking is Data Set Ready when High.
BankØ	Enable Bank Ø for use by Spooler
NoBankØ	Disable Bank Ø for use by Spooler
Bank1	Enable Bank 1 for use by Spooler
NoBank1	Disable Bank 1 for use by Spooler
Bank2	Enable Bank 2 for use by Spooler
NoBank2	Disable Bank 2 for use by Spooler
Bank3	Enable Bank 3 for use by Spooler
NoBank3	Disable Bank 3 for use by Spooler
Page x	Count of lines per page.
Ports x,x	IGC and Printer Port Addresses (8, 224 Standard)
Baud x	Printer Serial Interface Data Baud Rate
Init x,x,x,...	16 Byte Printer Initialization Sequence

Loading the Device Driver

Under CP/M the SPC: driver must be loaded into the system using the Loadd.COM utility provided. The SP: driver under HDOS will load automatically when used. To manually load the driver under HDOS type:

```
>LOAD SP:
```

Under CP/M type:

```
A>LOADD SPC:
```

If you get an error, it will most likely be because either the SP.DVD (HDOS) or SPC.DVD (CP/M) file is not present on the system disk. Error messages should be self explanatory.

Note that you may be prompted about space being available above the BIOS with CP/M. If you are unfamiliar with this, simply hit return. Refer to the section on the Loadd.COM utility for a complete explanation of this.

Printing Text Files with the Spooler

Under HDOS the SP: driver is a direct replacement for the standard LP: device. With CP/M the SPC: driver replaces the standard printer routine in the BIOS and becomes the system list device (LST:). This insures compatibility with application software and provides spooling capability to any program that uses these standard devices for printer output.

ASCII text files can be output to the printer using the Spooler with the standard PIP utility. For HDOS type:

```
>PIP SP:=DEMO.IGC
```

For CP/M type:

```
A>PIP LST:=DEMO.IGC,EOF:
```

Note that the files you copy to the Spooler must be smaller than the amount of memory allocated to the driver or the Spooler will stop and wait for the space to become available. Once the printer has accepted enough of the data, control will be returned to the system.

An important feature of the Spooler is its ability to remove its own interrupt vector when it is closed. This means that the Spooler only uses CPU processing time when it actually has data to print. However, this does require that the spooler be properly closed after a write operation. Under CP/M the driver has to be closed manually by appending the EOF: specification after the file name as shown in the above example.

Printing High Resolution Graphics

The IG:/IGC: graphics driver also uses the spooler for printer output to process a Print command. This provides spooling capability to graphics printing and reduces the size of the graphics driver. However, this requires that the spooler driver be resident in memory when a Print command is sent to the IG:/IGC: graphics driver. If the Spooler is not resident, the Print command will not be performed.

Before a graphics dump can be printed the SP:/SPC: driver must be manually loaded into memory. Under HDOS this can be done either temporarily or permenantly. For example:

```
>COPY SP:               (Temporary load for HDOS)
>LOAD SP:               (Permenant load for HDOS)
```

Under CP/M type:

```
A>LOADD SPC:
```

With the Spooler loaded, the graphics dump can now be performed by sending a Print command to the graphics driver. For HDOS type:

```
>COPY IG:=TT:
PRINT
(Control D)
```

Under CP/M type:

```
A>PIP PUN:=CON:
PRINT
(Control Z)
```

Aborting and Duplicating Print Operations

The Spooler driver features two commands for aborting a spooled printout, and making multiple copies. These commands are Stop and Copy respectively. The IGC software distribution disks contain two files which consist of these commands to simplify their use. The syntax for these commands is [STOP] and [COPY]. If these files are present on your system disk you can abort a spooled print operation under HDOS by simply typing:

```
>COPY SP:=STOP
```

Under CP/M type:

```
A>PIP LST:=STOP
```

The Copy command places the Spooler into multiple copy mode. In this mode, the driver will print continuous copies of the contents of the spooler buffer without the need to rewrite the data to the spooler.

To use this feature the Copy command must be given to the driver first, followed by the data to print. When the first copy is completed the driver will automatically begin another. This process will continue until the driver receives a Stop command as shown above. To enable multiple copy mode under HDOS type:

```
>COPY SP:=COPY
```

Under CP/M type:

```
A>PIP LST:=COPY
```

LOADD.COM

Device Driver Load Utility for CP/M

Version 2.Ø

Copyright (c) 1985 by

SigmaSoft and Systems
4488 Spring Valley, #1Ø7
Dallas, TX 75234

Introduction

The Loadd utility is used to install SigmaSoft device drivers into a CP/M 8Ø system. This utility dynamically allocates space from the Transient Program Area of the CP/M system and reserves it for the installed driver module. The principal advantage of this technique is ease of use. Whereas, in a standard CP/M system the I/O drivers and other continually resident programs must be written and assembled into the system BIOS, the Loadd utility provides the means to load device driver modules of the same capability with one simple command.

Theory

The Loadd utility begins by reading the specified driver file into memory and testing the code for validity. If the file does contain a valid driver then a search of the CP/M system is made to locate a possible matching driver which has already been loaded. If a matching driver is found, then the Loadd utility will request the resident driver to unload, so that the new version may be loaded in its place. Only if the resident driver responds to the unload request will the reload occur. This is to provide the capability to reload a driver which may have had set changes made while at the same time protect drivers which may be processing interrupts.

If a matching driver is not already present in the system, the user will be prompted for a location to load the new driver. This location can either be 2Ø48 bytes below the BDOS (top of the TPA), or above the system BIOS. This second option assumes that 64k of memory is installed in the computer and that the BIOS has been moved down to make room for the device driver module.

The Load Utility then uses a relocation table at the end of the device driver file to modify the driver code so that it will execute at the calculated load address. The modified driver code is then moved into place at the load address and called to sign on.

If the driver is requested to load below the CCP, two important patches are made to the CP/M system to reserve this memory area. The first is to the BDOS pointer in the System Parameter Area and the second is to the portion of the warm boot code in the BIOS which initializes this pointer.

Loading Device Driver Modules

Before any of the functions or features of a device driver can operate, the device driver must first be loaded into the CP/M system from the corresponding driver file using the Load utility. Examples of the syntax for doing this are:

```
A>Loadd IGC:

A>B:Loadd B:PDC:
```

The proper driver name to type following the load command will depend on the particular device driver you wish to load. Note that the driver file name specified to load must have the .DVD device driver file name extension which is the default.

The load utility will display a table of all of the currently loaded device driver modules showing their names, sizes, and load locations. The load locations can either be below the CCP (the default) or above the system BIOS. If this is the first time this driver has been loaded you will be asked if there is space available above the BIOS. If not, simply hit return and the driver will be loaded below the CCP.

Once loaded, the driver will be operational and will remain locked into the CP/M system until the next cold boot so that it need not be reloaded. Most drivers can be reloaded without rebooting so that set changes will take immediate effect. If a driver can not be reloaded, an appropriate error message will be displayed.

Multiple drivers of different names can be loaded simultaneously into the CP/M system. However, if the drivers conflict, (such as two printer drivers of different names configured for the same printer) only the last driver loaded will be active for that device.

Allocating Space Above the BIOS

The standard technique of installing device drivers below the CCP has two drawbacks. First, the CCP is locked into place removing 2k bytes of memory from the system. Second, installing drivers below the CCP is somewhat BIOS version dependent and so may not be possible with some non-standard CP/M implementations. If this is the case with the CP/M system you are using, a BIOS Incompatibility Error message will be displayed when you attempt to load a driver below the CCP.

Allocating space for a driver above the BIOS is a simple process which eliminates these problems. First, attempt to load the driver below the CCP so that the load utility will show you the size of the driver. This size specification is given in bytes and must be converted to a kilobyte count for the MOVCPM utility. Round the size given up to the nearest multiple of 1024 and drop the three right digits to convert the number to a kilobyte value as shown in the chart below.

Byte Size to Kilobyte Size Conversion Chart

Size in Bytes	Kilobyte Size	System Size
1 through 1024	1	63
1025 through 2048	2	62
2049 through 3072	3	61
3073 through 4096	4	60
4097 through 5120	5	59
5121 through 6144	6	58
6145 through 7168	7	57
7169 through 8192	8	56
8193 through 9216	9	55
9217 through 10240	10	54
10241 through 11264	11	53
11265 through 12288	12	52
12289 through 13312	13	51
13313 through 14336	14	50
14337 through 15360	15	49
15361 through 16384	16	48

Run the MOVCPM utility supplied with your CP/M software and specify the system size to be 64k minus the kilobyte size as shown in the above chart. For example, the PDC: driver is less than 1024 bytes in size so the command would be:

```
A>MOVCPM 63
```

The MOVCPM utility will create the new CP/M system in memory only. Use your CP/M SYSGEN utility to copy it to a disk. Type:

```
A>SYSGEN
```

Note that the names of the MOVCPM and SYSGEN utility may vary depending on the type of CP/M system you are using.

Driver Loading Errors

The process of loading a device driver is successful as long as an error message was not printed. If any error message appears then the driver will not be active and none of the functions of that driver can be used until a successful load operation is completed.

Introduction

Programmers experienced with graphics and Assembly Language may wish to develop custom software which communicates with the IGC directly rather than through a device driver. The following section is provided to help in this endeavor. However, this information is not required to use the IGC due to the capabilities of the device driver software that is provided with the IGC.

The primary advantage of direct communication with the IGC is greater speed and more effective memory usage for specialized applications. In an effort to encourage independent software development, SigmaSoft offers the source codes to the provided device drivers on diskette. The source to the graphics driver can be particuarly helpful in the development of custom graphics programs since it contains an extensive set of drawing primitives.

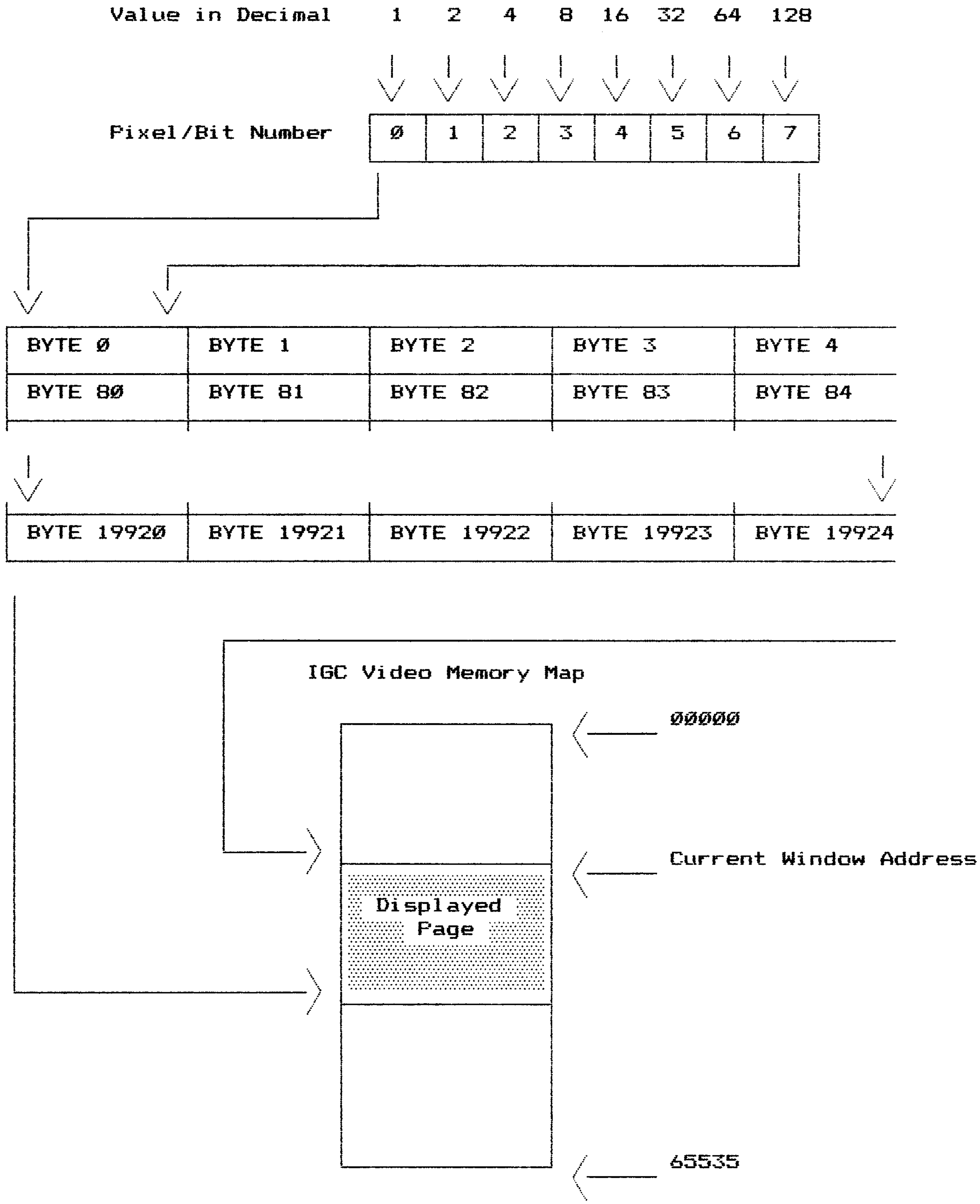
Video Memory Organization

The IGC consist of either 1, 2, or 4 banks of independent random access memory which is addressed by an array of counters and descrete logic through a set of eight parallel register/ports. It is through the first six of these ports that addresses, video memory data, and control data are accessed. The additional two ports are used on the interface board as Centronics interfaces.

When the pixel video output is enabled, the first bank is used to generate the even field frames. The odd field frames, if an interlace mode is selected, can be generated from either the second bank of IGC memory (bank interlace) or from the same first bank at an address 20000 bytes higher than the current even field frame address (page interlace). The second bank (if present) can only be displayed as the odd field in bank interlace mode. The third and fourth banks (if present) can never be displayed as video.

Data stored in the first IGC memory bank will be displayed starting at the current window address and ending at an address 20000 bytes higher. From the window address each byte is displayed with the least significant bit to the left and the most significant bit to the right. Each scan line is composed of 80 bytes of data. Frames consist of 240 scan lines when the 25th line of the H/Z19 is off and 250 scan lines when it is on. Note that this also affects the starting address of the odd field frame in page interlace mode. Below is a diagram to summarize.

The Organization of the IGC Video Memory (Non-interlace)



The I/O Address Registers

The 16 bits of address needed for the 64k of video memory is provided through the I/O address registers. The low order byte of the desired address is output to the Low I/O Address port and the high order byte is output to the High I/O Address port. These two registers collectively form the 16 bit IGC memory I/O address. Both of these registers are cleared by a hardware master reset.

The Video Memory Data I/O Register

Data can be either read or written from/to the addressed memory location through the Data I/O port. Each time a write operation is performed to this register, the I/O Address registers are automatically post incremented by 1. This allows contiguous blocks of video data to be moved without repeated addressing. Similarly, Read operations from the Data I/O port also post increment the I/O address registers, but only if enabled through the control register.

The Window Address Registers

The 16 bit Window Address is programmed similar to the way the I/O Address is programmed. The low order byte of the desired address is output to the Low Window Address port and the high order byte is output to the High Window Address port. These two registers collectively form the 16 bit IGC Window Address. The Window Address is not incremented by data I/O operations. A hardware master reset also clears these two registers.

The Light Pen Address Registers

The Light Pen Address registers contain the low and high bytes of the current light pen address. This is the address of the IGC video memory image byte that the light pen device (if present) is currently pointed at. Note that the validity of this address must be tested by a multiple polling process and/or checking the status of the light pen switch, which can be input through the most significant bit of the Left Input Device register. Software can locate the pen in a dark area by flashing the screen with the video invert enable through the control register.

The Input Device Registers

The Left and Right Input Device registers are two general purpose input ports for joystick and trackball type devices. Both of these ports are pinned out to a separate interface on the IGC board. SigmaSoft offers an adapter cable that connects one of these interfaces directly to any digital Atari standard trackball or joystick. The standard Atari interface uses a DB-9 type connector.