# TABLE OF CONTENTS

## CHAPTER 1
**INTRODUCTION**
- Microneye Bullet .................................................. 1-1
- Microneye Camera .................................................... 1-1
- IS32 Opticram ....................................................... 1-2

## CHAPTER 2
**TECHNIQUES FOR OPERATING THE MICRONEYE**
- Focus and f-stop Adjustments .................................... 2-1
- Close-up Ring .......................................................... 2-1
- Lighting Considerations ............................................. 2-2

## CHAPTER 3
**USING THE MICRONEYE WITH THE APPLE**
- Installation and Set Up ............................................ 3-1
- Files Included on Your Microneye Diskette .................... 3-2
- The Microneye Program ............................................. 3-3
- The Commander Program ............................................ 3-11
- The Greypic Program ................................................. 3-12
- The Greyscreen Program ............................................. 3-14
- The Enhanced Eye Program ......................................... 3-15

## CHAPTER 4
**USING THE MICRONEYE WITH THE IBM PC**
- Creating a Bootable Diskette ..................................... 4-1
- Installation and Setup .............................................. 4-2
- Files Included on Your Microneye Diskette .................... 4-4
- The Meye Program .................................................... 4-4

## CHAPTER 5
**USING THE MICRONEYE WITH THE COMMODORE 64**
- Installation and Set Up ............................................ 5-1
- Files Included on Your Microneye Diskette .................... 5-2
- The Microneye Program ............................................. 5-2
- The Assembly Language Connection (Meye6510.ex) ............. 5-6

## CHAPTER 6
**USING THE RS-232 MICRONEYE CAMERA**
- Hardware Requirements ............................................. 6-1
- Software ............................................................... 6-2
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>HOW YOUR COMPUTER TALKS TO THE MICRONEYE</td>
<td>7-1</td>
</tr>
<tr>
<td></td>
<td>MICRONEYE VERSIONS</td>
<td>7-1</td>
</tr>
<tr>
<td></td>
<td>THE SERIAL CONNECTION</td>
<td>7-1</td>
</tr>
<tr>
<td></td>
<td>COMMAND DEFINITIONS</td>
<td>7-3</td>
</tr>
<tr>
<td></td>
<td>EFFECTS OF COMMAND MODE COMBINATIONS</td>
<td>7-6</td>
</tr>
<tr>
<td>A</td>
<td>APPENDIX A BAUD RATE MODIFICATION</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>APPENDIX B TRANSMISSION TIME CONSIDERATIONS</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>APPENDIX C TROUBLESHOOTING</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>APPENDIX D IS32 OPTICRAM TECHNICAL INFORMATION</td>
<td>D-1</td>
</tr>
<tr>
<td></td>
<td>OPERATION</td>
<td>D-1</td>
</tr>
<tr>
<td></td>
<td>IS32 TECHNICAL SPECIFICATIONS</td>
<td>D-3</td>
</tr>
<tr>
<td></td>
<td>TOPOLOGY</td>
<td>D-4</td>
</tr>
<tr>
<td>E</td>
<td>APPENDIX E ANNOTATED ASSEMBLY LANGUAGE DRIVER FOR THE IBM PC</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>APPENDIX F GUIDE TO OPTICS SELECTION AND LIGHTING TECHNIQUES</td>
<td>F-1</td>
</tr>
<tr>
<td></td>
<td>LIGHTING CONSIDERATIONS FOR THE IS32 OPTICRAM</td>
<td>F-1</td>
</tr>
<tr>
<td></td>
<td>OPTICS</td>
<td>F-4</td>
</tr>
<tr>
<td></td>
<td>OTHER CONSIDERATIONS</td>
<td>F-17</td>
</tr>
<tr>
<td>G</td>
<td>APPENDIX G HARDWARE DESCRIPTION</td>
<td>G-1</td>
</tr>
<tr>
<td></td>
<td>TIMING GENERATION CIRCUIT</td>
<td>G-1</td>
</tr>
<tr>
<td></td>
<td>COMMAND RECEIVER CIRCUIT</td>
<td>G-2</td>
</tr>
<tr>
<td></td>
<td>ADDRESS REGISTERS</td>
<td>G-3</td>
</tr>
<tr>
<td></td>
<td>ADDRESS DESCRAMBLE, SOAK/, AND DIN/DOUT CIRCUITS</td>
<td>G-4</td>
</tr>
<tr>
<td></td>
<td>TRANSMITTER AND INTERRUPT GENERATOR CIRCUIT</td>
<td>G-6</td>
</tr>
<tr>
<td></td>
<td>ADDER AND END-OF-FRAME CIRCUIT</td>
<td>G-8</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

The MicronEye is the easiest and least expensive solution to numerous applications requiring a low cost, all digital imaging system. The MicronEye is an electro-optical system suitable for use with your computer as a peripheral. The three necessary dimensions -- optics, hardware and software -- are furnished with the standard package. There are three basic MicronEye systems:

1.1 MICRON EYE BULLET

(Shown in Illustration 1a). This system has the drive electronics located on a 9"x3" card inserted in the computer. The IS32 OpticRAM is located in the 1" diameter cylindrical Bullet case. The Bullet and computer are connected via a standard 16-wire flat ribbon cable. Micron recommends that the cable be less than 5 feet long and furnishes a 4-foot cable with the standard Bullet package.

1.2 MICRON EYE CAMERA

(Shown in Illustration 1b). This system has all of the drive electronics and IS32 OpticRAM on a 6"x3"x1" card which is mounted in a rectangular camera case. It also includes a 3"x3" serial interface card suitable for inserting in the Apple II, IBM PC, Radio Shack TRS-80 Color Computer and Commodore 64. The advantage the Camera has over the Bullet is that the Camera may be located remotely from the computer (up to 50 feet away).

For computers on which custom MicronEye interfaces are not available, an RS-232 compatible version of the MicronEye Camera is available. The RS-232 MicronEye Camera comes equipped with a male DB25P connector. Pin 2 (transmitted data) carries data from the MicronEye to the computer. Pin 3 (received data) sends data from the computer to the MicronEye. Pin 7 is a common ground. Power for the MicronEye (+5V capable of driving a 50 mA load) must be made available
1.3 IS32 OPTICRAM

The heart of the MicronEye is the OpticRAM. The OpticRAM was developed and is manufactured by Micron Technology, Inc. The OpticRAM is composed of 65,536 individual image sensing elements called pixels. These pixels are organized into two rectangles (often referred to as arrays) of 128 x 256 pixels each. Each array of cells is separated by an optical "dead" zone of about 25 elements in width.

When an image is focused onto the OpticRAM, a digital representation of the image is "exposed" on the OpticRAM. The MicronEye transmits this image from the OpticRAM to the computer. The software included with the MicronEye takes the transmitted image and displays it on the computer's graphics screen.

Because the image created by the OpticRAM is digital, the image produced is black and white. The MicronEye may produce shades of gray by multiple scans at different exposure times. MicronEye users with an Epson printer can produce pictures with grey tones with the software provided.

The low cost of the MicronEye is directly attributable to the technological advance represented by Micron's OpticRAM. In terms of cost per pixel, the OpticRAM represents a 1000x reduction in price over earlier generation image-sensing chips such as the CCD. As a result, the MicronEye brings capabilities to your computer which were previously available only to large industrial users.

The electronics in the MicronEye provide an interface between the OpticRAM and computer. It also provides a means by which the MicronEye can receive commands from the computer. Using a crystal to assure accuracy, the MicronEye drive electronics provides all the requisite timing signals and circuitry to execute commands received from the computer. The MicronEye automatically sequences the OpticRAM so that each image sensing element in the OpticRAM is accessed and the appropriate video information is returned to the computer for display or processing.

In addition, the MicronEye's electronic shutter is easily controlled by sending the MicronEye the appropriate commands. A command to the MicronEye to SOAK, "opens" the shutter. After the appropriate period of exposure has elapsed, a command to the MicronEye to REFRESH will "close" the shutter. The software provided automatically performs these functions. Chapter 7 explains the commands available for controlling the MicronEye for users who want to design their own assembly language interfaces. (For most users, the routines provided should be more than adequate.) As you might suspect,
the MicronEye's shutter is not a mechanical shutter. The MicronEye controls whether or not the OpticRAM is sensitive to light or not. This feature allows for precise continuous control of the MicronEye's "shutter speed."

If for any reason you must remove the OpticRAM from its socket, caution is imperative. The OpticRAM is susceptible to static and can be damaged by static electricity. Be certain to properly orient the OpticRAM when reinserting it into the socket. For the bullet, the OpticRAM is oriented properly when the red edge of the ribbon cable is on the same side of the camera as the Pin 1 notch on the OpticRAM. For the camera, the OpticRAM is oriented properly when the Pin 1 notch on the OpticRAM is on the same edge as the Pin 1 notch on other IC's in the camera. Removal of the OpticRAM from the bullet may require that the tips of the chip extractor tool be bent out slightly to accommodate the narrowness of the bullet housing.

Illustration 1

(a) MicronEye Bullet   (b) MicronEye Camera   (c) IS32 OpticRAM
CHAPTER 2

TECHNIQUES FOR OPERATING THE MICRONEYE

2.1 FOCUS AND F-STOP ADJUSTMENTS

The lens supplied with your MicronEye is an F1.6 16mm lens with adjustable f-stop. Please note that the lens has two controls which must be adjusted before the MicronEye will operate successfully: f-stop and focus control. The f-stop controls the amount of light admitted through the lens while the focus control focuses the image on to the surface of the image sensing device (the IS32 OptiCRAM).

For normal use, the lowest f-stop setting (1.6) is recommended. Any increase in the f-stop requires a compensating increase in the light source or in the exposure time. Please note there is a "C" setting which completely closes the aperture. A mechanical shutter is not needed since this function is performed electronically by the MicronEye.

The depth of focus (the distance the scene can move in relation to the MicronEye and still be in focus) is increased at higher f-stops. To optimize the result, increase the amount of light and/or the exposure time. A tradeoff of lighting, exposure time, f-stop and scene-to-MicronEye position is necessary to optimize the result.

2.2 CLOSE-UP RING

The lens is designed for viewing objects at a distance of at least 18 inches. Also supplied with the MicronEye is a close-up ring which allows the MicronEye to view objects as near as five inches. From this distance, normal text is clearly readable. The ring can be installed by unscrewing the lens from the MicronEye, inserting the ring over the threads of the lens screw, and screwing the lens back into the MicronEye. The ring acts as a spacer and extends the focal length of the lens. For experimenting with viewing objects as close as two to three inches, an acceptable short-term solution is to slowly unscrew the lens until the object comes into focus (taking care not to unscrew the lens so far that there are insufficient threads to hold
the lens onto the MicronEye).

For viewing objects at close range it is recommended that the user purchase a close-up lens. Since the MicronEye utilizes a standard C-mount lens, most camera retailers provide a wide assortment of special purpose lenses directly compatible with the MicronEye.

2.3 LIGHTING CONSIDERATIONS

The MicronEye requires a high contrast scene in order to image the object onto the OpticRAM. Unlike a TV camera which can respond to shades of gray, the OpticRAM is a digital device where each picture element will only respond to a black and white representation of a scene. All portions of the scene lighter than an arbitrary threshold are considered white and all portions of the scene darker than the threshold are considered black. If the exposure time is increased more of the scene falls on the white side of the threshold barrier. As the exposure time is decreased more of the scene falls on the black side of the threshold level.

The threshold level can be affected in one of three ways: (1) changing the exposure time; (2) changing the f-stop on the lens; and (3) changing the light on the scene itself. Doubling the exposure time is the same as opening the f-stop by one stop (changing the f-stop to the next smaller number) or, in other words, doubling the amount of light.

For optimum results from your MicronEye, careful consideration must be paid to lighting. In general, arbitrary lighting of the environment will not produce optimum results as it may result in low-contrast images, reflections, shadowing and extraneous details. A good lighting system illuminates the scene so that the complexity of the image is minimized while the information required for inspection or manipulation is enhanced.

2.3.1 Front Lighting

A front lit scene (where the MicronEye is on the same side of the scene as the light source or ambient light) sometimes lacks adequate contrast. Front lighting with a diffused light source can often be used to increase the contrast in a scene. If defects or points of interest are to be emphasized, side lighting such that the defects or points of interest cast a shadow or appear brighter through increased reflectivity may be used.
To set up a front lit scene, one or more flood lamps (found at most hardware stores) are arranged around the scene far enough away so that there are no shadows. Then the f-stop, focus control and lamps are adjusted for maximum contrast and focus. It is usually helpful to adjust the focus where the smallest part of the scene has the most detail.

In many instances you will want to diffuse the light coming from the flood lamps. Diffusing the light increases the uniformity of the light on the image. You can diffuse the light as simply as placing a piece of paper over the lamp. A better method of diffusion is to take a sheet of frosted mylar, diffused white plastic, or a sandblasted pane of glass and place it between the lamp and the subject. A diffused light source is most commonly used in defect detection and visual inspection applications.

2.3.2 Back Lighting

For a backlit scene, the light comes from behind the scene so that the object being viewed is shadowed into the MicronEye. Backlighting the object for maximum contrast will give the best repeatable results. Backlighting is recommended when using the MicronEye to measure an object or certain aspects of an object. Backlighting is often ideal for part recognition.

The backlit light source must be large enough so that the MicronEye, without the object in the field of view, will see a uniform amount of light. This is normally accomplished by using several flood lamps and shining the flood lamps onto a diffused surface (ground glass, or diffused white plastic, or frosted mylar), such that a uniform light source is created. Placing the object between the diffused surface and the MicronEye will shadow the object into the MicronEye with maximum contrast. Adjust the f-stop to the maximum value that the amount of light and exposure time will allow.
CHAPTER 3

USING THE MICRONEYE WITH THE APPLE

3.1 INSTALLATION AND SET UP

The MicronEye configured for use with the Apple II requires at least 48K of memory. The MicronEye is compatible with the Apple II+ and the Apple IIe.

Remove your MicronEye from its shipping carton. If you have purchased a Bullet, it will already be fully assembled. All that is required of you is to unfold the legs of the tripod and stand the MicronEye upright. If you have purchased a Camera, you will have to connect the Camera to the interface board with the cord which is provided.

Take a moment to examine the lens provided with the MicronEye. You will notice that there are two lens controls which must be adjusted before the MicronEye will operate successfully: f-stop and focus control. The f-stop controls the amount of light admitted through the lens and, for normal use, the lowest setting (1.6) is recommended. Any increase in the f-stop requires a compensating increase in the light source or in the exposure time. The recommended operating distance of the MicronEye is 18 inches or greater from the object it is viewing. You may be required to make a slight adjustment to the f-stop setting and/or the focus control once you have the MicronEye actually viewing an object.

Switch off the power to your computer, and you are ready to install the interface card into any available slot in the Apple. With the computer keyboard facing you, insert the interface card into the computer with the components on the right side of the card. The computer initially expects it in slot 2, but this can be changed from the keyboard once inside the program.

Insert the MicronEye diskette into the disk drive and switch on the power to your Apple. The MICRONEYE program discussed in detail below is automatically invoked when the Apple is turned on.
USING THE MICRONEYE WITH THE APPLE FILES INCLUDED ON YOUR MICRONEYE DISKETTE

3.2 FILES INCLUDED ON YOUR MICRONEYE DISKETTE

To assist you in developing personal applications for the MicronEye, both source listings and programs have been included in your diskette. A catalog and brief description of the files found on your diskette follows:

APPLESOFT CATALOG

A 045 MICRONEYE (MICRONEYE program as discussed below)
A 012 COMMANDER (COMMANDER program as discussed below)
A 014 GREYPIC (GREYPIC program as discussed below)
A 011 GREYSCREEN (GREYSCREEN program as discussed below)
A 011 ENHANCED EYE (ENHANCED EYE program as discussed below)
A 003 SLIDE SHOW (runs GREYSCREEN pictures on this disk)
T 033 T.CAMASM (source for CAMASM)
B 006 CAMASM (6502 routines for MICRONEYE and CAMASM)
T 011 T.EPRINT (source for EPRINT)
B 003 EPRINT (6502 Epson screendump routine)
T 029 T.GREYASM (source for GREYASM)
B 005 GREYASM (6502 routines for GREYPIC)
T 018 T.GSCRASM (source for GSCRASM)
B 003 GSCRASM (6502 routines for GREYSCREEN)
T 041 T.ENHANCER (source for ENHANCER)
B 006 ENHANCER (6502 routines for ENHANCED EYE)
T 002 EYEPARMS (parameter file for MICRONEYE)
B 034 BANBI (picture created using GREYSCREEN)
B 034 BANBI AND FLOWER (picture created using GREYSCREEN)
B 034 TEXT (picture created using GREYSCREEN)
B 034 ROBOTARM (picture created using GREYSCREEN)
B 034 EDITOR (picture created using GREYSCREEN)
B 034 WINNIE (picture created using GREYSCREEN)
B 034 BEARS (picture created using GREYSCREEN)

PASCAL DIRECTORY

MICRONEYE.CODE 19 (MICRONEYE program as discussed below)
COMMANDER.CODE 4 (COMMANDER program as discussed below)
GREYPIC.CODE 12 (GREYPIC program as discussed below)
CAMASM.CODE 9 (6502 routines for MICRONEYE/CAMASM)
GREYASM.CODE 6 (6502 routines used by GREYPIC)
SCREENIO.CODE 7 (Screen handling library code file)
MICROCAM.TEXT 30 (Source code for MICRONEYE)
COMMANDIT.TEXT 6 (Source code for COMMANDER)
GREYPIC.TEXT 10 (Source code for GREYPIC)
CAMASM.TEXT 26 (Source code for CAMASM)
GREYASM.TEXT 16 (Source code for GREYASM)
SCREENIO.TEXT 10 (Source code for SCREENIO)
EYEPARMS 1 (Parameter file used by MICRONEYE)
NOTE: The Pascal version of the MicronEye does not have the Applesoft equivalents of GREYSCREEN and ENHANCED EYE.

3.3 THE MICRONEYE PROGRAM

The MICRONEYE program lets a non-technical user harness a great deal of the MicronEye's power. The program incorporates the ability to show pictures transferred from the MicronEye onto your computer's screen, save pictures to disk for future use, and print pictures to a graphics printer.

When the program is invoked, a menu similar to the screen below is displayed:

<table>
<thead>
<tr>
<th>MICRONEYE DEMONSTRATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHAT WOULD YOU LIKE TO DO?</td>
</tr>
<tr>
<td>(1) START CAMERA</td>
</tr>
<tr>
<td>(2) SET UP CAMERA PARAMETERS</td>
</tr>
<tr>
<td>(3) DISPLAY REAL-TIME COMMANDS</td>
</tr>
<tr>
<td>(4) SAVE CURRENT CAMERA SETUP</td>
</tr>
<tr>
<td>(5) RECALL CAMERA SETUP FROM DISK</td>
</tr>
<tr>
<td>(6) CHANGE SLOT AND BAUD RATE</td>
</tr>
<tr>
<td>(7) TARGET PRACTICE</td>
</tr>
<tr>
<td>(8) EXIT PROGRAM</td>
</tr>
</tbody>
</table>

3.3.1 START CAMERA

Starting the MicronEye causes the screen to blank, and prepares the computer to begin the display of pictures using your computer's high resolution graphics capabilities. The MicronEye then begins sending what it sees to your computer. The computer then displays this picture onto the computer's screen. The size of the picture displayed can be modified by using the "SET UP CAMERA PARAMETERS" option.
When the MicronEye begins sending pictures to your computer, the MicronEye has no way of knowing if the picture is properly focused or if the proper exposure time has been selected. If you are having difficulty focusing or selecting the proper exposure setting, refer back to the chapter 2 on OPERATING TECHNIQUES.

There are several single-key commands that you can use when the camera is operating. These commands allow you to increase or decrease the exposure time, save pictures to disk, recall pictures from disk, print pictures to a printer, enable and disable the display of information about each picture displayed, select fixed or automatic exposure times, etc. These commands are called real-time commands and are discussed in the "REAL-TIME COMMANDS" section.

While the MicronEye is operating, you can return to the main menu at any time by typing "Q".

3.3.2 SET UP CAMERA PARAMETERS

When you select this option, a screen similar to the one shown below will be displayed:

```
MICRONOEY SETUP
SELECT LETTER OF DESIRED OPTION...
(PRESS <RETURN> TO EXIT)

PICTURE SIZE: A) 128 X 64   C) 256 X 128
                B) 256 X 64   D) 512 X 128

PICTURES/SCREEN: E) 1 PER SCREEN
                  F) 2 PER SCREEN

EXPOSURE CONTROL: G) FIXED EXPOSURE TIME
                   H) AUTO-ADJUST EXPOSURE

STATUS READOUTS: I) ENABLED
                  J) DISABLED

LIGHT MARGIN K)

--------------------------------------------------
PICTURE SIZE: 256 X 128 (1 PIC/SCREEN)
READOUTS ARE: ENABLED
EXPOSURE IS: FIXED

EXPOSURE LEN: 250 MSECS
LIGHT LEVEL: 45 %   MARGIN 5 %
```
3.3.2.1 PICTURE SIZE - Options "A" through "D" select the size of the picture that the MicronEye sends to the computer. Each picture is made up of thousands of black and white dots called pixels. When we say a picture is 128 x 64 in size, this means that the picture is made up of 64 rows of dots and that each row contains 128 dots of pixels. A 256 x 128 picture is made up of 32,768 pixels. Each pixel is either black or white.

The 128 x 64 and 256 x 128 picture size selections are compressed in the horizontal direction. The 256 x 64 and 512 x 128 picture size selections produce an image of normal proportions. Only the leftmost 280 pixels of the 512 x 128 picture will fit on the graphics screen.

3.3.2.2 PICTURES PER SCREEN - The MicronEye can take either one or two pictures at a time. If you elect to look at two pictures per screen, the computer will put the second picture right below the first picture. At first glance it may appear that you have just one picture that is twice as high when the computer is showing one picture per screen. If you look closely though, you may see that where the two pictures meet there is a slight discontinuity. For some applications this may not matter. In more exacting applications, you should restrict yourself to using only one picture per screen.

3.3.2.3 EXPOSURE CONTROL - You have the option of using a fixed or variable exposure time. Exposure time corresponds to the shutter speed of conventional 35mm cameras. If the picture from the MicronEye is too dark then a longer exposure time can be specified. If the picture is too light then a shorter exposure time can be specified. Exposure time can alternately be controlled by the use of real-time commands. The exposure time is specified in milliseconds. The speed at which the camera operates is equal to the exposure setting as long as the exposure time is greater than the time required for the MicronEye to transmit the picture to the computer. A more complete discussion of the interaction between exposure time and transmission time can be found in Appendix B.

As an alternative to manual exposure time control, automatic exposure adjustment can be specified from this setup menu or as a real-time command. Selecting the auto-adjust option tells the computer to evaluate the picture as it comes from the MicronEye to determine what percent of the pixels are white and what percent are black. When readouts are enabled, the percentage associated with LIGHT LEVEL is an approximation of how white the picture is: 100% being all white, 0% being all black.
When you select the auto-adjust feature you are requested to specify a light level between 0 and 100 and a margin which specified the allowed discrepancy from the prescribed light level. If you specify a light level of 45% and a margin of 5% then after each picture is received from the MicronEye, the computer will determine if the light level was between 40% and 50% (45% plus/minus 5%). If the light level was within the set bounds then the exposure time is left alone. If the light level is out-of-bounds then the exposure time is adjusted upward or downward to try and bring the next picture into the prescribed range.

The margin setting is also utilized by the alarm mode to set sensitivity. The alarm mode is explained in the section on real-time commands.

3.3.2.4 STATUS READOUTS - After displaying a picture from the MicronEye, the computer can optionally display the exposure time and light level of the picture just displayed. When status readouts are enabled, this information is displayed. Enabling this option, will slow down the rate at which pictures are updated on the screen. How much slower will depend on the exposure time setting and the type of computer you have.

In addition to being able to control readouts from the setup menu, a real-time command is available to enable and disable the readout display. On some computers, you may experience a difference in your picture's light level when switching back and forth between having readouts enabled and disabled.

3.3.2.5 LIGHT MARGIN - This is a convenient way of setting the light margin without altering the light level setting. It is especially useful for changing the MicronEye's sensitivity when being used in the alarm mode.

3.3.3 DISPLAY REAL-TIME COMMAND

There are several keystroke commands that can change how the MicronEye operates. After the computer displays each picture on the screen, it checks to see if a key has been pressed on the keyboard. If a key has been pressed, the computer checks to see if the key hit corresponds with its list of valid real-time commands. If so, the command is executed. If more than one key has been pressed during the scan only the last key struck is used.
Selecting the "DISPLAY REAL-TIME COMMANDS" options shows you the list of valid real-time commands. The screen should look somewhat like this:

<table>
<thead>
<tr>
<th>REAL-TIME COMMAND SUMMARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;  -- DECREASE EXPOSURE TIME</td>
</tr>
<tr>
<td>&gt;  -- INCREASE EXPOSURE TIME</td>
</tr>
<tr>
<td>A  -- TOGGLE ALARM MODE ON/OFF</td>
</tr>
<tr>
<td>C  -- CLEAR SCREEN</td>
</tr>
<tr>
<td>F  -- FIX EXPOSURE TIME TO CURRENT SETTING</td>
</tr>
<tr>
<td>L  -- LOAD PICTURE FROM DISK</td>
</tr>
<tr>
<td>N  -- PRINT PICTURE NEGATIVE ON EPSON</td>
</tr>
<tr>
<td>P  -- PRINT PICTURE ON EPSON</td>
</tr>
<tr>
<td>Q  -- QUIT AND RETURN TO MAIN MENU</td>
</tr>
<tr>
<td>R  -- TOGGLE DISPLAY READOUTS ON/OFF</td>
</tr>
<tr>
<td>S  -- SAVE PICTURE TO DISK</td>
</tr>
<tr>
<td>T  -- USING BLACK/WHITE RATIO (LIGHT LEVEL) OF CURRENT PICTURE, START AUTOMATIC LIGHT LEVEL TRACKING</td>
</tr>
</tbody>
</table>

The effects of each real-time command are explained in the pages that follow.

3.3.3.1 DECREASE EXPOSURE TIME - This command is activated by pressing the less-than key (comma also works). Each time this command is issued, the computer will decrease the MicronEye's exposure time. Each time the command is given the computer will decrease the exposure time in larger and larger steps. If the steps get too large, the computer may decide to decrease the exposure time in smaller and smaller steps. You may want to enable readouts and experiment with the increase and decrease exposure commands to get a better feel for how the commands interact and how the step size is increased and decreased by different combinations of the commands.

3.3.3.2 INCREASE EXPOSURE TIME - This command is activated by pressing the greater-than key (period also works). Its operation is similar to the "DECREASE EXPOSURE TIME" command except that the exposure time is increased rather than decreased.
3.3.3.3 TOGGLE ALARM MODE - This command is activated by the "A" key. If the alarm mode is off when you give this command, then alarm mode will be turned on. If the alarm mode is enabled then giving this command will disable the alarm mode. When you issue the command the computer will tell you whether you have enabled or disabled the alarm.

The alarm mode allows the MicronEye to function as a surveillance device. The light margin setting determines the sensitivity of the alarm. The greater the light margin setting, the less sensitive the MicronEye will be to change. The alarm is activated by changes in light level. If an object moves across the camera's field of view, an alarm will sound until a key is struck on the Apple's keyboard.

A user can also customize the computer's response to the alarm being tripped. The computer could automatically dial a phone number, activate recording equipment, etc.

3.3.3.4 CLEAR SCREEN - The computer clears the screen when the "C" key is struck. This command is rarely needed because the computer tries to clean up after itself whenever the size of the viewing area is changed.

3.3.3.5 FIX EXPOSURE TIME TO CURRENT SETTING - This command is invoked by striking the "F" key. The MicronEye normally uses the same exposure setting time after time, and only modifies the exposure setting when told to do so. This is referred to as a fixed exposure setting. The MicronEye can also operate such that the exposure time will change dynamically to maintain a specified light level. This is referred to as an auto-adjust setting.

When the camera is in the auto-adjust mode and you want to return to the fixed exposure mode use this command. The camera will fix the exposure time to the exposure time being used at the time the command is given.

3.3.3.6 LOAD PICTURE FROM DISK - A picture that was previously taken by the MicronEye and saved to disk can be displayed on the computer's screen by using this command. The load command is invoked by striking the "L" key.

The computer will then ask for the name given the picture when it was stored to disk. If the computer can find the file on disk, the picture will be displayed until a key is typed on the keyboard. Otherwise, an error message will be displayed and the computer will resume displaying pictures from the MicronEye. If you simply press
the <RETURN> key when prompted for a file name, then the computer will resume displaying pictures.

3.3.3.7 PRINT PICTURE ON EPSON - The "P" key causes the current picture being displayed to be printed on an Epson graphics printer in slot 1. This command can also be used after loading a picture from disk, by typing a "P" when prompted to "press <RETURN> to continue..."

The routine is intended for an Epson printer using a parallel interface. Attempting to select the print option without a printer or a non-Epson parallel printer will cause the program to hang. Some early models of the Epson graphics printer may not work properly either. The reason for all of the problems associated with printing graphics is that the standard PRINT and COUT routines will insert unwanted line feeds and carriage returns into the print stream.

If you have a screen dump routine for your printer, you should modify lines 2010, 1180, and 1190 of the MicronEye program to use your screen dump routine rather than the one supplied. An alternative to this approach would be to save the picture in uncompressed format (refer to SAVE PICTURE section) and then run your screen dump program to print the picture.

3.3.3.8 PRINT PICTURE NEGATIVE ON EPSON - This option is invoked by typing the "N" key. It operates exactly like the normal print option with the exception that white areas on the screen will print black, and black areas will print white.

3.3.3.9 QUIT AND RETURN TO MAIN MENU - You can return to the main menu by typing "Q". When you no longer wish to operate the MicronEye, select this option.

3.3.3.10 TOGGLE DISPLAY READOUTS ON/OFF - Display readouts are enabled or disabled by typing "R". If readouts are enabled then after each picture is received from the MicronEye, the computer will display the exposure time and light level for that picture. When readouts are enabled, the picture rate may be slowed down dramatically, so it is usually advisable to have readouts disabled whenever possible.
3.3.3.11 SAVE PICTURE TO DISK - Typing an "S" when the camera is operating tells the computer to save the current picture to disk. The computer will prompt for a filename and attempt to save the picture to disk. If an error is encountered in attempting to save the picture (usually due to insufficient disk space) then a message is displayed. Otherwise the picture is stored to disk.

(Applesoft only.) Normally, the MicronEye program will compress the picture before storing it to disk. Although this saves a lot of disk space, the pictures saved are incompatible with commercially available graphics manipulation packages and screen dump programs. If you prefer that the MicronEye program store pictures in a conventional, non-compressed format then perform the following sequence of DOS and Applesoft commands:

```
LOAD MICRONEYE
2027 FF = 1
SAVE MICRONEYE
```

The Pascal version of MicronEye saves pictures in compressed format only. This is because a standard format for a .FOTO file has not been defined by Apple or other graphics software companies.

3.3.3.12 TRACK EXPOSURE TIME USING AUTO LIGHT LEVEL ADJUST - The auto-adjust mode is selected by typing a "T". When auto-adjust is selected as a real-time option, the computer will use the light level of the current picture as the ideal light level. The light margin is the acceptable level of deviation from the ideal light level and should have been set previously from the MICRONEYE SETUP screen.

After each picture is received from the MicronEye, the computer determines if the light level was within the established bounds. If not, the computer will increase or decrease the exposure time of the next picture to try and get back to an acceptable light level. The auto-adjust mode is intended for applications where the MicronEye is focused on a fixed or semi-fixed scene.

3.3.4 SAVE CURRENT CAMERA SETUP

Selecting this option from the main menu tells the computer to save the currently defined setup as the setup the computer should initially use when starting the MicronEye program. The setup variables that are stored include PICTURE SIZE, PICTURES PER SCREEN, EXPOSURE METHOD, EXPOSURE TIME, READOUT SETTING, LIGHT LEVEL, LIGHT MARGIN, MICRONEYE SLOT, and BAUD RATE. The setup is saved to a file called EYEPARMS.
3.3.5 RECALL CAMERA SETUP FROM DISK

This option restores the camera setup to the settings in the EYEPARMS file. This is handy when you have been experimenting with a non-standard setup and want to go back to using your normal setup.

3.3.6 CHANGE SLOT AND BAUD RATE

When shipped from the factory the MicronEye has been set to operate at a baud rate of 153,600 bits/second. Also the MicronEye program expects the MicronEye to go in slot 2. The baud rate will not normally be changed by the user. However, since a slot may currently contain another card it is helpful to be able to specify an alternate slot for the MicronEye. It is usually desirable to save the current setup to disk after modifying the slot or baud rate since these changes are fairly permanent in nature.

3.3.7 TARGET PRACTICE

This option may prove useful to some users. It temporarily puts the MicronEye in auto-adjust mode, sets the ideal light level to 50%, and adjusts the exposure time after each frame until a 50% light level is achieved. When in target practice, striking any key on the keyboard will return you to the main menu and return the setup to what it was prior to invoking target practice.

The target practice feature was included mainly to demonstrate how auto-adjust mode works.

3.4 THE COMMANDER PROGRAM

The COMMANDER program is a lower level program than the MICRONEYE program. The program asks for a hexadecimal (Pascal) or decimal (Applesoft) command. This command corresponds with the command descriptions found in Chapter 7.

If the SEND mode is selected in the command byte, the user is also prompted for a soaktime. In the COMMANDER program, soaktime is the time in milliseconds that the program will wait at the end of each frame to allow the camera additional exposure time. If SOAK mode is selected, then the total exposure time will be the transmission time plus the soaktime.
USING THE MICRONEYE WITH THE APPLE
THE COMMANDER PROGRAM

If SOAK mode is not selected then total exposure time will equal
the soaktime. The computer will continue to send the camera the
specified command until the user types a key on the keyboard. The
user will be reprompted for another command unless the letter typed
was a "Q". A "Q" will exit the program.

Commands less than decimal 192 (CO hex) will inhibit the camera
from operating properly and probably cause the computer to hang.

3.5 THE GREYPIC PROGRAM

The GREYPIC program is a simplistic but effective demonstrator of
the MicronEye's grey scale capabilities. By taking the same picture
at several exposure settings, the program assigns a grey level to each
pixel depending on the number of times it was white throughout the
several exposure settings. Utilizing several of the features of the
MICRONEYE program, the GREYPIC program allows for real-time adjustment
of exposure time, saving and retrieving grey scale pictures on disk,
and pasting together several pictures to make a larger composite
picture. The GREYPIC program should be easily changed to work on
other graphics printers or even on standard dot matrix or line
printers.

The program is designed to operate with an EPSON printer (with
Graftrax) in slot 1. When the program is run you are asked to specify
the slot the MicronEye is in. You are also reminded to make sure that
the printer is online to prevent the program from hanging. The
program then begins displaying the picture being received from the
MicronEye on the upper third of the screen. The exposure time is
initially set to 1/3 of a second.

The GREYPIC program requires some setup to get a clear image. A
high and low setting for the exposure range must be set. This can be
accomplished with the use of the L, H, and B commands. The high
exposure setting must be decided upon such that the image is not too
light to display all of the details of the object being viewed on the
screen. The dark exposure setting should be set so that no streaking
occurs on the screen. Any slight discrepancy in exposure time can be
corrected using the increase and decrease exposure time commands. Be
sure that the object is in focus and the F-Stop is at the correct
setting.

To make a composite picture, place the object being viewed to
show the uppermost details which are desired to be displayed on the
screen. On the screen, some distance should be allowed between the
edge of the picture and the image of the object. After the picture is
sharp and clear, a printout can be made. Press the P command and the
upper third of the screen will be printed.

3-12
Press the "2" key to display the picture on the middle third of the screen. Raise the level of the object until the image on the upper third of the screen is directly on top of the image being displayed on the middle third of the screen. A flowing, continuous picture should be evident on the screen with no recognizable division between the two pictures on the screen. Press the P command and the middle third of the screen will be printed.

Press the "3" key to display the picture on the bottom third of the screen. Again, raise the level of the object until the bottom image is directly aligned with the middle third of the screen. A continuous picture should be displayed on the screen with no obvious breaks between the three sections of the screen. Press the P command and the bottom third of the screen will be printed.

By using the N command to scroll the image, a picture of any desired length can be printed. Using the same method already described, scroll the screen up one third, raise the level of the object and print the section.

If the object being displayed on the screen by the camera is in a fixed position, the height of the camera can be raised or lowered. Keep in mind at all times that the MicronEye should be kept parallel and perpendicular with the object being viewed. Setting an object on a movable platform (like a music stand) is one possible way to raise the level of an object.

If it is desired to save the picture to disk, each section of the screen must be saved separately. Press the S command and the image will be saved to be used at a later date.

The grey-scaled picture can be recalled from disk using the R command. If you desire to print the picture that has been recalled then press "P" rather than <RETURN> once the picture has been displayed.
The following real-time commands control the MicronEye while using the GREYPIC program:

< -- Decrease exposure time (comma also works).
> -- Increase exposure time (period also works).
C -- Clear entire screen.
1 -- Use upper third of screen to display picture on.
2 -- Use middle third of screen to display picture on.
3 -- Use bottom third of screen to display picture on.
N -- Rollup screen. (Middle third of screen moves to upper third, bottom third moves to middle third, and bottom third of screen is cleared. When using this option, it is best to be displaying the picture in the bottom third of the screen.) By using this command and the 1, 2, and 3 commands the user can piece together a picture of any length.
E -- Display the current exposure time and change the exposure time to a new value.
L -- Use current exposure setting as the lowest exposure setting when creating a grey-scaled picture.
H -- Use current exposure setting as the highest exposure setting when creating a grey-scaled picture.
B -- Bracket the exposure range for a grey-scaled picture. (User will be prompted for a high and a low setting.)
P -- Create a grey-scaled picture using the current high and low exposure settings, and print the picture on the Epson. (Seven intermediate exposure levels are used in addition to the high and low values to create a picture with nine levels of grey.)
S -- Save a grey-scaled picture to disk rather than print it.
R -- Recall a grey-scaled picture from disk and print it.
D -- dump (BSAVE) entire hi-res screen to disk.
G -- get previously BSAVE'D hi-res screen from disk and display it.
Q -- Exit program.

The GREYPIC program is easily modified to create images with up to 256 levels of grey. Although your computer has no means of displaying this many levels of grey, there are some rather expensive devices available for displaying and printing such images.

3.6 THE GREYSCREEN PROGRAM

The GREYSCREEN program is a takeoff from the GREYPIC program. However, the GREYSCREEN program attempts to show pseudo-greytone images on the screen. Because the Apple has no true shades of grey, we must simulate the grey by alternating black and white pixels. As in the GREYPIC program the MicronEye uses different exposure times to determine shades of grey.
The GREYSCREEN program uses two different exposure times which are controlled from the keyboard. If a pixel from the camera is white for both exposure times, then the wide pixel on the screen is all white. If a pixel from the camera is black for both exposure times, then the wide pixel on the screen is all black. If a pixel from the camera is white at one exposure time, and black at the other, then one side of the wide pixel on the screen will be black and the other will be white.

As in the GREYPIC program, the screen is divided into three partitions. These partitions are selectable from the keyboard and allow a composite image to be created on the screen which may be printed or stored for later retrieval or manipulation.

The concept utilized by the GREYSCREEN program is easily transferrable to other computers. Computers such as the IBM PC, Commodore 64, and TRS-80 Color Computer have implemented a medium resolution graphics mode which uses two bits to represent the pixel color on the screen. At the very least, black, white, light grey, and dark grey are available for creating an image. The obvious advantage over the Apple is the fact that real shades of grey are available for display.

The real-time commands available for use with the GREYSCREEN program are:

- < -- Decrease exposure time (comma also works)
- > -- Increase exposure time (period also works)
- 1 -- Use top 1/3 of graphics screen
- 2 -- Use middle 1/3 of graphics screen
- 3 -- Use bottom 1/3 of graphics screen
- N -- Scroll screen up by one-third (N also works)
- S -- Save (BSAVE) screen to disk
- L -- Load (BLOAD) screen from disk
- C -- Clear screen
- E -- Set exposure time
- P -- Print image to EPSON (slot 1, requires Graftrax)
  (Can also be used immediately after LOAD command)
- M -- Modify delta between high and low exposure (default 20)
- Q -- Quit program
- SPACEBAR -- Freezes frame until key hit

3.7 THE ENHANCED EYE PROGRAM

The ENHANCED EYE program goes a step further than any of the other Apple MicronEye programs. It manipulates the pixels received from the camera to improve the image quality. Because of the amount of processing required for manipulation, the processing is done between scans. This slows down the frame to frame operation of the camera but provides an image of greater quality than any of the other
methods demonstrated in other programs.

When the program begins, the display image is 256 x 64. Once the subject has been focused and the appropriate light level determined, the user can type "E" to enter the ENHANCE mode, "F" to enter the ENHANCE mode with FILLIN, 'U' for UNENHANCED mode, and 'N' to return to the 256 x 64 mode.

The actual enhancement of the image is done relatively fast. But because of the way the Apple high resolution graphics are implemented (1 color bit and 7 data bits per byte) and the fact that the enhancement is performed on a bitmap image (8 data bits per byte), the time required to convert the bitmap image to the Apple format takes in excess of a second. For non-display applications the ENHANCEER assembly language routine could be modified to perform the enhancement but skip the display to screen, thereby greatly increasing the operating speed of this program.

The following set of real-time commands are available from the ENHANCED EYE program:

- `<` -- Decrease exposure time (comma also works)
- `>` -- Increase exposure time (period also works)
- `E` -- Enhance image without fillin (512 x 128)
- `F` -- Enhance image with fillin (512 x 128)
- `U` -- Display unenhanced image
- `N` -- Display 128 x 64 image (rest of screen is not cleared)
- `G` -- Create grey-tone image from dual exposures
- `L` -- Load (BLOAD) screen from disk
- `P` -- Print image to EPSON (slot 1, requires Graftrax)
  (Can also immediately follow LOAD or GREY command)
- `S` -- Save (BSAVE) screen to disk
  (Can also immediately follow GREY command)
- `C` -- Clear screen
- `T` -- Set exposure time
- `M` -- Modify delta between high and low exposure (default 20)
- `Q` -- Quit program
- `SPACEBAR` -- Freeze frame until key hit
CHAPTER 4
USING THE MICRONEYE WITH THE IBM PC

4.1 CREATING A BOOTABLE DISKETTE

The diskette included with your MicronEye is NOT copy-protected and contains source code for all programs. The MicronEye diskette does not contain any system files, and as such is not bootable. This section shows you how to create a "working" copy of your diskette that includes all the necessary system files. Alternately, you could choose to always use the MicronEye diskette in drive B: and change the MEYE.BAT file to read "A:BASIC B:MEYE".

To create a bootable version of the MicronEye diskette and a backup of the diskette you need the following:

1. Your MicronEye diskette;

2. Your DOS system diskette (with the DOS utilities on it);

3. An unused diskette (new unformatted or old unused with data).

The two step process below will create a working copy of the MicronEye diskette on the unused diskette. This allows you to keep your original diskette in a safe place as a backup.

4.1.1 MULTIPLE DRIVE SYSTEMS Versus SINGLE DRIVE SYSTEMS

The same process that follows can be used by both multiple and single drive system owners -- the single drive owners just get to shuffle diskettes in and out of their drive a lot more.
USING THE MICRON EYE WITH THE IBM PC
CREATING A BOOTABLE diskette

4.1.2 STEP 1

This step puts the system on your MicronEye diskette. Insert the MicronEye diskette in drive B (Drive A for you single drive owners) and type "B:MOV DOS" and press return.

For this first step of the process, when the computer asks you to mount a diskette in:

DRIVE A: insert the DOS diskette

DRIVE B: insert the MicronEye diskette.

Step one is complete when the message "Insert target diskette in A" appears.

4.1.3 STEP 2

This step builds your working MicronEye copy on your unused diskette. Be aware that this step will destroy the current content of your unused diskette if it contains data.

For this second part of the process, when the computer asks you to mount a diskette in:

DRIVE A: insert your unused diskette.

DRIVE B: insert the MicronEye diskette.

Step two is complete when the "$ $$MOV DOS IS COMPLETE $$" message appears.

Your working copy of the MicronEye is now ready. Store your original MicronEye diskette in a safe place as a backup.

4.2 INSTALLATION AND SETUP

The MicronEye configured for use with the IBM PC requires at least 96K of memory and the Color/Graphics Monitor Adaptor board. A color monitor is not required. Your monitor should be attached via the composite video connector or the RGB connector. The MicronEye is compatible with both the IBM PC and the IBM XT.

Remove your MicronEye from its shipping carton. If you have purchased a Bullet, it will already be fully assembled. All that is required of you is to unfold the legs of the tripod and stand the MicronEye upright. If you have purchased a Camera, you will have to
connect the Camera to the interface board with the cord which is provided.

Take a moment to examine the lens provided with the MicronEye. You will notice that there are two lens controls which must be adjusted before the MicronEye will operate successfully: f-stop and focus control. The f-stop controls the amount of light admitted through the lens and, for normal use, the lowest setting (1.6) is recommended. Any increase in the f-stop requires a compensating increase in the light source or in the exposure time. The recommended operating distance of the MicronEye is 18 inches or greater from the object it is viewing. You may be required to make a slight adjustment to the f-stop setting and/or the focus control once you have the MicronEye actually viewing an object.

Switch off the power to your computer, and you are ready to install the interface card into any available slot in the computer. With the computer keyboard facing you, insert the interface card into the computer with the components on the right side of the card. The interface card does not include a mounting bracket. Remove the retaining bracket corresponding to the slot into which you are inserting the interface card. The cable between the MicronEye and interface card should be routed through the opening created by removing the retaining bracket. Replace the cover on your computer and turn on your computer.
4.3 FILES INCLUDED ON YOUR MICRONEYE DISKETTE

To assist you in developing personal applications for the MicronEye, both source listings and programs have been included in your diskette. A catalog and brief description of the files found on your diskette follow:

- MOVDO.S.BAT: (Command file to create bootable diskette)
- MICRONEYE.BAT: (Command file to invoke the MEYE program)
- MEYE.BAT: (Same as MICRONEYE.BAT)
- MEYEDRVR.ASM: (Source file for 8088 MicronEye routines)
- MEYEDRVR.BAS: (BLOADable 8088 routines for the MicronEye)
- MEYE.BAS: (BASIC MEYE program described below)
- MEYECOMP.BAT: (Command file to assemble MEYEDRVR)
- MEYESAVE.BAS: (BASIC program used by MEYECOMP)
- MEYPEPARMS: (Optional parameter file for MEYE)

4.4 THE MEYE PROGRAM

The MICRONEYE program lets a non-technical user harness a great deal of the MicronEye's power. The program incorporates the ability to show pictures transferred from the MicronEye onto your computer's screen, save pictures to diskette for future use, and print pictures to a graphics printer. Run the program by simply typing MICRONEYE or MEYE in response to the system prompt.
When the program is invoked, a menu similar to the screen below is displayed:

**MicronEye Demonstrator**

<table>
<thead>
<tr>
<th>MICRONEYE ACTIVITY OPTIONS</th>
<th>MICRONEYE SETUP OPTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Selection...</strong></td>
<td></td>
</tr>
<tr>
<td>1) Start MicronEye</td>
<td>Picture size and type</td>
</tr>
<tr>
<td>2) Change setup</td>
<td>a) 128 x 64 (black/white)</td>
</tr>
<tr>
<td>3) Recall setup from diskette</td>
<td>b) 512 x 64 (black/white)</td>
</tr>
<tr>
<td>4) Save setup to diskette</td>
<td>c) 512 x 64 (grey)</td>
</tr>
<tr>
<td>5) Explain real-time commands</td>
<td>d) 512 x 128 (black/white)</td>
</tr>
<tr>
<td>6) Exit program</td>
<td>e) 640 x 128 (black/white)</td>
</tr>
<tr>
<td></td>
<td>f) 640 x 128 (grey)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CURRENT MICRONEYE SETUP</th>
<th>Mode settings (toggled)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture size: 640 x 128 (grey)</td>
<td>g) Pictures/screen(1 or 2)</td>
</tr>
<tr>
<td>Pics/screen: 1</td>
<td>h) Status readouts(ON/OFF)</td>
</tr>
<tr>
<td>Readouts: DISABLED</td>
<td>i) Exposure (FIXED/AUTO)</td>
</tr>
<tr>
<td>Exposure: FIXED</td>
<td></td>
</tr>
<tr>
<td>Exposure time: 300</td>
<td>Exposure control</td>
</tr>
<tr>
<td>Light level: 50%</td>
<td>j) Set exposure time</td>
</tr>
<tr>
<td>margin: 5%</td>
<td>k) Set light level</td>
</tr>
<tr>
<td></td>
<td>l) Set light margin</td>
</tr>
</tbody>
</table>

4.4.1 START CAMERA

Starting the MicronEye causes the screen to blank, and prepares the computer to begin the display of pictures using your computer's high resolution graphics capabilities. The MicronEye then begins sending what it sees to your computer. The computer then displays this picture onto the computer's screen. The size of the picture displayed can be modified by using the "CHANGE SETUP" option.

When the MicronEye begins sending pictures to your computer, the MicronEye has no way of knowing if the picture is properly focused or if the proper exposure time has been selected. If you are having difficulty focusing or selecting the proper exposure setting, refer to the chapter 2 on OPERATING TECHNIQUES.

There are several single-key commands that you can use when the camera is operating. These commands allow you to increase or decrease the exposure time, save pictures to diskette, recall pictures from diskette, print pictures to a printer, enable and disable the display.
of information about each picture displayed, select fixed or automatic exposure times, etc. These commands are called real-time commands and are discussed in the "REAL-TIME COMMANDS" section.

While the MicronEye is operating, you can return to the main menu at any time by typing "Q".

4.4.2 CHANGE SETUP

After selecting this option the computer expects you to change one of the parameters (A through L) displayed on the right half of the screen. After changing the desired parameters simply press the SPACEBAR to exit the CHANGE SETUP mode.

4.4.2.1 PICTURE SIZE AND TYPE - Options "A" through "F" select the size of the picture that the MicronEye sends to the computer. Each picture is made up of thousands of black and white dots called pixels. When we say a picture is 128 x 64 in size, this means that the picture is made up of 64 rows of dots and that each row contains 128 dots of pixels. A 512 x 128 picture is made up of 65,536 pixels. Each pixel is either black or white.

The 128 x 64 and 512 x 128 picture size selections are compressed in the horizontal direction. The 512 x 64 and 640 x 128 picture size selections produce an image of normal proportions. The 512 x 64 and 640 x 128 pictures sizes allow for two types of pictures--black & white or grey. Although the black and white picture may appear to have grey in it, this is a pseudo-grey caused by closely spaced black and white pixels. The grey picture is created by taking a second exposure of the same picture with a 20% shorter exposure time. The two pictures are then combined in software to produce a single picture on the screen. On a sophisticated imaging system this method is used to produce pictures with over 64 levels of grey.

4.4.2.2 PICTURES PER SCREEN - The MicronEye can take either one or two pictures at a time. If you elect to look at two pictures per screen, the computer will put the second picture right below the first picture. At first glance it may appear that you have just one picture that is twice as high when the computer is showing one picture per screen. If you look closely though, you may see that where the two pictures meet there is a slight discontinuity. For some applications this may not matter. In more exacting applications, you should restrict yourself to using only one picture per screen.
4.4.2.3 EXPOSURE CONTROL - You have the option of using a fixed or variable exposure time. Exposure time corresponds to the shutter speed of conventional 35mm cameras. If the picture from the MicronEye is too dark then a longer exposure time can be specified. If the picture is too light then a shorter exposure time can be specified. Exposure time can alternately be controlled by the use of real-time commands. The exposure time is specified in milliseconds. The speed at which the camera operates is equal to the exposure setting as long as the exposure time is greater than the time required for the MicronEye to transmit the picture to the computer. A more complete discussion of the interaction between exposure time and transmission time can be found in the section 5.0 of the manual.

As an alternative to manual exposure time control, automatic exposure adjustment can be specified from this setup menu or as a real-time command. Selecting the auto-adjust option tells the computer to evaluate the picture as it comes from the MicronEye to determine what percent of the pixels are white and what percent are black. When readouts are enabled, the percentage associated with LIGHT LEVEL is an approximation of how white the picture is: 100% being all white, 0% being all black.

When you select the auto-adjust feature you are requested to specify a light level between 0 and 100 and a margin which specified the allowed discrepancy from the prescribed light level. If you specify a light level of 45% and a margin of 5% then after each picture is received from the MicronEye, the computer will determine if the light level was between 40% and 50% (45% plus/minus 5%). If the light level was within the set bounds then the exposure time is left alone. If the light level is out-of-bounds then the exposure time is adjusted upward or downward to try and bring the next picture into the prescribed range.

The margin setting is also utilized by the alarm mode to set sensitivity. The alarm mode is explained in the section on real-time commands.

4.4.2.4 STATUS READOUTS - After displaying a picture from the MicronEye, the computer can optionally display the exposure time and light level of the picture just displayed. When status readouts are enabled, this information is displayed. Enabling this option, will slow down the rate at which pictures are updated on the screen. How much slower will depend on the exposure time setting and the type of computer you have.

In addition to being able to control readouts from the setup menu, a real-time command is available to enable and disable the readout display. On some computers, you may experience a difference in your picture's light level when switching back and forth between
having readouts enabled and disabled.

4.4.2.5 LIGHT MARGIN - This is a convenient way of setting the light margin without altering the light level setting. It is especially useful for changing the MicronEye's sensitivity when being used in the alarm mode.

4.4.3 DISPLAY REAL-TIME COMMAND

There are several keystroke commands that can change how the MicronEye operates. After the computer displays each picture on the screen, it checks to see if a key has been pressed on the keyboard. If a key has been pressed, the computer checks to see if the key hit corresponds with its list of valid real-time commands. If so, the command is executed. If more than one key has been pressed during the scan only the last key struck is used. Selecting the "DISPLAY REAL-TIME COMMANDS" options shows you the list of valid real-time commands. The screen should look somewhat like this:

REAL-TIME COMMAND SUMMARY

< -- Decrease exposure time (comma also works)
> -- Increase exposure time (period also works)
A -- Toggle alarm mode on/off
C -- Clear screen
F -- Fix exposure time to current setting
L -- Load picture from diskette
P -- Print picture on printer
Q -- Quit and return to main menu
R -- Toggle display readouts on/off
S -- Save picture to diskette
T -- Use auto-adjust exposure (light level tracking)
/ -- Toggle pictures per screen (1 or 2)

1 -- 128 x 64 picture (black & white)
2 -- 512 x 64 picture (black & white)
3 -- 512 x 64 picture (grey)
4 -- 512 x 128 picture (black & white)
5 -- 640 x 128 picture (black & white)
6 -- 640 x 128 picture (grey)
The effects of the various real-time commands are explained in the pages that follow.

4.4.4 DECREASE EXPOSURE TIME

This command is activated by pressing the less-than or comma key.

Each time this command is issued, the computer will decrease the MicronEye's exposure time. Each time the command is given the computer will decrease the exposure time in larger and larger steps. If the steps get too large, the computer may decide to decrease the exposure time in smaller and smaller steps. You may want to enable readouts and experiment with the increase and decrease exposure commands to get a better feel for how the commands interact and how the step size is increased and decreased by different combinations of the commands.

4.4.4.1 INCREASE EXPOSURE TIME - This command is activated by pressing the greater-than, or period key. Its operation is similar to the "DECREASE EXPOSURE TIME" command except that the exposure time is increased rather than decreased.

4.4.4.2 TOGGLE ALARM MODE - This command is activated by the "A" key. If the alarm mode is off when you give this command, then alarm mode will be turned on. If the alarm mode is enabled then giving this command will disable the alarm mode. When you issue the command the computer will tell you whether you have enabled or disabled the alarm.

The alarm mode allows the MicronEye to function as a surveillance device. The light margin setting determines the sensitivity of the alarm. The greater the light margin setting, the less sensitive the MicronEye will be to change. The alarm is activated by changes in light level. If an object moves across the camera's field of view, an alarm will sound until a key is struck on the keyboard.

A user can also customize the computer's response to the alarm being tripped. The computer could automatically dial a phone number, activate recording equipment, etc.
4.4.4.3 CLEAR SCREEN - The computer clears the screen when the "C" key is struck. This command is rarely needed because the computer tries to clean up after itself whenever the size of the viewing area is changed.

4.4.4.4 FIX EXPOSURE TIME TO CURRENT SETTING - This command is invoked by striking the "F" key.

The MicronEye normally uses the same exposure setting time after time, and only modifies the exposure setting when told to do so. This is referred to as a fixed exposure setting. The MicronEye can also operate such that the exposure time will change dynamically to maintain a specified light level. This is referred to as an auto-adjust setting.

When the camera is in the auto-adjust mode and you want to return to the fixed exposure mode use this command. The camera will fix the exposure time to the exposure time being used at the time the command is given.

4.4.4.5 LOAD PICTURE FROM Diskette - A picture that was previously taken by the MicronEye and saved to diskette can be displayed on the computer's screen by using this command. The load command is invoked by striking the "L" key.

The computer will then ask for the name given the picture when it was stored to diskette. If the computer can find the file on diskette, the picture will be displayed until a key is typed on the keyboard. Otherwise, an error message will be displayed and the computer will resume displaying pictures from the MicronEye. If you simply press the <RETURN> key when prompted for a file name, then the computer will resume displaying pictures.

4.4.4.6 PRINT PICTURE ON EPSON - The "P" key causes the current picture being displayed to be printed on an Epson or IBM printer. This command can also be used after loading a picture from diskette, by typing a "P" when prompted to "press <RETURN> to continue..."
4.4.4.7 QUIT AND RETURN TO MAIN MENU - You can return to the main menu by typing "Q". When you no longer wish to operate the MicronEye, select this option.

4.4.4.8 TOGGLE DISPLAY READOUTS ON/OFF - Display readouts are enabled or disabled by typing "R". If readouts are enabled then after each picture is received from the MicronEye, the computer will display the exposure time and light level for that picture. When readouts are enabled, the picture rate may be slowed down dramatically, so it is usually advisable to have readouts disabled whenever possible.

4.4.4.9 SAVE PICTURE TO Diskette - Typing an "S" when the camera is operating tells the computer to save to current picture to diskette. The computer will prompt for a filename and attempt to save the picture to diskette. If an error is encountered attempting to save the picture (usually due to insufficient diskette space) then a message is displayed. Otherwise the picture is stored to diskette.

4.4.4.10 TRACK EXPOSURE TIME USING AUTO LIGHT LEVEL ADJUST - The auto-adjust mode is selected by typing a 'T'. When auto-adjust is selected as a real-time option, the computer will use the light level of the current picture as the ideal light level. The light margin is the acceptable level of deviation from the ideal light level and should have been set previously from the MICRONEYE SETUP screen.

After each picture is received from the MicronEye, the computer determines if the light level was within the established bounds. If not, the computer will increase or decrease the exposure time of the next picture to try and get back to an acceptable light level. The auto-adjust mode is intended for applications where the MicronEye is focused on a fixed or semi-fixed scene.

4.4.5 SAVE CURRENT CAMERA SETUP

Selecting this option from the main menu tells the computer to save the currently defined setup as the setup the computer should initially use when starting the MicronEye program. The setup variables that are stored include PICTURE SIZE, PICTURES PER SCREEN, EXPOSURE METHOD, EXPOSURE TIME, READOUT SETTING, LIGHT LEVEL, and LIGHT MARGIN. The setup is saved to a file called MEYEPPAARS.
4.4.6 RECALL CAMERA SETUP FROM Diskette

This option restores the camera setup to the settings in the MEYEPARMS file. This is handy when you have been experimenting with a non-standard setup and want to go back to using your normal setup.
5.1 INSTALLATION AND SET UP

Remove your MicronEye from its shipping carton. If you have purchased a Bullet, it will already be fully assembled. All that is required of you is to unfold the legs of the tripod and stand the MicronEye upright. If you have purchased a Camera, you will have to connect the Camera to the interface board with the cord which is provided.

Take a moment to examine the lens provided with the MicronEye. You will notice that there are two lens controls which must be adjusted before the MicronEye will operate successfully: f-stop and focus control. The f-stop controls the amount of light admitted through the lens and, for normal use, the lowest setting (1.6) is recommended. Any increase in the f-stop requires a compensating increase in the light source or in the exposure time. The recommended operating distance of the MicronEye is 18 inches or greater from the object it is viewing. You may be required to make a slight adjustment to the f-stop setting and/or the focus control once you have the MicronEye actually viewing an object.

Switch off the power to your computer, and you are ready to install the interface card into the cartridge slot located at the right rear of the computer. Insert the interface card into the slot using the orientation indicated on the enclosure.

Insert the MicronEye diskette into the disk drive and switch on the power to the computer.
5.2 FILES INCLUDED ON YOUR MICRONEYE DISKETTE

To assist you in developing personal applications for the MicronEye, both source listings and programs have been included in your diskette. A catalog and brief description of the files found on your diskette follows:

- MICRONEYE (Sample BASIC program using MEYE6510.EX)
- MEYE6510.EX (Assembly language routines discussed below)
- MEYE.SRC.1 (Source listing part 1 for MEYE6510.EX)
- MEYE.SRC.2 (Source listing part 2 for MEYE6510.EX)
- MEYE.SRC.3 (Source listing part 3 for MEYE6510.EX)

5.3 THE MICRONEYE PROGRAM

The MICRONEYE program lets a non-technical user harness a great deal of the MicronEye's power. The program incorporates the ability to show pictures transferred from the MicronEye onto your computer's screen, save pictures to disk for future use, and print pictures to an Epson or Gemini graphics printer.

To execute the MICRONEYE program, type LOAD "MICRONEYE",8 (followed by the RETURN key) and then type RUN. The program will load the assembly language routines (MEYE6510) for the MicronEye from disk. Once the routines have been loaded, the Commodore 64's screen immediately turn white and the MicronEye will begin sending pictures to the computer. You will see on the top two lines of the display the current operating mode of the MicronEye. When the program begins execution the mode display should read "B&W 1-PICTURE NORMAL". The second line of the display should read "SOAK TIME: 350".

Between pictures from the MicronEye, the computer checks for commands entered by the user on the keyboard. Because the MicronEye must operate with the interrupts turned off on the Commodore 64, the computer might not notice a key being pressed unless you keep the key pressed down a bit longer than you may be accustomed. As you work with the MicronEye you will acquire a feel for how long to keep the key pressed down. The best way to tell that the computer has noticed your command is to watch the mode display at the top of the screen. The mode display will be updated as soon as the command is detected.

The only problem associated with keeping the key pressed down is that when issuing the SAVE or LOAD commands you may need to use the DEL key to get rid of any extra characters that are displayed on the screen after FILENAME? before entering the LOAD or SAVE file name.
The MICRONEYE program allows several commands. They are explained in detail below. A summarization of the commands follows the explanation.

5.3.1 BLACK AND WHITE MODE

The BLACK AND WHITE option is selected by typing the "B" key. The MicronEye sees only in black and white. However, the computer can tell the MicronEye to take several pictures of the same scene at varied exposure times. The computer can then combine these several images together into a single picture with grey levels. Normally, a black and white image is adequate for processing an image. In this mode, the computer receives pictures from the MicronEye and displays each picture on the screen after it has been received.

5.3.2 GREY MODE

The GREY mode is selected by typing the "G" key. The GREY mode is the multiply-exposed grey level picture-taking technique alluded to above. In this mode, the computer instructs the MicronEye to take three pictures at varied exposure times. After the computer has received these three pictures, it "adds" them together and displays the result on the screen. Grey mode operates much slower than black and white mode because the computer has to get three pictures from the MicronEye for every picture displayed.

5.3.3 PICTURES PER SCREEN

The MicronEye can take either one or two pictures at a time. This is because the IS32 OpticRAM has 2 separate arrays which are both light-sensitive. By pressing the "1" key, the 1-PICTURE mode is selected. By pressing the "2" key, the 2-PICTURE mode is selected.

If you elect to look at two pictures per screen, the computer will put the second picture right below the first picture. At first glance it may appear that you have just one picture that is twice as high when the computer is showing one picture per screen. If you look closely though, you may see that where the two pictures meet there is a slight discontinuity. For some applications this may not matter. In more exacting applications, you should restrict yourself to using only one picture per screen.

You should be aware that when using the 2-PICTURE mode, the lower picture may have a tendency to be slightly darker than the upper picture. This is because the upper and lower array in the OpticRAM
have a slightly different sensitivity to light. Since the OptiCRAM was designed with the intent that only one of the arrays was to be used at a time, you might consider the second picture a freebie.

5.3.4 ENHANCED MODE

The ENHANCED mode is selected by typing the "E" key. The MicronEye can send images with either 128 x 64 resolution or 256 x 128 resolution. When using the 256 x 128 image size, resolution is increased fourfold. The increased resolution costs in two ways. First, it takes four times longer to send the 256 x 128 image than the 128 x 64 image. Second, the 256 x 128 image must be massaged through an enhancement algorithm to make a crisp image. This all takes time. To the extent that time is not a factor, the enhanced mode will generate much better pictures than the normal mode.

5.3.5 NORMAL MODE

NORMAL mode is selected by typing the "N" key and is the opposite of enhanced mode. Selecting NORMAL mode instructs the MicronEye to transmit 128 x 64 sized pictures.

5.3.6 DECREASE EXPOSURE TIME BY 10 MILLISECONDS

This command is activated by pressing the less-than key. Each time the less-than key is pressed, the computer will decrease the MicronEye's exposure time by 10 milliseconds. Keeping the less-than key pressed down continually will cause the exposure time to be decreased by some multiple of 10 milliseconds.

5.3.7 DECREASE EXPOSURE TIME BY 1 MILLISECOND

This command is activated by pressing the comma key (unshifted less-than key). Each time the comma key is pressed, the computer will decrease the MicronEye's exposure time by 1 millisecond. Keeping the comma key pressed down continually will cause the exposure time to be decreased by several milliseconds.
5.3.8 INCREASE EXPOSURE TIME BY 10 MILISECONDS

This command is activated by pressing the greater-than key. Each time the greater-than key is pressed, the computer will increase the MicronEye's exposure time by 10 milliseconds. Keeping the greater-than key pressed down continually will cause the exposure time to be increased by some multiple of 10 milliseconds.

5.3.9 INCREASE EXPOSURE TIME BY 1 MILISECOND

This command is activated by pressing the period key (unshifted greater-than key). Each time the period key is pressed, the computer will increase the MicronEye's exposure time by 1 millisecond. Keeping the period key pressed down continually will cause the exposure time to be increased by several milliseconds.

5.3.10 LOAD PICTURE FROM DISK

A picture that was previously taken by the MicronEye and saved to disk can be displayed on the computer's screen by using this command. The load command is invoked by pressing the "L" key.

The computer will then ask for the name given the picture when it was stored to disk. If the computer can find the file on disk, the picture will be displayed until a key is typed on the keyboard. Otherwise, an error message will be displayed and the computer will resume displaying pictures from the MicronEye. If you simply press the RETURN key when prompted for a file name, then the computer will resume displaying pictures.

5.3.11 PRINT PICTURE ON EPSON

The "P" key causes the current picture being displayed to be printed on an Epson printer. The screen dump routine is adequate for most purposes. There are other commercially available screen dump routines with multiple options which can also be used for printing pictures created with the MicronEye.
5.3.12 SAVE PICTURE TO DISK

Typing an "S" when the MicronEye is operating tells the computer to save the current picture to disk. The computer will prompt for a filename and attempt to save the picture to disk. If an error is encountered attempting to save the picture (usually due to insufficient disk space) then a message is displayed. Otherwise the picture is stored to disk.

5.3.13 QUIT

You can exit the MICRONEYE program by pressing the "Q" key. If you no longer wish to operate the MicronEye, select this option.

5.3.14 COMMAND SUMMARY

The following list summarizes the commands which can be used to control the MicronEye:

E -- Enhanced picture (256 x 128 image)
N -- Normal picture (128 x 64 image)
B -- Black and white imaging (bi-level)
G -- Grey level imaging (4-level)
L -- Load picture from disk
S -- Save picture to disk
P -- Print picture on printer
1 -- Use one array of the OpticRAM
2 -- Use both arrays of the OpticRAM
< -- Decrease exposure time by 10 milliseconds
, -- Decrease exposure time by 1 millisecond
> -- Increase exposure time by 10 milliseconds
. -- Increase exposure time by 1 millisecond

5.4 THE ASSEMBLY LANGUAGE CONNECTION (MEYE6510.EX)

The MICRONEYE program discussed above is a simple four line BASIC program that calls the assembly language program MEYE6510.EX. MEYE6510 loads into address $C000 (12*4096) of memory. Before the routine is called the "limit of memory pointer" at location 55-56 should be set to $2000. The MEYE6510 program uses all memory above this for screen storage and the MEYE6510 program itself. Set the limit with the BASIC instruction:

POKE 56,2*16 : POKE 55,0 : CLR
To help the user here is a general list of the way memory is allocated by the MEYE6510 program. All values are expressed in hexadecimal:

- $2000-$3FFF  This area is used for the hi-res screen.
- $4000-$7FFF  This area is the buffer used to receive the image from the MicronEye.
- $8000-$BFFF  This area is used by the ENHANCED mode. The final image to be displayed is stored here.
- $C000-$CAF7  This is where MEYE6510 resides.
- $E000-$FFFF  This area is used to store the incoming image from the MicronEye when in ENHANCED mode.

The MEYE6510 program was written with the intention that other users could write their own programs in BASIC and manipulate the MicronEye via calls to the various subroutines provided. The program was also designed to be extensible so that additional functions can be added as desired. A description of each of the primary subroutines available in MEYE6510 follows. The source listing is well-documented and should be referred to if you wish to make use of the more primitive subroutines which are not described below.

The number in parenthesis after the routine name is the decimal number that would be used in the SYS command to call the routine.

- **ONEARRAY (49869):** Sets necessary parameters to operate MicronEye in 1-PICTURE mode.
- **TWOARRAY (49895):** Sets necessary parameters to operate MicronEye in 2-PICTURE mode.
- **ENHANCED (49947):** Sets necessary parameters to operate MicronEye in ENHANCED mode.
- **NORMAL (49918):** Sets necessary parameters to operate MicronEye in NORMAL mode.
- **BANDW (49583):** Sets necessary parameters to operate MicronEye in BLACK AND WHITE mode.
- **CSHADE (49477):** Sets necessary parameters to operate MicronEye in GREY mode.
- **CRAISE (49511):** Decrease exposure time by 10 milliseconds. The exposure time can be set directly by POKEing
USING THE MICRONEYE WITH THE COMMODORE 64
THE ASSEMBLY LANGUAGE CONNECTION (MEYE6510.EX)

EXPOSURE TIME / 256 into location 682 and POKEing
EXPOSURE TIME mod 256 into location 681.

CRAS1 (49502): Decrease exposure time by 1 millisecond.

CLOWER (49551): Increase exposure time by 10 milliseconds.

CLOW1 (49542): Increase exposure time by 1 millisecond.

DSAVE (51014): Saves the current picture at location $2000-$4002 onto the disk drive. The routine exits hi-res mode, moves the picture to $8000, asks the user for a filename, and then attempts to save the picture to disk.

DLOAD (51086): The user is asked for a filename and the program then attempts to load the file. The error channel is not checked. The parameter set up at the time the picture was saved becomes the new parameter setup for the MicronEye. This routine assumes the hi-res screen is located at $2000.

SDUMP (51238): Dumps the hi-res picture to an Epson or Epson-workalike graphics printer.

UPDATE (50238): Updates the mode display (top 2 lines of hi-res screen) to reflect the current parameter settings and exposure time.

ENMODE (49671): When in ENHANCE mode, use this routine to get a picture from the MicronEye and display it on the screen.

GETIT (49594): When in NORMAL mode, use this routine to get a BLACK AND WHITE picture.

SHDBIT (49608): When in NORMAL mode, use this routine to get a GREY picture.

TEXTMD (50931): Exits hi-res mode. Use when exiting from program so that the text screen shows like it should in BASIC.
CHAPTER 6
USING THE RS-232 MICRONEYE CAMERA

6.1 HARDWARE REQUIREMENTS

There are four lines running between the RS-232 MicronEye camera and the computer -- transmit, receive, ground and 5V. The RS-232 MicronEye provides a standard DB-25P connector for interfacing. The pinout for the connector is as follows:

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Transmit data (from MicronEye)</td>
</tr>
<tr>
<td>3</td>
<td>Receive data (from computer)</td>
</tr>
<tr>
<td>7</td>
<td>Signal ground</td>
</tr>
<tr>
<td>11</td>
<td>5V (from computer) must drive 50mA load</td>
</tr>
</tbody>
</table>

A general purpose 6850 ACIA buffered serial-to-parallel and parallel-to-serial interface chip is used by the MicronEye to transfer data to the computer. There are five lines running between the camera and the computer -- transmit, receive, ground, 5V, and external clock signal.

Standard RS-232 pinouts do not provide power to the RS-232 MicronEye camera. The user must supply a 5V DC, 50mA power source on pin 11 of the DB-25P connector. This can be done by tapping the 5V supply on the computer or by using a separate voltage source. In either case, pin 7 is used as the ground line. Be certain that the power source is only 5V. Voltages in excess of 6V can permanently damage the MicronEye.

With power supplied to the cable and the cable attached to the camera and computer, the MicronEye camera is ready to operate.

The interface between the MicronEye and the computer is an RS-232 serial link. The connection is via a 6-line telephone cable. The lines are used for Vcc, ground, receive and transmit. The unused lines are not connected. The operating speed is controlled by the baud rate jumper setting on the MicronEye.
When using real-time image processing the programmer must make certain that the time required to perform special tasks between bytes does not exceed the time available. For example, a 9600 baud transmission rate means that 960 bytes per second will be transmitted. The user can therefore expect to receive a new byte from the MicronEye every 1041 microseconds. Some computer configurations may require that the baud rate be slowed to guarantee receipt of data. Many single-user systems should be able to increase the baud rate to 19,200 without problem.

The standard RS-232 MicronEye is shipped with the following interface configuration:

- One start bit
- Eight data bits
- One stop bit
- 9600 baud

This configuration applies to both transmit and receive lines. The RS-232 MicronEye camera operates only in an asynchronous mode.

6.2 SOFTWARE

Appendix E contains the complete assembly language driver used for the IBM PC version of the MicronEye. It is annotated throughout and provides a reasonable baseline for developing sophisticated drivers for the MicronEye for microcomputer and minicomputer systems. The 8088 assembly language has an architecture somewhat similar to the Z-80 microprocessor.

Please note that the IBM is a version B interface with respect to bit ordering while the RS-232 is a version A interface. Keep this in mind when performing shift and rotate instructions.

It is probable that a great deal of the code included in Appendix E will not be required for specific applications. The software does demonstrate communication techniques between the computer and MicronEye, enhancement techniques for the 256 x 128 image, 2-bit grey scale, and a printer dump routine for the Epson dot matrix graphics printer.

It is not necessary to use exclusively assembly language when working with the MicronEye. The use of higher level languages is more than appropriate if the code executes with adequate speed. The only time-critical code is the loop that receives an image from the MicronEye. By using the annotated listing it should be fairly easy to translate the various routines into higher level languages.
CHAPTER 7
HOW YOUR COMPUTER TALKS TO THE MICRONEYE

7.1 MICRONEYE VERSIONS

This section explains how to talk to the MicronEye and how to get information back from the MicronEye. We strongly recommend that users who are interested in developing their own assembly language drivers for the MicronEye study this section along with the assembly language routines included on the MicronEye diskette. We feel that the assembly language routines we have prepared are fairly complete and would advise the user to first determine that they would not be adequate for their needs before developing their own assembly language programs from scratch.

As you are aware, there are presently five different versions of the MicronEye, each specially designed to interface with a particular computer -- the Apple II, IBM-PC, Commodore 64, TRS-80 Color Computer and RS-232. Insofar as hardware configuration, the Apple II, Commodore 64 and RS-232 are similar and can be categorized together for purposes of this section. They will be referred to as "Version A" systems. The IBM-PC and TRS-80, likewise, are similar and will be referred to in this section as "Version B" systems. The difference between the the Version A and Version B systems is in the arrangement of the data bits. In Version A, the least significant bit represents the leftmost image pixel in the byte. In version B, the most significant bit represents the leftmost image pixel in the byte. This affects both commands being transmitted to the MicronEye and data being received from the MicronEye. The reason for the difference lies in the way the various computers display graphic information.

7.2 THE SERIAL CONNECTION

The interface between the MicronEye and the computer is a serial link utilizing a Motorola 6850 ACIA. The connection is via a 6-line telephone cable. The lines are used for Vcc, ground, receive, transmit, and external clock. The 6th line is not connected. The operating speed is controlled by the baud rate jumper setting on the
MicronEye circuit board. When using real-time image processing, the programmer must make certain that the time required to perform special tasks between bytes does not exceed the time available.

The ACIA is composed of a data register and a status register. Writing to the status register allows the user to configure items such as parity, stop bits, start bits, clocking, etc. Before accessing the MicronEye, the ACIA has to be initialized to the proper configuration, as follows:

<table>
<thead>
<tr>
<th>VERSION A</th>
<th>VERSION B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write to status register:</td>
<td>Write to status register:</td>
</tr>
<tr>
<td>hex $03</td>
<td>hex $C0</td>
</tr>
<tr>
<td>followed by</td>
<td>followed by</td>
</tr>
<tr>
<td>hex $14</td>
<td>hex $28</td>
</tr>
</tbody>
</table>

The first byte performs a master reset on the ACIA, while the second byte specifies that the transmission protocol is 1 start bit, followed by 8 data bits, followed by 1 stop bit; and a x1 clock mode is to be used. (x1 clock mode requires that an external clock accompany the data to and from the computer which is furnished by the standard MicronEye interface card. NOTE: RS232 users will need to use a hex $15 to select the x16 clock mode which will allow the MicronEye and computer to communicate without the clock signal from the MicronEye.

Reading the status register allows the user to determine when new data has been received and when the ACIA is ready to send data. The status bits, when set, mean:

<table>
<thead>
<tr>
<th>VERSION A</th>
<th>VERSION B</th>
<th>STATUS BIT DESCRIPTION (READ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit 0</td>
<td>Bit 7</td>
<td>Data has been received from MicronEye. In normal use, this bit is only checked when seeing if data is available from the MicronEye.</td>
</tr>
<tr>
<td>Bit 1</td>
<td>Bit 6</td>
<td>A command may be sent to the MicronEye.</td>
</tr>
<tr>
<td>Bit 4</td>
<td>Bit 3</td>
<td>Received data improperly framed. Usually only used in a debug mode.</td>
</tr>
<tr>
<td>Bit 5</td>
<td>Bit 2</td>
<td>Data received before previous byte read. Usually only used in a debug mode.</td>
</tr>
</tbody>
</table>

Once the status register indicates that a command can be sent to the MicronEye, write the command to the data register. Conversely, when receiving an image from the MicronEye, read the data register...
when the status register indicates that data is available. When receiving an image from the MicronEye, it is a good idea to incorporate a timeout mechanism in case the MicronEye stops sending bytes before the program expects. Otherwise the program can hang if the software misses even a single byte.

7.3 COMMAND DEFINITIONS

The MicronEye has several operating modes. The command byte is organized as follows:

<table>
<thead>
<tr>
<th>VERSION A</th>
<th>VERSION B</th>
<th>COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit 7</td>
<td>Bit 0</td>
<td>Always 1</td>
</tr>
<tr>
<td>Bit 6</td>
<td>Bit 1</td>
<td>Always 1</td>
</tr>
</tbody>
</table>
| Bit 5     | Bit 2     | 0 = Even rows and columns only (ALTBIT)  
|           |           | 1 = All pixels in array (NOALTBIT)    |
| Bit 4     | Bit 3     | 0 = Double send each pixel (WIDEPIX) 
|           |           | 1 = Send normally (NARROWPIX)         |
| Bit 3     | Bit 4     | 0 = 7-bit data bytes for Apple (7BIT)  
|           |           | 1 = 8 data bits per byte (8BIT)       |
| Bit 2     | Bit 5     | 0 = transmit 1 array (1ARRAY)      
|           |           | 1 = transmit upper and lower array (2ARRAY) |
| Bit 1     | Bit 6     | 0 = refresh instead of soak (REFRESH) 
|           |           | 1 = soak instead of refresh (SOAK)    |
| Bit 0     | Bit 7     | 0 = Send the requested image (SEND)  
|           |           | 1 = Don't send -- soak or refresh (NOSEND) |

7.3.1 ALTBIT And NOALTBIT MODES

The MicronEye will transmit only the pixels from the even-numbered rows and columns in the array. Because of the placement of the pixels in the image sensor, this mode will usually produce an image of clearer resolution than the NOALTBIT mode unless the image undergoes the enhancements discussed elsewhere in this manual. Software is provided on your disk that performs this enhancement.
HOW YOUR COMPUTER TALKS TO THE MICRONEYE
COMMAND DEFINITIONS

With NARROWPIX and ALTBIT the image from the MicronEye is 128 x 64. With WIDEPIX and ALTBIT the image sent to 256 x 64. NARROWPIX and NOALTBIT causes a 256 x 128 image to be transmitted. WIDEPIX and NOALTBIT causes a 512 x 128 image to be sent.

7.3.2 WIDEPIX AND NARROWPIX MODES

The MicronEye will "double transmit" each pixel in the array when WIDEPIX is selected. Since each image sensing element in the IS32 OpticRAM is twice as wide rectangular in shape, "double transmitting" maintains the proper width to height ratio for displaying the image. There are many applications, however, where maintaining the proper ratio is less important than receiving the image as quickly and compactly as possible. In such a situation NARROWPIX would be the appropriate mode choice.

7.3.3 7BIT AND 8BIT MODES

The Apple computer is somewhat peculiar in its implementation of high resolution graphics. The most significant bit of each byte on the graphics page is reserved as the 'color' bit, while the other 7 bits are the pixels being displayed. In 7BIT mode, the MicronEye transmits data so that it is compatible with the Apple's high resolution format; or, in other words, 7 bits of image pixels per byte.

The alternative to 7BIT mode is 8BIT mode. 8BIT mode causes the MicronEye to transmit in normal bitmap format (all 8 bits in the byte contain image data). 8BIT mode is used by all computers other than the Apple. For non-display use on the Apple the 8BIT mode can be useful. (The GREYPIC program in the Apple software uses both the 8BIT mode and 7BIT mode as it creates grey-scale images for the Epson).

7.3.4 1ARRAY AND 2ARRAY MODES

The image sensor used by the MicronEye is comprised of dual 128 x 256 pixel arrays. If you remove the camera lens and look at the image sensor, you can clearly see the two arrays. Using the 1ARRAY mode, only the image focused on the lower array is transmitted from the MicronEye. On the other hand, using the 2ARRAY mode causes both arrays to be transmitted from the MicronEye. The 2ARRAY mode has a split screen effect because of the spacing between the two arrays in the image sensor chip. In addition, the sensitivity to light of the two arrays is usually noticeably different. These two factors tend to make 2ARRAY mode inappropriate for many applications.
7.3.5 REFRESH AND SOAK MODES

The MicronEye takes a picture much like any other camera. The MicronEye must have the proper amount of light to make the image develop properly. Too much light will overexpose the image, while too little light will underexpose the image.

There are two ways to control the amount of light that the MicronEye sees. First, the available light can be increased or decreased. Second, the 'shutter speed' or 'exposure time' of the MicronEye can be increased or decreased. Exposure time is determined by how long the user allows the image sensor to be exposed to light without 'refreshing' the device. Simply put, refreshing the image sensor causes each cell in the image sensor to be written to 5 volts. If the image sensor is not continually refreshed, then the light focused on each cell causes the voltage in each cell to leak away at a rate proportionate to the intensity of the light. When the image sensor is not being refreshed, we say it is 'soaking.' Allowing the image sensor to soak for a longer period of time allows the MicronEye to 'see' better in dimmer light.

When the REFRESH mode is selected, the MicronEye keeps the image sensor refreshed while it is sending an image. When SOAK mode is invoked, the MicronEye "soaks" while it is transmitting an image. When the MicronEye completes the transmission of an image, it automatically goes into 'soak' mode until another command is received. In REFRESH mode, exposure time is determined by how long the MicronEye is allowed to soak before it is sent a command to transmit the image. In SOAK mode, exposure time is equal to how long the MicronEye is allowed to soak plus the time required to transmit the image. Therefore, if the exposure time exceeds the transmission time, a great deal of time can be conserved by setting the SOAK mode. If the required exposure time is less than the transmission time, then the MicronEye will have to use the REFRESH mode to inhibit 'soaking' during transmission of the image, thereby restricting the exposure time to the time the MicronEye is allowed to soak between the transmission of each image.

For example, if the desired exposure time was 600 milliseconds and the transmission time for an image was 120 milliseconds then the user could send the image with SOAK set and wait 480 milliseconds at the completion of each image transmission to create a 600 millisecond exposure time. Alternately, the user could send the image with REFRESH set and wait 600 milliseconds at the completion of each image transmission.

Note that when the MicronEye is commanded to perform the SOAK command, it continuously reads cell location R0 C0, causing the OpticRAM internal circuitry to continuously refresh all cells in Row 0. Therefore, when using the SOAK command, the cells in Row 0 will not be light sensitive. This is not a problem when using 1ARRAY mode.
HOW YOUR COMPUTER TALKS TO THE MICRONEYE
COMMAND DEFINITIONS

because ARRAY mode uses the lower array which is comprised of rows 128 to 255.

7.3.6 SEND MODE

When a command is sent to the MicronEye with SEND mode selected, the MicronEye will begin transmitting an image. In nearly all cases, the command sent to the MicronEye will have the SEND mode selected. The only time SEND mode is not desirable is the situation where a significant amount of processing must take place between transmission of images. In this situation, a user may choose to receive an image from the MicronEye, send a command to the MicronEye and REFRESH without sending, go away and process the image, send a command to the MicronEye to SOAK without sending, wait for the desired exposure time, and then send a command to have the MicronEye transmit the image.

When the MicronEye is sent a command with the SEND bit clear, the MicronEye begins transmitting an image to the computer. After the image has been sent, the MicronEye stops transmitting data, goes into a soak state, and waits for a new command. When the MicronEye is sent a command with the SEND bit set to 1 then the camera will not transmit data and will refresh or soak depending on the setting of the REFRESH bit.

Please note that when the SEND bit is set to 1, the ALTBIT bit should also be set to 1. Failure to do so will cause the first row of the subsequent image to be offset by 1 pixel.

7.4 EFFECTS OF COMMAND MODE COMBINATIONS

The following table shows the effects of different commands to the MicronEye. The REFRESH/SOAK and SEND/NOSEND bits are not considered for purposes of this table.
HOW YOUR COMPUTER TALKS TO THE MICRONEYE
EFFECTS OF COMMAND MODE COMBINATIONS

### VERSION A SYSTEMS

<table>
<thead>
<tr>
<th>COMMAND (HEX)</th>
<th>(DEC)</th>
<th>ROWS</th>
<th>BYTES PER ROW</th>
<th>PIXELS PER ROW</th>
<th>MODE SELECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0</td>
<td>192</td>
<td>64</td>
<td>37</td>
<td>256</td>
<td>ALT WIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>C4</td>
<td>196</td>
<td>128</td>
<td>37</td>
<td>256</td>
<td>ALT WIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>C8</td>
<td>200</td>
<td>64</td>
<td>32</td>
<td>256</td>
<td>ALT WIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>CC</td>
<td>204</td>
<td>128</td>
<td>32</td>
<td>256</td>
<td>ALT WIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>D0</td>
<td>208</td>
<td>64</td>
<td>19</td>
<td>128</td>
<td>ALT NOWIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>D4</td>
<td>212</td>
<td>128</td>
<td>19</td>
<td>128</td>
<td>ALT NOWIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>D8</td>
<td>216</td>
<td>64</td>
<td>16</td>
<td>128</td>
<td>ALT NOWIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>DC</td>
<td>220</td>
<td>128</td>
<td>16</td>
<td>128</td>
<td>ALT NOWIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>E0</td>
<td>224</td>
<td>128</td>
<td>73</td>
<td>512</td>
<td>NOALT WIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>E4</td>
<td>228</td>
<td>256</td>
<td>73</td>
<td>512</td>
<td>NOALT WIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>E8</td>
<td>232</td>
<td>128</td>
<td>64</td>
<td>512</td>
<td>NOALT WIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>EC</td>
<td>236</td>
<td>256</td>
<td>64</td>
<td>512</td>
<td>NOALT WIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>F0</td>
<td>240</td>
<td>128</td>
<td>37</td>
<td>256</td>
<td>NOALT NOWIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>F4</td>
<td>244</td>
<td>256</td>
<td>37</td>
<td>256</td>
<td>NOALT NOWIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>F8</td>
<td>248</td>
<td>128</td>
<td>32</td>
<td>256</td>
<td>NOALT NOWIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>FC</td>
<td>252</td>
<td>256</td>
<td>32</td>
<td>256</td>
<td>NOALT NOWIDEPIX 8BIT 2ARRAY</td>
</tr>
</tbody>
</table>

### VERSION B SYSTEMS

<table>
<thead>
<tr>
<th>COMMAND (HEX)</th>
<th>(DEC)</th>
<th>ROWS</th>
<th>BYTES PER ROW</th>
<th>PIXELS PER ROW</th>
<th>MODE SELECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
<td>03</td>
<td>64</td>
<td>37</td>
<td>256</td>
<td>ALT WIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>07</td>
<td>07</td>
<td>128</td>
<td>73</td>
<td>512</td>
<td>NOALT WIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>0B</td>
<td>11</td>
<td>64</td>
<td>19</td>
<td>128</td>
<td>ALT NOWIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>0F</td>
<td>15</td>
<td>128</td>
<td>37</td>
<td>256</td>
<td>NOALT NOWIDEPIX 7BIT 1ARRAY</td>
</tr>
<tr>
<td>13</td>
<td>19</td>
<td>64</td>
<td>32</td>
<td>256</td>
<td>ALT WIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>17</td>
<td>23</td>
<td>128</td>
<td>64</td>
<td>512</td>
<td>NOALT WIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>1B</td>
<td>27</td>
<td>64</td>
<td>16</td>
<td>128</td>
<td>ALT NOWIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>1F</td>
<td>31</td>
<td>128</td>
<td>32</td>
<td>256</td>
<td>NOALT NOWIDEPIX 8BIT 1ARRAY</td>
</tr>
<tr>
<td>23</td>
<td>35</td>
<td>128</td>
<td>37</td>
<td>256</td>
<td>ALT WIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>27</td>
<td>39</td>
<td>256</td>
<td>73</td>
<td>512</td>
<td>NOALT WIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>2B</td>
<td>43</td>
<td>128</td>
<td>19</td>
<td>128</td>
<td>ALT NOWIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>2F</td>
<td>47</td>
<td>256</td>
<td>37</td>
<td>256</td>
<td>NOALT NOWIDEPIX 7BIT 2ARRAY</td>
</tr>
<tr>
<td>33</td>
<td>51</td>
<td>128</td>
<td>32</td>
<td>256</td>
<td>ALT WIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>37</td>
<td>55</td>
<td>256</td>
<td>64</td>
<td>512</td>
<td>NOALT WIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>3B</td>
<td>59</td>
<td>128</td>
<td>16</td>
<td>128</td>
<td>ALT NOWIDEPIX 8BIT 2ARRAY</td>
</tr>
<tr>
<td>3F</td>
<td>63</td>
<td>256</td>
<td>32</td>
<td>256</td>
<td>NOALT NOWIDEPIX 8BIT 2ARRAY</td>
</tr>
</tbody>
</table>
APPENDIX A

BAUD RATE MODIFICATION

The MicronEye's transmission speed (baud rate) is normally set at the factory to 153,000 baud for the IBM, Apple and Commodore 64 computers. A baud rate of 76,800 is used for the TRS-80 Color Computer. The RS-232 version is factory set to 9,600 baud. If you wish to change the baud rate, proceed according to the following paragraph.

Modifying the baud rate of the MicronEye requires some soldering so caution is advised. The baud rate on the MicronEye is set by soldering two wire jumpers as specified below. An end of one wire has been soldered to pad 9 (located beside IC D1) and one end of the other wire has been soldered to pad 10 (located between IC D5 and IC E5). These two connections are never changed so they do not have to be removed. However, the other end of both of the wires should be unsoldered and resoldered according to the desired baud rate.

Beside IC B5 are 8 pads. Pads 1, 4, and 8 are labeled on the board. To select one of the five standard baud rates, use the table below to rejumper for the desired baud rate.

<table>
<thead>
<tr>
<th>TO SELECT THIS BAUD RATE</th>
<th>CONNECT THE WIRE FROM PAD 9 TO</th>
<th>CONNECT THE WIRE FROM PAD 10 TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>153,600</td>
<td>PAD 5</td>
<td>PAD 1</td>
</tr>
<tr>
<td>76,800</td>
<td>PAD 6</td>
<td>PAD 2</td>
</tr>
<tr>
<td>9,600</td>
<td>PAD 7</td>
<td>PAD 3</td>
</tr>
<tr>
<td>4,800</td>
<td>PAD 4</td>
<td>PAD 6</td>
</tr>
<tr>
<td>300</td>
<td>PAD 8</td>
<td>PAD 8</td>
</tr>
</tbody>
</table>
BAUD RATE MODIFICATION

Other baud rates are obtainable by soldering the wires to certain pins on IC B5. However, this practice does not make a reliable connection and is not recommended. If one of the baud rates listed below is required, use the following table to make the proper connections:

<table>
<thead>
<tr>
<th>TO SELECT THIS BAUD RATE</th>
<th>CONNECT PAD 9 TO IC B5</th>
<th>CONNECT PAD 10 TO IC B5</th>
</tr>
</thead>
<tbody>
<tr>
<td>38,400</td>
<td>PIN 3</td>
<td>PIN 9</td>
</tr>
<tr>
<td>19,200</td>
<td>PIN 2</td>
<td>PIN 7</td>
</tr>
<tr>
<td>2,400</td>
<td>PIN 12</td>
<td>PIN 3</td>
</tr>
<tr>
<td>1,200</td>
<td>PIN 14</td>
<td>PIN 2</td>
</tr>
<tr>
<td>600</td>
<td>PIN 15</td>
<td>PIN 4</td>
</tr>
</tbody>
</table>
APPENDIX B

TRANSMISSION TIME CONSIDERATIONS

The following table outlines the milliseconds required to send an image from the MicronEye to the computer as a function of rows, bytes per row and baud rate. The table may prove useful in doing exposure time calculations. The times are calculated using the following equation:

\[
\text{TIME} = \frac{\text{ROWS} \times \text{BYTES-PER-ROW} \times 10000}{\text{BAUD-RATE}}
\]

TRANSMISSION RATE TABLE

The following table is provided as an aid to the programmer by listing all row and column combinations (excluding send and soak bits).

<table>
<thead>
<tr>
<th>ROWS PER ROW</th>
<th>BYTES BAUD RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>300 9600 19200 76800 153600</td>
<td></td>
</tr>
<tr>
<td>64 16</td>
<td>34133</td>
</tr>
<tr>
<td>64 19</td>
<td>40533</td>
</tr>
<tr>
<td>64 32</td>
<td>68267</td>
</tr>
<tr>
<td>64 .37</td>
<td>78933</td>
</tr>
<tr>
<td>128 16</td>
<td>68266</td>
</tr>
<tr>
<td>128 19</td>
<td>81066</td>
</tr>
<tr>
<td>128 32</td>
<td>136533</td>
</tr>
<tr>
<td>128 37</td>
<td>157867</td>
</tr>
<tr>
<td>128 64</td>
<td>273067</td>
</tr>
<tr>
<td>128 73</td>
<td>311466</td>
</tr>
<tr>
<td>256 32</td>
<td>273067</td>
</tr>
<tr>
<td>256 37</td>
<td>315733</td>
</tr>
<tr>
<td>256 64</td>
<td>546133</td>
</tr>
<tr>
<td>256 73</td>
<td>622933</td>
</tr>
</tbody>
</table>

B-1
APPENDIX C

TROUBLESHOOTING

If you have problems with your MicronEye there is a good chance that the problem is setup-related. If you encounter a problem with your MicronEye, run through this checklist of common setup problems to verify that your MicronEye has been setup properly:

1. Verify that the card is plugged in properly. Symptoms of this problem include a peppered pattern on the screen, or an all white screen that doesn't go away even when you cover the lens, or the computer just 'hanging' when it attempts to send a command or receive an image.

2. Make certain that the lens cap is off and that the aperture setting is not set to 'C' (closed).

3. If the display appears to be all black, set the f-stop to the lowest setting and aim the MicronEye at a light source. If any of the screen turns white then the problem may be exposure time related.

4. If the display is all white, try setting the f-stop up or reducing the exposure time.

5. Try turning off the computer, brush off the interface card with a soft brush, clean the fingers on the card with propanol alcohol and a cotton swab, reinstall the card, and power up the computer.

6. Make certain that the cable connecting the MicronEye and the interface card is firmly in place.

7. If you are still unable to make the MicronEye operate properly, please contact us at Micron:

MICRON TECHNOLOGY, INC.          TEL. (208) 383-4040
VISION SYSTEMS GROUP               TWX 910-970-5973
2805 E. COLUMBIA RD.              BOISE, ID 83706
D.1 OPERATION

The heart of the MicronEye is the IS32 OpticRAM, developed and manufactured by Micron Technology, Inc. The integrated circuit is Micron's 64K Dynamic RAM assembled in a standard 16 pin ceramic DIP package with a clear glass lid. The IS32 is composed of 65,536 individual image sensing elements called pixels. These pixels are organized into two arrays of 128 rows and 256 columns. (Typical applications will utilize only one of the sensor arrays since the arrays are separated by an optical non-light sensing zone of amplifiers). Each of the elements in the IS32 is a light sensitive capacitor which can be accessed randomly by simply strobing in the appropriate row and column address of the particular element to be accessed.

The device operates by focusing the reflected light from an object onto the 32,768 light sensitive elements of the array. Light striking a particular element will cause the capacitor, which is initially precharged to a fixed voltage, to discharge toward zero volts. The capacitor will discharge at a rate proportional to both the intensity and duration it is exposed to light.

To determine if a particular element is black or white, the user would read the appropriate row and column address associated with the physical location of that particular element. The IS32 would read the voltage value of the capacitor and perform a digital comparison between the voltage of the capacitor and the fixed threshold voltage. The output pin of the IS32 would be set to a logic level of 1 if the voltage on the capacitor was above the threshold point. It would set the output to a logic level of 0 if it was below the threshold voltage.

The logic level of 0 will be associated with a white pixel. A logic level of 1 will be associated with a black pixel. A white pixel indicates the capacitor was exposed to a light intensity sufficient to discharge the capacitor past the threshold point. A black pixel indicates the light intensity was not enough to discharge the
capacitor past the threshold, therefore it retained the charge and is read as a logic 1.

The other significant factor affecting the discharge of the light sensitive capacitors is the length of the time which the capacitors are exposed to light. This period of time is measured from the initial exposure of an element until the time the particular element is read or refreshed.

The combination of the light intensity and the scan rate (the amount of time the elements are exposed before being read) will determine the optimum imaging environment. The faster the elements are scanned, or read, the greater the light intensity is required.

Perhaps the most important consideration the user must keep in mind is that the MicronEye requires a high contrast scene in order to image the object onto the IS32. Unlike a TV camera which can respond to "shades of gray," the IS32 is a digital chip where each picture element will only respond to a dark/light (1/0) binary part of the scene around an arbitrary amount of light used as a threshold. Shades of gray can be achieved by averaging multiple scans together using either a different threshold voltage or varying the scan rate. By changing the threshold voltage, keeping both the image and light intensity constant, the outputs produced during each scan will not change where pixels are definitely black or white. Change will be exhibited where the image is gray and the amount of reflected light striking the capacitors is near the threshold voltage. If an area of the image is a dark shade of gray, the output will generate more logic level 1's than logic level 0's. Where the image is a lighter shade of gray the output will generate more logic 0's than logic 1's. By averaging these outputs over a number of scans, the appropriate shade of gray is produced.

The nominal threshold with pin 1 open is 2.1 volts. This threshold can be adjusted via pin 1 from 1.5 volts to 3.0 volts. It is suggested that gray scale capability be achieved by varying the scan rate rather than adjusting the threshold voltage. By varying the scan rate (varying the discharge time) you can more accurately achieve gray scale capability.

If for any reason you must remove the IS32 from its socket, caution is imperative. The IS32 is susceptible to static and can be damaged by static electricity. Removal of the IS32 from the Bullet may require that the tips of the chip extractor tool be bent out slightly to accommodate the narrowness of the Bullet housing. When reinserting an IS32 into the socket, be certain it is properly oriented. For the Bullet, the IS32 is oriented properly when the red edge of the ribbon cable is on the same side of the Bullet as the Pin 1 notch on the IS32. For the Camera, the IS32 is oriented properly when the Pin 1 notch on the OpticRAM is on the same edge as the Pin 1 notch on the other IC's in the camera.
D.2 IS32 TECHNICAL SPECIFICATIONS

There are two versions of the IS32 OptiCRAM: the IS32 and IS32A. Beginning in September 1983, the IS32 was replaced in favor of the IS32A. The only difference between the two devices is size. The IS32A is exactly 20 percent smaller in the horizontal and vertical dimensions. The dimensions below are for the IS32A. To calculate dimensions for the larger IS32 device, multiply by 1.25.

D.2.1 DIMENSIONS

1. ARRAY: 128 x 256 electrical addressable elements per array (4420 microns x 876.8 microns). The physical organization of the array is actually a 514 x 129 grid with staggered cell placement as indicated in figure D-1.

2. ROW: 877 microns.

3. COLUMN: 4420 microns.

4. ELEMENT SIZE: 6.4 microns vertical by 6.4 microns horizontal.

5. VERTICAL PITCH (Row Pitch): 6.8 microns.

6. HORIZONTAL PITCH (Column Pitch): 8.6 microns.

7. SPACING between left and right array: 120 microns.

8. DISTANCE from surface of OpticRAM chip to top of the glass = 940 microns (plus or minus 100 microns).

D.2.2 SENSITIVITY

Broad band sensitivity of the IS32 OptiCRAM is approximately 2\mu J/sq cm.

Silicon detectors have a useful optical sensitivity over the region of the spectrum in which silicon absorbs photons. This extends from 200 nanometers to 1100 nanometers. However, a complete characterization of the IS32 is still under way. The sensitivity follows the silicon characteristic curve since the IS32 is built using silicon. The IS32 is impervious to damage by high light intensity. It has a high quantum efficiency and a binary output that is
proportional to the amount of incident light and integration time (referenced to a threshold). However, oversaturation of the IS32 by more than 4 F-stops will, for the duration of oversaturation, make the first half of the array all light and the other half all dark. This is only a temporary situation for the duration of the saturation. The IS32 is sensitive up to near UV.

The IS32 chip is mounted in the package with 20 mils tolerance in both the X and Y axis. This suggests that if an OpticRAM package is replaced in a camera, a physical realignment of the camera to the scene is necessary. The tolerance from surface of the array to the lens mount from camera to camera is 20 mils with a 6 degree rotational tolerance.

D.3 TOPOLOGY

D.3.1 Address Descramble

If you access a cell (pixel) in the OpticRAM using an address of zero for both the row and column, the Optic RAM will not physically select Row 0 and Column 0. This is because the internal address decoding does not provide a one-to-one correspondence between the address count and the physical row and column. A simple circuit, consisting of a 7486 and a 7404, performs the necessary code conversion to achieve the desired one-to-one correspondence. See Figure D-3

D.3.2 Pixel Layout

One of the primary goals in designing a low cost integrated circuit such as the OpticRAM, is to minimize its physical size. To achieve this goal, the cells in the OpticRAM are arranged in an interleaved pattern. If an image is read out of the OpticRAM by counting successively down the rows and columns, the image will look "fuzzy" around the edges because the pixels will be slightly misplaced in the graphics matrix.

To accommodate the pixel misplacement, the data from the OpticRAM must be mapped into the graphics matrix so that the arrangement of the pixels in the graphic matrix matches the physical arrangement of the cells in the OpticRAM. Due to the interleaved cell pattern on the OpticRAM, the array is much longer than it is wide, resulting in spaces between the cells in the column direction. Because of the spaces, the 128 X 256 array of cells will map very nicely into a 129 X 514 matrix. We will call this matrix the Cell Placement Grid.
D.3.3 Cell Placement Grid

The cell placement grid is shown in Figure D-2 below. For a single array, there are a total of 129 rows and 514 columns. Only the corners of the array are shown. The placement grid indicates where the information from each cell in the OpticRAM should be mapped. For instance, if the cell at address Row 1, Column 1, in the OpticRAM is read, the value (a 1 or 0) should be placed in the placement grid at location X=2, Y=3.

When every cell has been read and the values placed in the appropriate locations, about half of the grid remains empty. We will call these empty locations "space pixels." The space pixels can be set all high or all low to provide a light or dark background for the image. Another alternative is to set each space pixel to the level that agrees with the majority of its nearest neighbors. For example, let's say the pixel at grid locations X=2, Y=2 (R1 C1) and X=3, Y=1 (R1 C0) are high, and the pixel at grid location X=3, Y=3 (R3 C0) is low. These are the three nearest neighbors of grid location X=3, Y=2. The majority of these nearest neighbors is high, so the previously empty grid location X=3, Y=2 is set high also. This technique can be applied to all empty grid locations except those near the edge of the array. A modified technique can be used for these edge space pixels, although there is less optical data to work with. Another alternative is to simply not use the edge rows and columns.

Having the cells laid out in the IS32 the way they are, gives the IS32 much greater resolving power than if the cells were laid out linearly.
Figure D-1. IS32 OptiCRAM Physical Layout.
Figure D-2. IS32 OpticRAM Cell Spacing.

Figure D-3. IS32 OpticRAM Address Descramble Logic.
APPENDIX E

ANNOTATED ASSEMBLY LANGUAGE DRIVER FOR THE IBM PC
This driver module takes care of all the necessary arrangements for getting an image from the MicroEye to the graphics page in memory of the IBM PC. The routine is assembled to be relocatable.

Because the routine normally resides within the BASIC workspace, the first part of the BASIC program locates these routines as high in the BASIC segment as possible. Although the program makes certain that there is enough room initially for both the assembly language routines and the BASIC program, there is no assurance that the declaration of large amounts of data space will not overlap the machine language programs.

There are 4 assembly routines available to the user--PARAMCALC, SCREENDUMP, SFRSCH and GETPIC. PARAMCALC takes the setup parameters entered at the main menu and calculates the commands that SETCALC will send the MicroEye and the number of bytes the MicroEye will return. SCREENDUMP dumps the current picture to an IBM or Epson printer. SFRSCH saves the picture in the WORKSP area to the screen. The proper calling sequence from a BASIC program is as follows:

```plaintext
DEF DEF(GM-Bxxx) where xxx specifies the assembler routine address
BLOAD "PARAMCALC"
BIFCALC = 0
PARAMCALC = &
SCREENDUMP = 12
SFRSCH = 16
DATAAREA = 20
...
CALL PARAMCALC(pic_type,pics_per_screen,exit_at_end,expose_time)
CALL SCREENDUMP
CALL GETPIC(screen_start,BSTATE,PSTATE,key_value)
CALL SFRSCH
```

All variables in the argument list are assumed to be of type INTEGER. The variables are defined as follows:

- **PIC_TYPE** -- determines the format of the image transmitted from the MicroEye. The following are valid command:
  
  - 0: 128 x 64 picture (black & white)
  - 1: 512 x 64 picture (gray)
  - 2: 128 x 64 picture (gray)
  - 3: 512 x 128 picture (black & white)
  - 4: 512 x 128 picture (gray)
  - 5: 640 x 128 picture (gray)

- **PICS_PER_SCREEN** -- if set to 2 then both blocks of image sensors on the OptiCAM will be transmitted from the MicroEye. The blocks of (or arrays) are separated by a dead zone so the displayed picture will appear to be split-screen.

- **EXIT_AT_END** -- if true (an odd number) then control is returned to the calling program at the end of each picture transmission. Otherwise, pictures are continually processed until a key is pressed.

- **EXPOSE_TIME** -- the number of milliseconds for which the image should be exposed.

- **SCREEN_START** -- the byte position on the screen page at which the picture should start. This position is calculated as:

  ```plaintext
  (ROW+1) * COLUMNS
  ```

  where `ROW` must be an even number between 0 and 134, Column must be even number between 0 and 512 and divisible by 8.

---

**SCREENDUMP**

The SCREENDUMP sub-routine takes a BASIC program and adds the commands that will save the current picture in the WORKSP area to the screen. The proper calling sequence is:

```plaintext
CALL SCREENDUMP
```

**GETPIC**

The GETPIC sub-routine saves the current picture to the WORKSP area. The proper calling sequence is:

```plaintext
CALL GETPIC(screen_start,BSTATE,PSTATE,key_value)
```

---

**PARAMCALC**

This sub-routine takes the parameters entered at the main menu and calculates the commands that will transmit the current picture to the MicroEye. The proper calling sequence is:

```plaintext
CALL PARAMCALC(pic_type,pics_per_screen,exit_at_end,expose_time)
```
GETPIC PROC
RET

GETPICT PROC
GETPICT
RET

PARMCALC PROC
RET

PICDUMP PROC
RET

SCRIFER PROC
RET

BITMAP DB 10400 DUP (11)

CONTR DB 2D8H

DATAIN DB 209H

DATAOUT DB 209H

PHECT DB 24H

PHTAB DB 16000

RMTAB DB 2048

RHTAB DB 4096

RDPRT DB 8192
SCREENRIP ROUTINE -- prints image in WORKBUF buffer

MOVECR -- Move picture to graphics screen

RET

; SUBTIL FRAMEGRABBER ROUTINES -- gets image from MicronEye

PAGE 2-3

PAGE 3-1
the routine takes the 75x 120 image from the Advance KieronEye and converts it to a 512 x 128 image with properly placed pixels.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCEroutine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.

The ENHANCE routine does the following:

- Moves the offset buffer to the new position.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
- Shifts the bits of the new buffer.
- Updates the positions in the DB and DI registers.
The 180 Personal Computer Assembly 07-19-83
PAGE 4-2
MICROLOGIC ASSEMBLER ROUTINES (continued)

GREYAD ROUTINE for 512 x 64 image

PAGE 4-3
MICROLOGIC ASSEMBLER ROUTINES (continued)

FILLIN ROUTINE for 640 x 128 picture

The 180 Personal Computer Assembly 07-19-83
PAGE 4-6
MICROLOGIC ASSEMBLER ROUTINES (continued)

FILLIN ROUTINE for 640 x 128 picture

The IBM Personal Computer Assembly 07-19-83
PAGE 4-8
MICROLOGIC ASSEMBLER ROUTINES (continued)

FILLIN ROUTINE for 640 x 128 picture

The IBM Personal Computer Assembly 07-19-83
PAGE 4-10
MICROLOGIC ASSEMBLER ROUTINES (continued)

FILLIN ROUTINE for 640 x 128 picture

The IBM Personal Computer Assembly 07-19-83
PAGE 4-12
MICROLOGIC ASSEMBLER ROUTINES (continued)

FILLIN ROUTINE for 640 x 128 picture
ENHANCE ROUTINE lor 640 I

The IBM Personal Computer Assembler 07-19-83 PAGE 4-7
ASSEMBLER ROUTINES (REFERENCE)

ENHANCE ROUTINE for 640 x 128 picture

The IBM Personal Computer Assembler 07-19-83 PAGE 4-8
ASSEMBLER ROUTINES (REFERENCE)

GREYADD ROUTINE for 640 x 128 picture
The IBM Personal Computer Assembler 07-19-83
MICROASSEMBLY ROUTINES (MCM:EMM)

Segments and groups:

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Align</th>
<th>Combine</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSEG</td>
<td>256B</td>
<td>PARA</td>
<td>PUBLIC</td>
<td></td>
</tr>
</tbody>
</table>

Symbols:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Value</th>
<th>Attr</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACICLR</td>
<td>N PROC</td>
<td>408</td>
<td>CSEG</td>
</tr>
<tr>
<td>ARRICT</td>
<td>L WORD</td>
<td>5750</td>
<td>CSEG</td>
</tr>
<tr>
<td>BEEP</td>
<td>N PROC</td>
<td>8724</td>
<td>CSEG</td>
</tr>
<tr>
<td>BITCT</td>
<td>L BYTE</td>
<td>6175</td>
<td>CSEG</td>
</tr>
<tr>
<td>BITMAP</td>
<td>L BYTE</td>
<td>2862</td>
<td>CSEG</td>
</tr>
<tr>
<td>BITMAP2</td>
<td>L BYTE</td>
<td>5168</td>
<td>CSEG</td>
</tr>
<tr>
<td>BYTES</td>
<td>L WORD</td>
<td>6508</td>
<td>CSEG</td>
</tr>
<tr>
<td>CLEAR</td>
<td>L WORD</td>
<td>6560</td>
<td>CSEG</td>
</tr>
<tr>
<td>CMD</td>
<td>N PROC</td>
<td>6798</td>
<td>CSEG</td>
</tr>
<tr>
<td>CNL</td>
<td>L WORD</td>
<td>6412</td>
<td>CSEG</td>
</tr>
<tr>
<td>CMCL</td>
<td>L WORD</td>
<td>6417</td>
<td>CSEG</td>
</tr>
<tr>
<td>CMD</td>
<td>L WORD</td>
<td>6894</td>
<td>CSEG</td>
</tr>
<tr>
<td>COMMAND</td>
<td>L WORD</td>
<td>6418</td>
<td>CSEG</td>
</tr>
<tr>
<td>CONTROL</td>
<td>L WORD</td>
<td>6420</td>
<td>CSEG</td>
</tr>
<tr>
<td>CPU</td>
<td>L BYTE</td>
<td>6432</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATE</td>
<td>L WORD</td>
<td>6436</td>
<td>CSEG</td>
</tr>
<tr>
<td>DMCH</td>
<td>L WORD</td>
<td>6438</td>
<td>CSEG</td>
</tr>
<tr>
<td>DME</td>
<td>L BYTE</td>
<td>6432</td>
<td>CSEG</td>
</tr>
<tr>
<td>DPAL</td>
<td>L BYTE</td>
<td>6418</td>
<td>CSEG</td>
</tr>
<tr>
<td>EDM</td>
<td>L WORD</td>
<td>6429</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN1</td>
<td>L WORD</td>
<td>6420</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN2</td>
<td>N PROC</td>
<td>6466</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN3</td>
<td>N PROC</td>
<td>4443</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN4</td>
<td>L WORD</td>
<td>6467</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN5</td>
<td>L WORD</td>
<td>6469</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN6</td>
<td>L WORD</td>
<td>6116</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENUN7</td>
<td>N PROC</td>
<td>6469</td>
<td>CSEG</td>
</tr>
<tr>
<td>FEZ</td>
<td>L WORD</td>
<td>6472</td>
<td>CSEG</td>
</tr>
<tr>
<td>FILLINZ</td>
<td>N PROC</td>
<td>6469</td>
<td>CSEG</td>
</tr>
<tr>
<td>FLDT</td>
<td>L WORD</td>
<td>6428</td>
<td>CSEG</td>
</tr>
<tr>
<td>FRAMEBAR</td>
<td>N PROC</td>
<td>4328</td>
<td>CSEG</td>
</tr>
<tr>
<td>GI</td>
<td>L WORD</td>
<td>6116</td>
<td>CSEG</td>
</tr>
<tr>
<td>GETPIC</td>
<td>N PROC</td>
<td>4180</td>
<td>CSEG</td>
</tr>
<tr>
<td>GETPTC</td>
<td>N PROC</td>
<td>4180</td>
<td>CSEG</td>
</tr>
<tr>
<td>GNEMO5</td>
<td>L BYTE</td>
<td>4185</td>
<td>CSEG</td>
</tr>
<tr>
<td>GNEMO6</td>
<td>L BYTE</td>
<td>4185</td>
<td>CSEG</td>
</tr>
<tr>
<td>GNEMO7</td>
<td>L BYTE</td>
<td>4185</td>
<td>CSEG</td>
</tr>
<tr>
<td>GRENH</td>
<td>N PROC</td>
<td>6466</td>
<td>CSEG</td>
</tr>
<tr>
<td>GREYH2</td>
<td>N PROC</td>
<td>6464</td>
<td>CSEG</td>
</tr>
<tr>
<td>GREY4</td>
<td>N PROC</td>
<td>6464</td>
<td>CSEG</td>
</tr>
<tr>
<td>GREY5</td>
<td>N PROC</td>
<td>6474</td>
<td>CSEG</td>
</tr>
<tr>
<td>GRIC</td>
<td>L WORD</td>
<td>6109</td>
<td>CSEG</td>
</tr>
<tr>
<td>GSPTR</td>
<td>L BYTE</td>
<td>1932</td>
<td>CSEG</td>
</tr>
<tr>
<td>HKEY</td>
<td>L WORD</td>
<td>6460</td>
<td>CSEG</td>
</tr>
</tbody>
</table>
| KEY | L WORD | 6460 | CSEG | }

The IBM Personal Computer Assembler 07-19-83
MICROASSEMBLY ROUTINES (MCM:EMM)

Symbols:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Attr</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARRAY</td>
<td>255</td>
<td>CSEG</td>
</tr>
<tr>
<td>BITAP</td>
<td>104</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENHANCE</td>
<td>2</td>
<td>CSEG</td>
</tr>
<tr>
<td>ENDSTR</td>
<td>5</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAOUT</td>
<td>5</td>
<td>CSEG</td>
</tr>
<tr>
<td>SETPIC</td>
<td>2</td>
<td>CSEG</td>
</tr>
<tr>
<td>CONTROL</td>
<td>510</td>
<td>CSEG</td>
</tr>
<tr>
<td>CLEARS</td>
<td>1024</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAIN</td>
<td>256</td>
<td>CSEG</td>
</tr>
<tr>
<td>CHK</td>
<td>256</td>
<td>CSEG</td>
</tr>
<tr>
<td>NOCC</td>
<td>61</td>
<td>CSEG</td>
</tr>
<tr>
<td>IC</td>
<td>4</td>
<td>CSEG</td>
</tr>
<tr>
<td>INI</td>
<td>1024</td>
<td>CSEG</td>
</tr>
<tr>
<td>BITAP</td>
<td>104</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAIN</td>
<td>256</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAOUT</td>
<td>5</td>
<td>CSEG</td>
</tr>
<tr>
<td>SETPIC</td>
<td>2</td>
<td>CSEG</td>
</tr>
<tr>
<td>CONTROL</td>
<td>510</td>
<td>CSEG</td>
</tr>
<tr>
<td>CLEARS</td>
<td>1024</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAIN</td>
<td>256</td>
<td>CSEG</td>
</tr>
<tr>
<td>DATAOUT</td>
<td>5</td>
<td>CSEG</td>
</tr>
<tr>
<td>SETPIC</td>
<td>2</td>
<td>CSEG</td>
</tr>
</tbody>
</table>

Warning: Severe
Errors: 0 0

PAGE Symbols-1

PAGE Symbols-2

MICROASSEMBLY ROUTINES (MCM:EMM)
GUIDE TO OPTICS SELECTION AND LIGHTING TECHNIQUES

F.1 LIGHTING CONSIDERATIONS FOR THE IS32 OPTICRAM

The IS32 OpticRAM lends itself to profiling scenes and component parts by imaging the dimension to be measured onto a matrix of light sensors where each light sensor is equal to some distance in physical space.

The MicronEye Camera needs a high contrast scene in order to image the object into the IS32. Unlike a TV camera which can respond to shades of gray, the IS32 is a digital chip where each picture element makes a black/white judgement based on an arbitrary light level used as a threshold (trip light level). Portions of the scene that are lighter than the threshold level will be judged as white while portions of the scene darker than the threshold level will be judged as black.

For example, if the trip light level is made lighter, then a new slice of the scene would be captured around that light threshold. One can look at shades of gray as planes of binary light level slices. One example: 64 gray scales means 64 binary light level slices.

The trip light level can be changed in one of three ways:

1. Changing the exposure time.
2. Changing the f-stop on the lens.
3. Changing the light on the scenes itself.

Doubling the exposure time is the same as opening the f-stop by one stop, (changing the f-stop to the next smaller number), or in other words doubling the amount of light. Contrast can now be defined as a minimum difference between adjacent slices. Example: In taking 64 gray scale slices there is normally only one slice where the adjacent slice is of a minimum difference.
GUIDE TO OPTICS SELECTION AND LIGHTING TECHNIQUES
LIGHTING CONSIDERATIONS FOR THE IS32 OPTICRAM

High contrast means that there are more than two adjacent slices that are about the same (usually three or more adjacent slices are about the same).

F.1.1 FRONT LIGHTING

Front lit scenes, where the camera is on the same side of the scene as the light source or ambient light, usually is low in contrast. In this situation extreme care in setting up uniform lighting on the scene is necessary and the optimum trip light level needs to be used. Front lighting requires a multiple diffused light source such that the contrast in the scene is increased. If defects or points of interest are to be emphasized, side lighting such that the defects or points of interest cast a shadow, or increase in spectral energy (reflection) will usually point out the defects.

To set up a front lit scene, normally one or more flood lamps (outdoor flood lamps purchased from a local hardware store are adequate) are arranged around the scene far enough away so that there are no shadows. Then the f-stop, focus and lamps are adjusted for maximum contrast and focus. Adjust the focus where the smallest part of the scene has the most detail. The depth of focus (the distance the scene can move in relation to the camera and still be in focus) is increased at higher f-stops. Increase the amount of light and/or the integration time to optimize the result.

A trade-off of lighting, integration time, f-stop and scene-to-camera positioning (also lens selection) is necessary to optimize the result. Due to light falling off (at a slope of \( \cos^4 \)) from the center of the lens going to the edges of the lens, the periphery of a scene takes more light for a uniform trip light threshold to capture the scene.

F.1.2 BACK LIGHTING

For a backlit scene, the light comes from behind the scene so that the object being viewed is shadowed into the camera. Backlighting the object, for maximum contrast will give the best repeatable results. Backlighting is recommended if the camera is used to measure the object or certain aspects of the object and/or for part recognition since the trip light level can move a large amount without degrading the results.

The backlit light source must be large enough so that the camera, without the object in the field of view will see a uniform amount of light. This is normally accomplished by using several flood lamps and shining the flood lamps onto a diffused surface (ground glass, or
diffused white plastic, or frosted mylar), such that a uniform light source is created. Placing the object between the diffused surface and the camera will shadow the object into the camera with maximum contrast. Adjust the f-stop to the maximum value that the amount of light and integration time will allow. NOTE: For non-contact measurement of the objects' size, the magnification changes in relation to its distance from the camera to the object.

In selecting a lens, the magnification change as the object moves in the Z axis must be considered. The farther the lens is from the object the less the size changes as the object moves in the Z axis. The equation that relates the Z axis motion of the object to the change in lens-to-object distance is:

\[ Z = \text{change in object motion to/from the camera} \]
\[ L = \text{Lens to object distance;} \]
\[ \% \text{area change} = 200 \times (Z/L + Z^2/L^2) \]

For example, if the Z axis motion is 1/2 inch and the lens to object distance is 20 inches, then the change in size of the scene, as the computer sees it, is 5.25% in area. In comparing the MicronEye camera, lighting and processing, to other industrial systems that do gray scale processing, where lighting is not a dominant factor, there is usually a 300 to 1 cost trade-off. Placing more emphasis to correct the lighting so that a single threshold can be used produces a saving of 300 times.

F.1.3 ILLUMINATION SOURCES

Some of the common illumination sources are tungsten, quartz halogen, quartz iodine, fluorescent, and mercury or xenon arc lamps, as well as various flash lamps, lasers and LED sources. The common ways to configure these sources are: 1) illumination of the scene, 2) backlighting (shadowing) of the scene or 3) a combination of both, depending on the type of information desired from the camera. See figure F-1.
The light intensity required by the image sensor must be well defined in order to have even illumination of the scene, since the camera uses a common threshold for the entire scene, calling it light or dark. Only a small portion of light from the light source, via the scene, actually ends up in the sensor. Therefore, in choosing a suitable light source, such factors as even illumination versus threshold, f-stop and magnification of the lens, and the surface of the object (light or dark, diffused or specular) must be considered. Certain sections of the object may require spotlights to create an even illumination where a meaningful threshold scene can be produced. The amount of light coming through the lens is increasingly attenuated as the angle between the center of the lens going to the edge of the lens increases.

F.2 OPTICS

The MicronEye comes standard with a C-mount lens. Special applications may require the use of other lenses or filters which are not of the C-mount variety. C-mount adaptors are available for the more common lens types discussed below.
F.2.1 LENS TYPES

Three common lens types are the C-mount series, U-mount series, and L-mount series.

F.2.1.1 The C-mount Lens - The C-mount has a flange focal distance of 17.526mm (.690”). The flange focal distance is the distance from the lens mounting flange to the convergence point of all parallel rays entering the lens when the lens is focused at infinity. The C-mount lens is the work horse of the TV camera world.

Its format is designed for performance over the diagonal of a standard television camera videocon. This lens was selected by Micron because of its popularity and ease of availability. The mounting thread characteristics are: 1" diameter, 32 threads/inch (machine thread information 1"-32um2A).

Generally, this lens is an excellent choice for the OptiRAM. However, due to geometric distortion and field angle characteristics, short focal length lenses should be evaluated as to suitability for metrology (measurement) imaging. For instance, an 8.5mm focal length lens should not be used with an image sensor greater than 1/8" in length (the OptiRAM is .174") if the application involves metrology. Also, the majority of lenses should not be used wide open because of the light falloff characteristics.

The lens-to-OptiRAM distance has been established by using the flange focal distance dimension for fixed focal length lenses (non-adjustable focus). For close-ups, lens extenders will be required. The lens extender is used behind the lens to increase the lens to OptiRAM distance.

\[ \text{Spacer Lens (in mm)} = \frac{\text{Focal Length}}{\text{Magnification}} \]

For a given lens, as magnification increases the distance between lens and focal plane decreases. Figure F-2 contains graphs of object distance versus magnification for common C-, U-, and L-mount lenses. These charts are a useful “ballpark” guide for lens focal length selection.

F.2.1.2 The U-mount Lens - The U-mount lens is a focusable lens having a flange focal distance of 46.52mm (1.7913”). The characteristic of the mounting threads is M42x1. This lens was primarily designed for 35mm photography applications. A C-mount to U-mount adapter can be purchased from most camera stores.
F.2.1.3 The L-mount Lens - The L-mount lens is a fixed-focus flat-field lens designed for committed industrial applications. This lens was originally designed for photographic enlargers. The flange focal distance is a function of the specification of each lens selected.

F.2.1.4 Microscope Lenses - There are standard microscope lenses available. These are to be used in applications where a magnification of less than one is required. However, a microscope lens to C-mount adapter in most cases needs to be individually designed because generally long lens extenders are needed.
GUIDE TO OPTICS SELECTION AND LIGHTING TECHNIQUES

Figure F-2
F.2.2 TERMS AND DEFINITIONS

ARRAY SIZE: The physical size of the OpticRAM array from the 1st to the last pixel. The size can be looked at from many points of view. Care must be exercised in how the scene is projected onto the array via the optics.

Example: From the 1st pixel to the last pixel the column size = 174.016 mils and the row size of either section = 34.52 mils. The row dimension of the total array (of both arrays plus space pixels) = 73.764 mils.

FIELD OF VIEW (FOV): The maximum image dimension plus an allowance for alignment and part variation.

FOCAL LENGTH (F): Type of lens, defined in millimeters. The present lens that is shipped with the camera is a 16mm C-mount lens.

F-STOP: The opening of the iris on the lens is calibrated in f-stops. Each higher number requires twice the light on the object for the same amount of light falling on the array.

LENS TO IMAGE DISTANCE (S'): The distance from the lens to the image (scene).

LENS TO OPTICRAM DISTANCE (S): The distance from the shoulder of the lens mount to the surface of the integrated circuit inside the OpticRAM package (plane of best focus). A lens extender may be required for objects that are closer to the lens than the normal lens design dictates.

MAGNIFICATION (M): A camera lens is a transformation device that will make the image projection onto the array either smaller or larger depending on the lens and the distance away from the lens. The ratio of the object's true size to the size of the projection on the array is called the magnification.

PIXEL COUNT: A count of the number of pixel pitches that an aspect of the image traverses on the array, directly proportional to the magnification. In image space each pixel pitch represents a minimum resolution (image resolution).

RESOLUTION: The smallest size that is of interest in the field of view of the camera. The resolution is pixel pitch times the magnification.

Z AXIS CHANGE: The change in the distance between the camera and the object. As the distance between the scene and the camera decreases, the image projected onto the OpticRAM gets bigger, and therefore covers more pixels. As the distance between the scene
and camera increases, the image gets smaller. If the distance between the camera and the scene is closer than the lens will focus, a spacer can be inserted between the lens and the camera to extend the focus range, or a different lens may be used to enhance the focus. The spacer length formula is used to determine the size of the spacer needed.

F.2.3 USEFUL EQUATIONS

Figure F-3. Simple Lens Equations.
GUIDE TO OPTICS SELECTION AND LIGHTING TECHNIQUES

OPTICS

METRIC CONVERSIONS:
1 INCH = 25.4 Millimeters
1 INCH = 2.54 Centimeters
1 FOOT = 304.8 Millimeters
1 FOOT = .3048 Meters
1 YARD = .9144 Meters
1 Millisecond = .001 seconds (msec)
1 Microsecond = .000001 seconds (usec)

The resolution in the scene is dependent on the pixel pitch times the magnification. However, since the row pitch and the column pitch are different, this will correspond to a different magnification in the XY plane. Care must be exercised in selecting the dominant pitch.

PERCENT OF MAGNIFICATION CHANGE PER IMAGE AXIS = (Z/S') * 100

As the scene moves towards the camera, each scene axis gets bigger. As the scene moves away from the camera, each scene axis gets smaller. This equation relates the total Z axis motion (to and away motion of the scene as related to the camera) to edge change providing the scene is still in focus.

F.2.4 LENS SELECTION CONSIDERATIONS

The selection of a lens requires the consideration of many parameters such as lighting, edge sharpness of the scene, Z axis motion of the scene, and distance from the camera to the scene. The lens provides a projection of the scene into the OpticRAM. This means if the lens is not selected properly or is misadjusted (out of focus, etc.) the information that the OpticRAM sees will not adequately represent the scene, (for the threshold data slice of the scene will not represent the scene). One will be hard pressed to interpret what the camera is looking at. The choice of a lens in terms of focal length and field of view are directly affected by restrictions which may exist on the working distance of the camera. For example, a room size may restrict the camera from moving back far enough to have the scene in focus or fully captured.

The least resolvable element or increment in a measurement system may be the dominant factor, implying that more than one camera may be required in the system. In our system, with a built in threshold sensing technique, the resolution is equivalent to one pixel. The scene resolution is the pixel pitch times the object magnification.

Accuracy is the degree of exactness to which the measurement can be made. Under controlled conditions, accuracy can equal the resolution. When measuring the distance between two edges of an image, the accuracy is equivalent to one element per edge under
conditions of having a sharp optical image of the object's edge. If lesser accuracy occurs, it is usually due to an unsharp edge, created by poor contrast between the object and the background, or due to dynamic aspects of object movements and integration time. However, by averaging one edge (or edges), the accuracy can be finer than the object resolution.

The following example discusses how one would select each component part for the camera and system configuration:

A disk is to be measured for its diameter on a translucent conveyor. The conveyor speed is 15 feet per minute. The disk size is .2 inches (with .02 inches of variation) with a height variation of 40 mils. This includes the conveyor thickness variation and vibration. NOTE: The limit tolerance in relation to the nominal size is .02 inches. However, the measurement of the part may require 10 times better resolution than the limit requires, 1% in this case.

In this example we will look at two ways to implement the solution. One solution is using a strobe light while the other solution is to analyze the motion of the part as it relates to the array. Figure F-4 describes the disk on the conveyor.

![Figure F-4. Dynamics of Sample Problem](image-url)
F.2.4.1 SOLUTION 1 - The Field of View (FOV) is \( 0.2 + (0.02 \times 2) = 0.240 \) inches. This gives a tolerance of the maximum disk size with 0.01 inches on top and bottom for location variation.

Calculate the magnification using the row dimension of 129 elements (34.52 mils). This is the dominant dimension in this case since the diameter of the disk need to be contained within the field of view of the camera. The column dimension of 514 elements is 174.016 mils.

\[
M = \frac{0.240 \text{ inches}}{0.03452 \text{ inches}} = 7.0
\]

**MAGNIFICATION USING THE ROW AXIS**

Resolution in the Row Axis = \( 7.0 \times 0.26772 \text{ mils} = 1.87 \text{ mils static resolution} \). Resolution in the column axis = \( 7.0 \times 0.33858 \text{ mils} = 2.37 \text{ mils static resolution} \). However, the 174.016 mils column axis times 7.0 = 1.218 inches. The FOV window at a magnification of 7.0 in space is 0.2416 inches by 1.218 inches. This gives a lot of space for the disk to move around, yet it can still be accurately measured.

This means that if we project the OptiCRAM array into the object plane, each row axis pixel will have a pitch of 1.87 mils and each column axis pixel will have a pitch of 2.37 mils.

Using the chart for C-mount lenses (Figure F-2) for a magnification of 7.0, the lens to image distance for different lenses could be:

- 12.5 mm = 2.5"
- 16 mm = 3.75"
- 25 mm = 5.75"
- 50 mm = 14"
- 75 mm = 22"

To find the image distance, find 7.0 on the magnification axis. Follow it until it intersects the lens types and read off the values of the working distance on the other axis.

The disk height variation of 40 mils creates a change of dimension (magnification change). The percent of dimensional change is related to the height variation, divided by the lens-to-object distance times 100 ((Z/C')*100). If a lens extender is required, the extender length can be calculated by dividing the lens focal length by the required magnification. Units are in millimeters. The resulting image will focus when the lens focus control is set in its mid-point position. The following lenses can all be used to give a magnification of 7.0:
The 75mm lens will provide the least amount of magnification distortion. If there is enough physical space, then selecting the 75mm lens with a 10.7mm extender ring places the camera and lens 22" above the disk conveyor.

The dynamic property of the system is the smudge. As the part passes the field of view of the camera, the edge of the part is smudged across several pixels as the camera integrates the light entering the camera. Since the part is traveling at 15 ft. per minute, what must the integration time be so that only one pixel will be smudged? Converting feet per minute to inches per second =

\[ 15 \text{ ft/min} \times 12 \text{ inches/ft} \times \frac{1 \text{ min}}{60 \text{ sec}} = 3 \text{ inches/sec} \]

As calculated before, 1 pixel of the row dimension = 1.87 mils. This means that for each frame scan the part can only move 1.87 mils per scan and since the part travels at 3 inches/second, then:

\[ 0.00187"/\text{scan} \times \frac{1 \text{ sec}}{3"} = 0.000623 \text{ sec/scan} = 623 \text{ microsec/scan} \]

This is clearly too fast for the camera, which can operate at only 4 scans per second. What is the solution? At each scan, the disk moves:

\[ 3 \text{ inches/sec} \times 0.25 \text{ seconds/scan} = 0.75 \text{ inches/scan} \]

The part is only .24 inches in diameter. This means for every scan, the part can move approximately four times its diameter through the field of view of the camera. The solution is to place a photo transistor looking across the conveyor to an LED. As the disk blocks the LED light to the photo transistor, it triggers a strobe light that is mounted below the translucent conveyor. Select a strobe light with a flash of peak energy shorter than 613 microseconds. The setup is shown in Figure F-5.

As the strobe light flashes, it also triggers the software that brings in the camera data. The camera integration time is directly linked with the part pitch. However, care must be taken so that the integration time does not exceed where the ambient light or dark current rises above the camera threshold. If the conveyor stops or no parts come down the conveyor, this fact must be sent to the software where it will input data from the camera and throw it away (dummy...
read) to refresh the pixels to keep the camera in the alert condition. By having a photo-transistor that precedes the strobe photo-transistor, the first photo-transistor does a dummy read. This arms the camera and after the flash the camera will contain the correct data. A strobe light is an effective tool to freeze action in dynamic situations. However, in many situations a strobe light may not be required.

![Diagram of a camera setup](image)

Figure F-5. Triggering Camera Based on Part Location

F.2.4.2 SOLUTION 2 - This solution shows how to approach the problem without using a strobe light. Assume that an incandescent light is used to backlight the part, and the OpticRAM is operated at 120 frames per second, which translates to 8.33 msec/frame. The part is still moving at 3 inches/sec, as we calculated in the previous solution.

Calculate the distance over which the disk is smudged:

\[
\text{.083 sec/scan} \times 3 \text{ inches/sec} = .025 \text{ inches/scan smudge}
\]

From scan to scan, the part moves .025 inches. Therefore, the field of view needs to be the size of the part (.24") plus 2x the smudge to allow for the smudge of the leading and trailing edges. Dividing the FOV (.29") by the row dimension (.03452) we are able to calculate a magnification constant of 8.4.

Assuming that a 75mm lens was selected gives a distance of 26" from camera to scene and a deviation of .15 percent of Z-axis magnification change with a spacer of 8.9mm. The row axis resolution is determined by the product of .268 mils * 8.4 giving 2.25 mils. The column axis resolution is the product of .33858 mils * 8.4 giving 2.84 mils.
This means that each edge has a gradient (in this case) of 12 pixel smudge motion. See figure F-6. If the threshold is centered to the midpoint of the light amplitude, the 12 pixels that are smudged will go to 6 pixels (actual edge) on each side. The actual size can be realized by either changing the intensity of the lamp via a fixed threshold or by changing the threshold and holding the intensity of the lamp constant. However, since size is directly related to light versus threshold levels, the lamp output needs to be accurately stabilized.

Figure F-6. Length Measurement of a Moving Object

We have talked so far about what happens to the middle of the part, now we need to talk about what happens at the left or right edge of the part in a dynamic situation. (Refer back to Figure F-4)

Assuming the right most edge or left most edge covers a pixel, the question is, for what duration is the pixel covered? Assume from scan to scan that the disk moves .025 inches. Using the formula for a chord of a circle (Figure F-7), we need to determine the error at point A and point B.
From earlier calculations, the pixel width in the column axis of 2.84 mils, with .1 inch radius is:

\[ C_0 = \sqrt{4(2 \times 0.00284 \times 0.1 - 0.00284^2)} = 0.0476 \]

0.0476" * scan/.025" = 190% of the time the A and B pixels are dark suggesting that the error at points A and B is negligible. (190% of the time is an awkward way of saying that the disk travels only about half the distance between points A and B in one scan period. When the percentage exceeds 60%, we can say for certain that the left/right edge pixel represents the part. Motion is always a problem even in static situations because between the camera and the scene there is vibration which may require careful attention to detail.

Once data is captured either by a strobe lamp or by back lighting (shadowing) and stored in the computer memory, statistical averaging is then done in order to improve the data. EXAMPLE: Using the formula to find how many row pixels should come dark at the same time at the entry and exit. The row resolution is .00225 inches per pixel. Using the formula for a chord of a circle:

\[ C_0 = \sqrt{4(2 \times 0.00225 \times 0.1 - 0.00225^2)} = 0.042 \text{ mils} \]
.042 mils * pixel/.00225 mils = 18 pixels

This indicates that if 18 pixels are averaged at the max/min points then the resolution and accuracy can be increased by a value of:

\[ \text{SQRT(number of pixels)} / 2 \]

Find the midpoint of the circle, then average the 10 pixels on either side (20 pixels)

\[ \text{SQRT}(20)/2 = 4.5/2 = \text{approx} 2 \]

This suggests a half a digit increase in accuracy.

1.84 mils/.2 diam \* 100 = .92% + .23% for Z axis Motion = 1.15%

\[ 1.15% / 2 = .575\% \text{ resolution (after calibration).} \]

From disk to disk, one should be able to resolve each disk to about .6% The design goal was 1%. If it is desired, an out of round figure of merit can also be calculated:

\[ \text{area} = \pi \times R^2 \]
\[ \text{circumference} = 2 \times \pi \times R \]
\[ \text{area/circumference} = R/2 \]

Adding the area pixels and dividing by the edge pixels, should give a number close to half the radius pixel as a ratio. The ratio should hold. If it does not, this is an indication of out of roundness. One can also sort parts for rough out-of-round tolerances.

F.3 OTHER CONSIDERATIONS

Since backlighting is a problem on most conveyors then using a structured light may be the solution.

In general, arbitrary lighting of the environment is not acceptable because it can result in low-contrast images, specular reflections, shadowing, and extraneous details. A well-designed lighting system illuminates the scene so that the complexity of the resulting image is minimized, while the information required for
inspection or manipulation is enhanced.

Once the data of the scene is in memory, further algorithms can be employed to extract useful feature data, such as: modeling the algorithm of an object to extract the following features: area, parameters, centroid, ratio of minimum to maximum moment of inertia, axis of least moment of inertia, diagonal length of a bonding rectangle, and simple dimensional measurements at key points that can resolve a problem.

Figure F-8. Other Useful Lighting Techniques
APPENDIX G

HARDWARE DESCRIPTION

G.1 TIMING GENERATION CIRCUIT

This circuit generates the timing signals for the operation of the MicronEye. A CMOS oscillator circuit generates the basic clock signal. This signal is divided down to produce the various possible baud rates and the timing signals which drive the IS32. The baud clock signals sequence the Interrupt Generator and the Transmitter circuit.

The oscillator circuit consists of a CMOS inverter, a crystal, two resistors and two capacitors. It generates a 4.9152 Mhz signal which is buffered by an inverter (A4,pin2). This frequency is divided in half by a D flip-flop at A3-5, and again at A3,pin9. Both outputs lead to baud rate selection pads. Flip-flop output A3,pin9 also connects to the clock input at B5,pin10. IC B5 does successive frequency divide-by-twos. The various outputs lead to other baud rate selection pads. Pads 5 through 8 are baud clock signals. One of these baud clocks is used in the transmitter and Interrupt Generator circuit. Pads 1 through 4 are clock signals that are 16 times higher in frequency than the baud clocks. One of these 16x clock signals is used in the receiver circuit.

The output of B5,pin7 drives the Optic RAM timing aircuitry which generates RAS, CAS and R/W (read/write). The outputs of inverters A4,pin4 and A4,pin6 are identical. A4,pin4 drives the RAS input to the Optic Ram, and is buffered separately because it is required to drive its signal through the ribbon cable if a Bullet MicronEye is used. A4,pin 6 is identical to the RAS signal, but it is used as inputs to other camera circuitry and is labeled RAS'.

When the camera is not in an Interrupt mode (i.e., is not transmitting data from the OpticRAM), CAS and R/W are disabled. The signal INT is low and INT/ (The "/" after a signal name indicates the complement of the signal.) is high, so the AND gate driving CAS remains low and the OR gate driving R/W remains high.

G-1
During an Interrupt cycle, INT goes high and INT/ goes low, enabling CAS and R/W. RAS' goes low with RAS which latches the Row address into the OptiCRAM. RAS' passes through a delay line consisting of 2 inverters and an RC network, and then causes CAS to go low, latching the Column Address into the OptiCRAM. At this time the R/W signal is still high, so the accessed pixel is read out. After another delay period, R/W goes low, which causes the OptiCRAM to write data into the accessed cell. The addressing circuitry presents the proper data on the Data In pin to make sure that 5 volts is written back into the cell.

When RAS' goes high, the Interrupt cycle is terminated and CAS and R/W are disabled.

G.2 COMMAND RECEIVER CIRCUIT

G.2.1 General Description

The serial command line carries the camera commands from the computer to the camera. This data enters the command receiver circuit one bit at a time. The first bit to arrive is the start bit, followed by 8 data bits and then the stop bit. The start bit enables the input shift register and starts the shift register clock. The clock is initially low. When it goes high, the start bit, which is a high, is latched into the first of eight data positions in the shift register. When the clock goes low, the first data bit arrives at the shift register input. On the rising edge of the clock, the shift register "shifts" the high start bit from position 1 to position 2, and shifts the first data bit from the shift register input, into position 1. As each successive bit arrives, each one is shifted into the shift register on the rising clock edge.

When the start bit finally shifts into position 8, the camera has received all of the command information. The first six data bits are transferred from the shift register into a latch (memory) called the Command Register. The clock is disabled and the shift register is cleared. Now the six camera command bits are in the Command Register and the receiver is ready to get another command.

G.2.2 Circuit Description

The start bit from the computer appears as a high level at the output of the inverter at G1-12. The rising edge of this start bit clocks flip-flop F1-9 to the high state. This line clears the reset on IC's F2 and F4. F2 is a shift register and F4 is used as a divide-by-16 counter. F4's input is a clock whose frequency is 16 times greater than the baud rate (16x clock). After eight clock cycles, the counter
HARDWARE DESCRIPTION
COMMAND RECEIVER CIRCUIT

output (F4-11) goes high, shifting the start bit into position 1 (F2-3) of the shift register. 8 clock cycles later, the shift register clock at F4-11 goes low and the first data bit arrives. 8 clock cycles later F4-11 goes high, shifting the data bit into position 2 (F2-3), and the start bit into position 1 (F2-4). This process continues until the start bit reaches position 8 (F2-13). The high start bit causes a low at the flip-flop RESET input (F1-13). This causes the flip-flop Q/ output (F1-8) to go high, latching the serial register data into the Command Register, F3. At the same time, the flip-flop Q output (F1-9) goes low, resetting F2 and F4.

G.3 ADDRESS REGISTERS

This circuit latches the Row, Column and Refresh pointers for the OpticRAM addressing.

G.3.1 General Description

Address registers C4 and C3 hold the RAS and CAS addresses, respectively. These registers are enabled only when the camera is to fetch and transmit a single bit of information from the OpticRAM. This fetch operation is initiated by the INT signal going high, and is called an Interrupt cycle. An Interrupt cycle is started on the rising edge of RAS' and is ended on the next rising edge of RAS'.

When the camera is not in an Interrupt cycle, the Refresh Register, C2, is active. This register increments the Row Address from 0 to 255, thus performing a refresh operation on the OpticRAM.

All three Registers have tri-state outputs and only one register is active at any one time. The selected register drives its data onto a common bus called the Present Address bus. The Present Address passes through the descramble and soak circuitry, to the OpticRAM, where it is used to select a Row or Column. The Present Address bus also connects to the Address Circuit, where a value of 0, 1 or 2 is added to the Present Address value.

The resulting sum is driven out of the adder onto the Next Address bus. This bus connects to the inputs of each of the Address Registers. The value on the Next Address bus is latched into the selected Address Register and then that Register is disabled.
HARDWARE DESCRIPTION
ADDRESS REGISTERS

G.3.2 Circuit Description

When the MicronEye is not in an Interrupt mode, the INT signal is low and the INT/ signal is high. This forces the Enable inputs (active low) to C3 and C4 to remain high. When RAS' and Td go high and INT is high, the NAND gate output at A1-3 is low, enabling C2. C2 drives its data onto the Present Address bus. The data propagates to the OpticRAM and to the Adder circuit. The Adder circuit adds a 1 to the value on the Present Address bus and drives the sum onto the Next Address bus where it appears at the inputs to C2. When RAS' goes low, the descrambled Present Address is latched into the OpticRAM, and the output of A1-3 goes high, clocking the value on the Next Address bus into C2 and turning off the outputs.

During an Interrupt cycle, INT/ is low, so C2 is disabled. The rising edge of RAS' initiates the Interrupt cycle, so initially RAS' (and Td) and INT will be high, driving the NAND gate A1-8 low and enabling the Row Register, C4. C4 drives its value onto the Present Address bus. Some value, either 0,1 or 2 is added to it in the Adder and the sum is placed on the Next Address bus. When RAS' goes low, the Next Address value is latched into the Row Register, the Row Register outputs are disabled and the Column Registers outputs are enabled. The data from the Column Register, C3, is driven onto the Present Address bus, through the Adder Circuit (where it may be incremented) and onto the Next Address bus. It also propagates to the OpticRAM where it is latched when CAS goes low. When RAS' goes high, the value on the Next Address bus is latched into the Column Register and its output drivers are disabled.

The Array Selection circuit determines whether one or both arrays are transmitted. If 2ARRAY/ is high, the output of the OR gate (B4-11) is always high and the Row Register value (C4) will never be less than 128. Thus, only the second array (rows 128 to 255) will be addressed. If 2ARRAY/ is low, however, the OR gate will appear transparent and the value on the Next Address bus line D7 will drive onto C4. This means all addresses from 0 to 255 will be selected and both arrays will be transmitted.

G.4 ADDRESS DESCRAMBLE, SOAK/, AND DIN/DOUT CIRCUITS

G.4.1 Address Descramble

The internal circuitry in the OpticRAM scrambles the Row and Column Address values when accessing a cell. The Address Descramble circuit reverses the OpticRAM scramble. It transforms the Data from the Address Registers into a new address, which the OpticRAM decodes to access the desired pixel.
The circuit consists of 2 inverters, 3 Exclusive-OR's and a multiplexor (D2). The invertors and Exclusive-ORs provide the descramble function on the Row and Column addresses. The multiplexor selects between the descrambled Row and Column address at the appropriate time and drives the address to the OpticRAM. The multiplexor uses RAS' to determine which address is selected. If RAS' is high at the multiplexor SELECT input (D2-1), the B inputs, which are the descrambled Row Address inputs, are selected. When RAS' is low, the A inputs, or descrambled Column Address inputs, are selected. The descramble truth-table is available in the IS32 data sheet.

G.4.2 SOAK/

The purpose of the SOAK/ circuit is to prevent the refresh from reaching the OpticRAM. The OpticRAM is light sensitive only when it is not being refreshed. When INT is low (which is when the Refresh Register is active) and SOAK/ is low, the output of the NOR gate, B3-13, is high. This sets the multiplexor Enable input (D2-15) high and drives the multiplexor outputs low. The high NOR gate output at B3-13 also forces a low at the inverter output E3-8, which forces the outputs of the four AND gates (D4-3,6,8,11) low. Thus, the OpticRAM address inputs remain low, and the refresh function is performed only on address 0, i.e., only Row 0 gets refreshed.

When SOAK/ goes high, the multiplexor and AND gate outputs are enabled and the refresh addresses reach the OpticRAM and the entire chip is refreshed, making it insensitive to light. The SOAK/ command can be thought of as an electronic shutter control.

G.4.3 Din/Dout Circuit

This circuit controls the input to the OpticRAM Din (Data In) pin and also detects when a cell in the OpticRAM has been "exposed" to the low state.

For a cell to be light sensitive, it must be initially charged to +5 volts. This is done by writing data into the cells. Due to the operation of the OpticRAM internal circuitry, a logic "1" must be written into all cells with row addresses between 0 and 127, and a logic "0" must be written into all cells with row addresses between 128 and 255. The most significant row address bit, Q7, is latched (during interrupt cycles) by flip-flop E4 on the falling edge of RAS'. When the row address is between 0 and 127, row address bit Q7 is a 0, and when the row address is between 128 and 255, row address bit Q7 is a 1. The inverting output of flip-flop E4 (E4-8) is connected to the Data In pin on the IS32. Thus, the proper data will be presented to the OpticRAM to write each cell to +5 volts.
HARDWARE DESCRIPTION
ADDRESS DESCRAMBLE, SOAK/, AND DIN/DOUT CIRCUITS

The Exclusive-OR gate (E2-8,9,10) compares the data out of the OpticRAM with the data that was read into it. Notice that the input to the Exclusive-OR gate at E2-8 is the complement of the value at the Din pin. Thus, if the OpticRAM cell being read out is still high, the two flip-flop outputs, E4-8 and E4-9, will be at opposite levels and the output of the Exclusive-OR (E4-10), will be high. Conversely, if the cell has been exposed to the low state, the two inputs to the Exclusive-OR will be the same and its output, E4-10, will be low. The output of E4-10 propagates to the Transmitter circuit, where it is latched and transmitted to the computer.

G.5 TRANSMITTER AND INTERRUPT GENERATOR CIRCUIT

This circuit transmits the serial information, inserting start and stop bits where appropriate, and generates the INT and INT/ signals for fetching pixel information.

G.5.1 General Description

At the heart of this circuit is the ripple Counter, D1. D1 is enabled when the MicronEye has been commanded to transmit data. It inhibits the Interrupt circuit when start and stop bits are being transmitted, and enables the Interrupt circuit when it is transmitting data. The Transmitter is clocked by the baud clock. On each baud clock cycle, only one start, stop or data bit is transmitted.

The Interrupt Generator is enabled by both the ripple counter (D1) and the baud clock, but the Interrupt cycle is clocked by RAS'. Remember the purpose of the Interrupt cycle is to fetch a single pixel for transmission, and only one pixel can be transmitted on each baud clock cycle. The rising edge of the baud clock enables the Interrupt circuit. The next rising edge of RAS initiates the Interrupt cycle, causing a pixel to be read from the OpticRAM. The INT/ signal feeds back into the Interrupt circuit, resetting the Interrupt enable. When RAS' goes high again, the Interrupt cycle is terminated. The next rising edge of the baud clock will enable the Interrupt circuit again (unless a start or stop bit is to be transmitted). Thus, only one pixel is transmitted during each baud clock cycle.

The WIDEPIX circuit is used to help compensate for the 2.5 to 1 aspect ratio of the OpticRAM. If the optic data is displayed on a screen with a 1 to 1 aspect ratio, the image will appear to be squeezed in the horizontal direction. The WIDEPIX circuit helps compensate for this by causing each pixel to be transmitted twice, doubling the width of the image. The circuit is enabled when the MicronEye is transmitting and the WIDEPIX command bit is high. This causes the flip-flop output A2-5 to toggle on every baud clock cycle.

G-6
This flip-flop inhibits the Interrupt cycle on alternate baud clock cycles. During baud clock cycles in which the Interrupt is inhibited, the pixel from the previous Interrupt cycle is transmitted again.

G.5.2 Circuit Description

When the MicronEye is not in a Transmit mode, the XMIT signal is low, driving the ripple counter RESET input high (01-15). This puts the ripple counter in a reset state in which output Q0 (D1-3) is high. The high on Q0 drives the RESET input at E5-1, low and the flip-flop Q/ output E5-6, high. E5-6 is the data transmission line to the computer. The high level of Q0 (D1-3) also drives the flip-flop data input (E5-12) high (let's assume LINE is low). This prevents any Interrupt cycles from occuring.

When the MicronEye receives a Transmit command, XMIT goes high, XMIT/ goes low and the ripple counter D1 is enabled. D1 is clocked by the rising edge of the BAUD clock. The first clock causes Q0 (D1-3) to go low and Q1 (D1-2) to go high. This sets the transmit line E5-6 low, representing the start bit. The first clock also forces a high at flip-flop data input, E5-12. The baud clock is delayed through an RC network (R3 and C2) and now clocks the high input at flip-flop E5-12 to the output at E5-9. This forces a high on the input of the Interrupt flip-flop, A2-12. When RAS' goes high at the flip-flop clock input A2-11, it initiates the Interrupt cycle. INT goes high and INT/ goes low. INT/ is an input to the AND gate. B2-1 and forces the flip-flop RESET inputs (E5-13) low. This forces A2-12 low, so on the next rising edge of RAS', the Interrupt cycle is terminated. INT/ going high clears the RESET at e8-13 and another interrupt will occur when the baud clock goes high again.

When the WIDEPIX bit is set high, the RESET input at A2-1 is high, enabling the flip-flop. The output toggles on each interrupt request and inhibits every other interrupt cycle by bringing the RESET input A2-13 low.

The LINE and LINE/ signals indicate that the Column Address Register has reached terminal count. These signals inhibit further interrupts from occurring during data bit transmissions, so the value of the last accessed data bit is repeated to complete the current byte transmission. This guarantees that the next byte transmitted contains information from the next row, i.e., no single byte will contain information from two rows. When the stop bit is to be transmitted, LINE at E1-5 causes an Interrupt Request and LINE/ at A1-4 ensures that the Interrupt flip-flop is enabled. This "dummy" interrupt is used to increment the Row Address Register. The pixel that is accessed during this cycle is blanked by the transmission of the Stop bit.
G.6 ADDER AND END-OF-FRAME CIRCUIT

This circuit adds the proper increments to the Row, Column and Refresh Registers and generates signals indicating End-of-Line and End-of-Frame in the OpticRAM.

G.6.1 General Description

When any of the Address Registers drive a value onto the Present Address bus, the Adder circuit receives this value, adds a 0, 1 or 2 to it (depending on the control inputs) and drives the sum onto the Next Address bus. The control lines are RAS', LINE, ALTBIT and INT. When the Refresh Register is active, the INT line causes a "1" to be added each cycle. During interrupt cycles, the Row and Column Registers are active. The Adder sequences these registers through the OpticRAM in a column-fast mode, i.e., the Adder adds a "zero" to the Row Address and a "one" to the Column Address until the end of the column (End-of-Line) is reached. The Adder then adds a 1 to both the Row and Column, thus incrementing the Row Register and resetting the Column Register to zero.

The ALTBIT input simply adds another "1" to the value on the Present Address bus during Interrupt cycles, thus the Row and Column Registers are incremented by 2 rather than 1.

G.6.2 Circuit Description

During Refresh cycles, the INT signal is low, forcing the Carry In input to the Adder (C1-13) to be high. Thus, a value of "1" is added to the value on the Present Address bus on each Refresh cycle.

During Interrupt cycles, the INT signal is high. Let's assume LINE and ALTBIT are low. For the first half of the Interrupt cycle, the Row Register is active and RAS' is high, forcing the Carry-In input of the Adder to be low. A zero is added to the Present Address value, so the Row Register address remains unchanged.

When RAS goes low, the Column Register is active and a high is driven onto the Adder's Carry-In input. A "1" is added to the Present Address bus and the incremented value is stored back into the Column Register. Thus, the Registers count down the columns in the same row.

When the last cell is accessed, the Column Address is at the Adder's terminal count of 255, setting the carry-out signal high. (The Column Register is incremented to zero). The high Carry-Out signal is latched by the rising edge of INT/ at F1-2, and forces the outputs, LINE and LINE/ (F1-5 and 6) to the 'asserted' state. These
signals cause the next Interrupt cycle to occur during the transmission of the next stop bit. The LINE input to the Exclusive-OR at E1-2, reverses the effect of RAS' on the Adders' Carry-In input. Thus, a "1" is added to the Row Register and a "0" is added to the Column Register. The pixel that is accessed during this Interrupt is blanked by the stop bit transmission. At the start of the next Interrupt cycle (when RAS' goes high), LINE and LINE/ are reset and the circuit sequences down this next row.

Let's assume the last pixel in the OpticRAM has been accessed and LINE has been set. The Column Register has been incremented to zero and the Row Register is at terminal count (255). The next Interrupt cycle forces the Row Register to drive its value of 255 onto the Present Address bus and to the Adder. The Adder adds a "1" to it and drives a value of zero onto the Next Address bus and also sets the Carry-Out (C5-14) high. The Carry-Out and LINE signals force the output of the AND gate (B2-11) high, thus setting the flip-flop input (E4-2) high also. When RAS' goes low, the NOR gate (B3-10) goes high, clocking E4-3. The Q/ output of the flip-flop (E4-6) goes low. This is the End-of-Frame signal. The EOF is connected to the reset input of the Command Register, so a low on the EOF line resets all of the command lines to zero. The XMIT command line is connected to the flip-flop reset (E4-1), so when XMIT goes low, flip-flop E4-1 is reset and the EOF signal is reset high. Note that the Row and Column Registers both now hold a value of zero.
COMMAND RECEIVER CIRCUIT

Diagram of a command receiver circuit with various components and connections labeled with signal names and numbers.
ADDRESS REGISTER CIRCUIT

Notes:
1) G outputs - Present address bus
2) D inputs - Next address bus
ADDRESS DESCRAMBLE, SOAK, DIN/DOUT CIRCUIT

G-14
TRANSMITTER AND INTERRUPT GENERATOR CIRCUIT
WARRANTY

If a MicronEye fails to perform properly due to a defect in workmanship or material within ninety (90) days from date of purchase, Micron will repair or replace it free of charge. Should this product require service during this warranty period, return the product to Micron at the following address, transportation charges prepaid:

MICRON TECHNOLOGY, INC.
2805 E. Columbia Road
Boise, Idaho 83706

Attach to the MicronEye your name, address, telephone number, a description of the problem and proof of date of retail purchase. This warranty does not apply to defects caused by unreasonable use.

THE FOREGOING IS IN LIEU OF ALL OTHER WARRANTIES EXPRESSED OR IMPLIED. MICRON TECHNOLOGY, INC. NEITHER ASSUMES NOR AUTHORIZES ANY PERSON TO ASSUME FOR IT ANY OTHER OBLIGATION OR LIABILITY IN CONNECTION WITH THE SALE OF THIS PRODUCT. IN NO EVENT SHALL MICRON TECHNOLOGY, INC. OR ITS DEALERS BE LIABLE FOR SPECIAL OR CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF THIS PRODUCT.

WARRANTY REGISTRATION

To receive full warranty protection, YOU MUST RETURN THIS REGISTRATION WITHIN 10 DAYS AFTER DATE OF PURCHASE to Micron Technology, Inc., Attn: Sales and Marketing Department, 2805 East Columbia Road, Boise, Idaho 83706.

Purchaser's Name ____________________________________________
Address ____________________________________________________
City ___________________ State ___________ Zip ___________
Telephone Number ____________________________
Place of Purchase __________________________________________
City ___________________ State ___________ Zip ___________
Serial Number ______________________________________________
MICRON TECHNOLOGY manufactures high quality semiconductors in beautiful Southwestern Idaho. Micron's products reflect its continuing emphasis on quality products competitively priced. On-going research and development projects are geared to the continued supply of unique innovative products that are easy to use and complimentary in a wide variety of applications.

We're building our reputation on innovation.