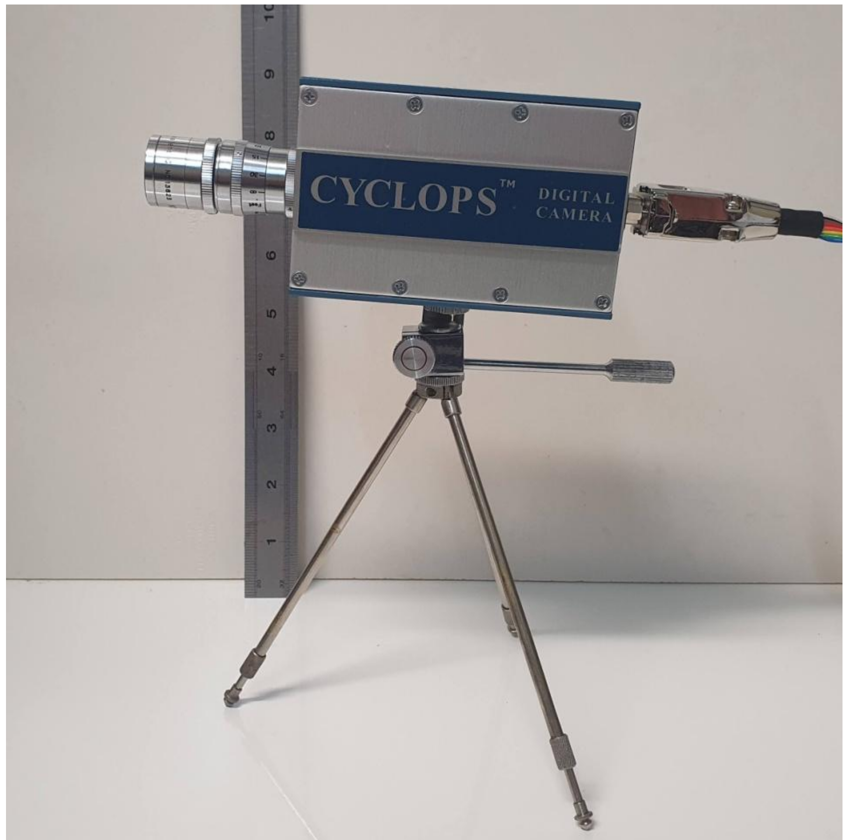
# RECREATING ONE OF THE WORLD'S FIRST DIGITAL VIDEO CAMERAS, THE CROMEMCO CYCLOPS.

**Dr. H. Holden. August, 2026.**


## BACKGROUND:

This article is a follow up to the previously published Silicon Chip article on the Cromemco's TV Dazzler. The TV Dazzler was a pair of S-100 bus computer PCB's. These carried a total of 72 TTL IC's. The system used Direct Memory Access (DMA) of an S-100 Computer's own RAM. The RAM containing the image data gets processed by the Dazzler so as to display a Video Image in NTSC color video format. The Dazzler has both Color and Monochrome video image modes available.

Terry Walker designed the Cyclops Camera while he was working on his Doctorate at Stanford University. He demonstrated the Camera to Roger Melen & Harry Garland in 1974. Roger and

Harry formed the Company Cromemco and they produced the Cyclops Solid State Video Camera in late 1974, as the company's first product.

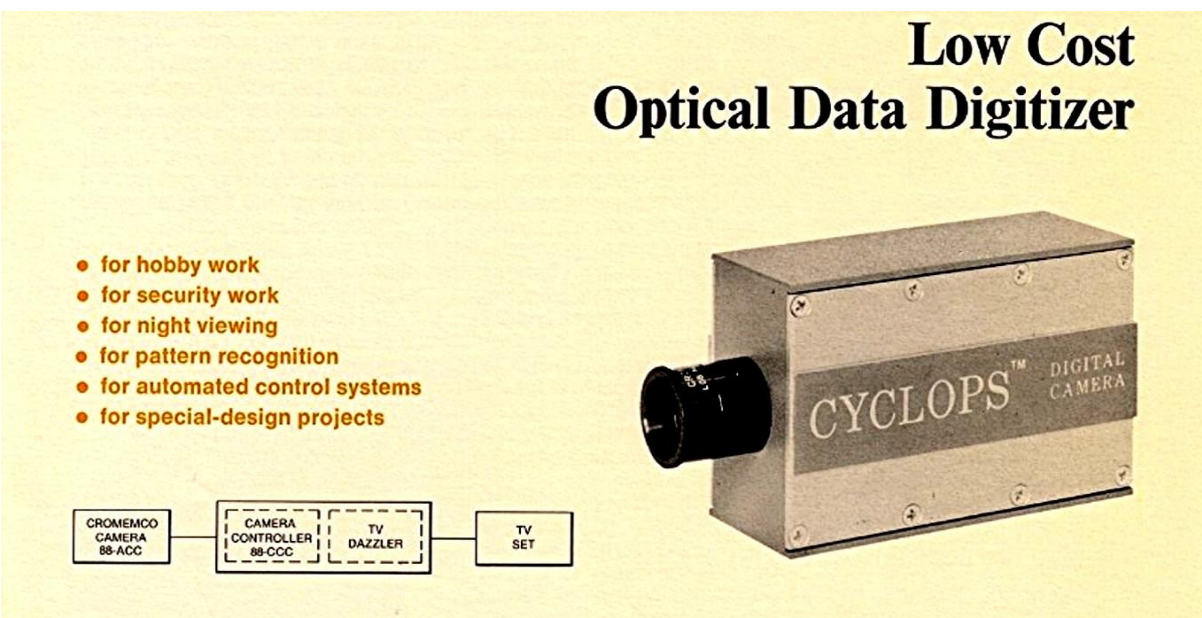
Initially the Cyclops camera was designed to feed an Oscilloscope in X-Y mode. However in late 1974, a smaller version of it, shown in the photos above was produced. It is called the "Altair Cyclops". This was designed to run in conjunction with the Altair S-100 Bus Computer.

To get the Cyclops Camera to run with a Computer, it required two S-100 camera controller boards and on top of that, also the the two S-100 Dazzler boards fitted to the host computer.

Terry Walker designed the Dazzler boards after being asked by Cromemco to make the Cyclops camera interface with a Computer, rather than an Oscilloscope. Terry did the PCB layouts for them all. He used 1mm tape on Mylar sheets to create 1mm wide tracks.

Terry also designed other products for Cromemco, including an S-100 board called a D + 7A I/O. This board contained 7 Analog I/O lines and one digital byte I/O. This board was also used later with Joystick units, made by Cromemco, to play games in conjunction with the TV Dazzler. This arrangement is the subject of another article.

The idea for the Cyclops camera was to use a MOS Dynamic RAM IC. This was a 1k x 1 bit IC, with its top cut off, to act as an optical image sensor to create a 32 x 32 pixel image. The image was focussed onto it by a common 16mm D mount Film Camera lens. A photo of the Cyclops camera advertising, showing how it combined with the Dazzler PCB's:



The Altair Cyclops Camera came as kit of parts with the image sensor IC, with its top removed. Though it appears the S-100 controller boards were provided in many cases assembled and tested. The camera itself contained three small PCB's. The camera is described in a Cromemco document titled 88-ACC. The two camera controller boards were described by Cromemco's 88-CCC document.

When the camera is combined with the two S-100 camera controller PCB's and the Dazzler pcb pair (set for 32 x 32 pixel monochrome mode) in the same S-100 host computer (typically the Altair) this created a digital video image system. The image derived from the camera head is ultimately displayed on a standard analog Monochrome VDU.

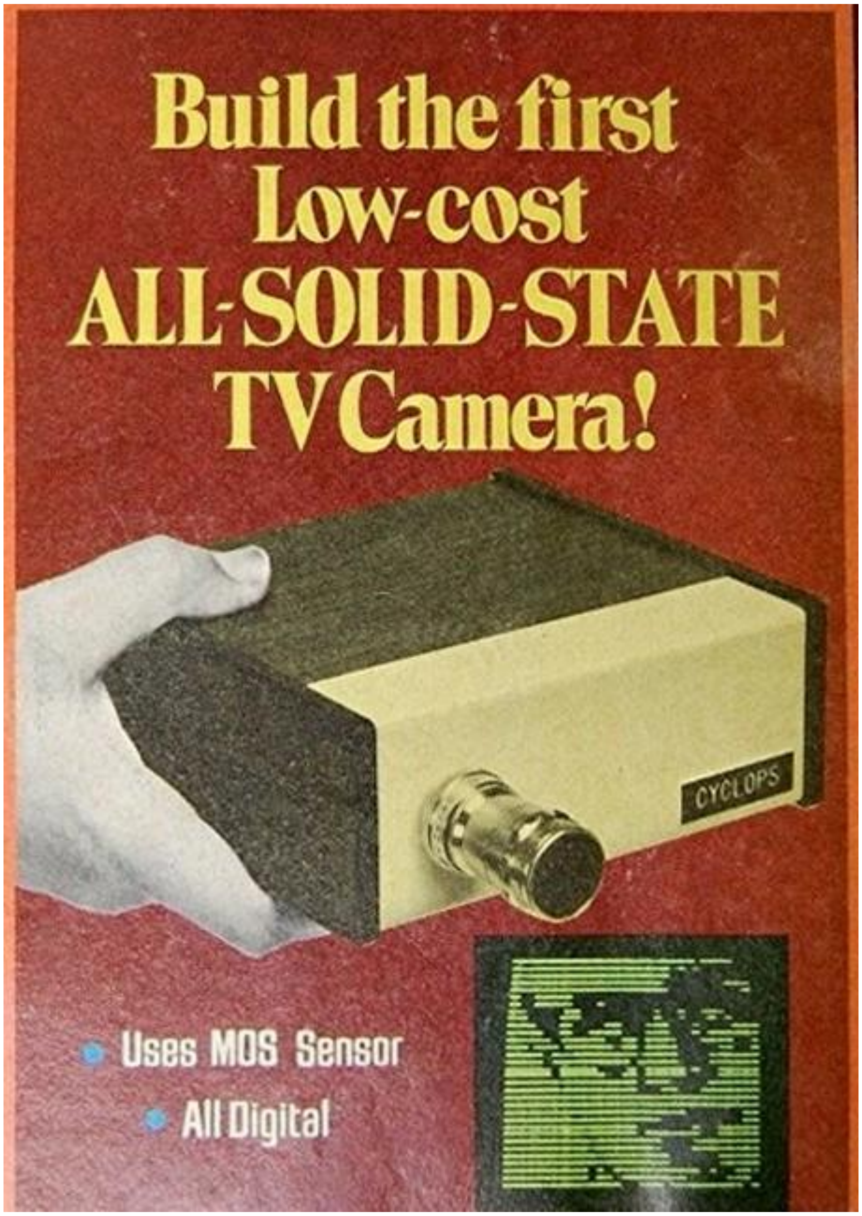
The whole system with the Cyclops's two S-100 camera controller boards and the two S-100 Dazzler boards, together sported a total of 130 TTL IC's, exclusive of the 5 IC's in the Camera head and a handful of transistors and voltage regulators. That is a very impressive IC count. And of course that does not count the myriad of IC's in the host computer.

There is a reason why the camera controller required so many IC's. This was because of the hardware processing required to create 16 shades of grey from the output of the MOS image sensor IC, which is in essence, is a 1 bit logic signal. The camera image is required to be loaded into a separate area of the computer's RAM, then a number of image fields processed by the firmware driver controlling the camera controller boards. When processed a frame of data is transferred into the Dazzler's section of RAM. The Dazzler reads this section of RAM and provides the Video Output signal, to make the image visible on the VDU.

The initial version of the Cyclops camera was published in Popular Electronics where the Camera control circuit generated an X-Y image from an Oscilloscope instead. Of note, this "Oscilloscope" display version of the Cyclops camera controller is not covered in this article, but it is interesting to look at.

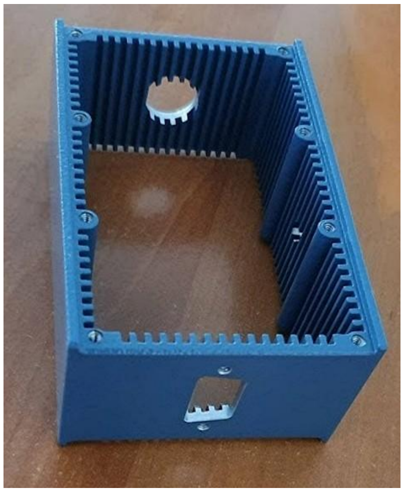
The green picture in the lower left of the front cover below was said by Terry Walker to have been taken taken with the Cyclops camera, of the magazine's Editor at the time, Mr. Les Solomon.

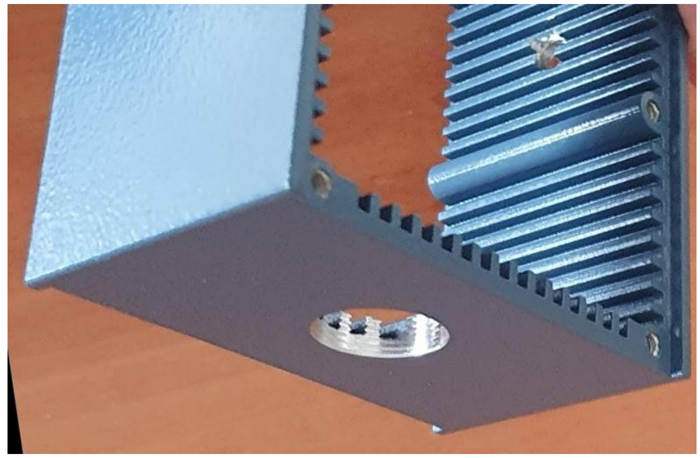
From the Cover, Popular Electronics Magazine, February 1975:




**Building the Cyclops Camera Replica and the two 88-CCC Controller Boards:**

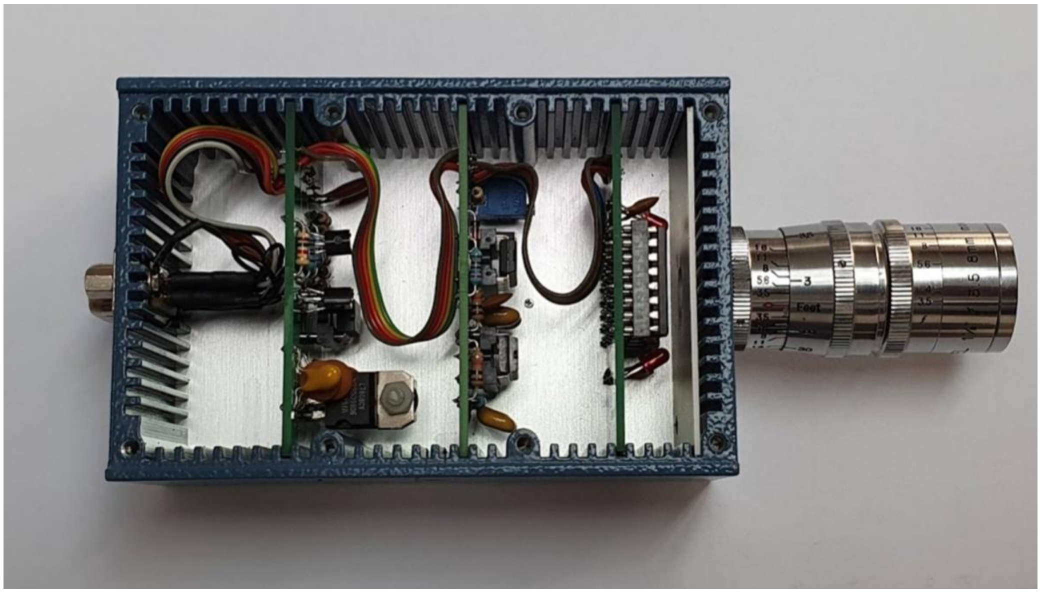
The exact Camera housing which Cromemco used was still available as a new part. This required some machining to cut the thread for the D mount lens, requiring a 5/8" 32TPI UNEF TAP that I bought for the task, and the hole for the D9 connector on the rear took some time to fashion exactly. There is also a ¼ inch 20 TPI threaded hole (which is a standard camera mount hole for a tripod)



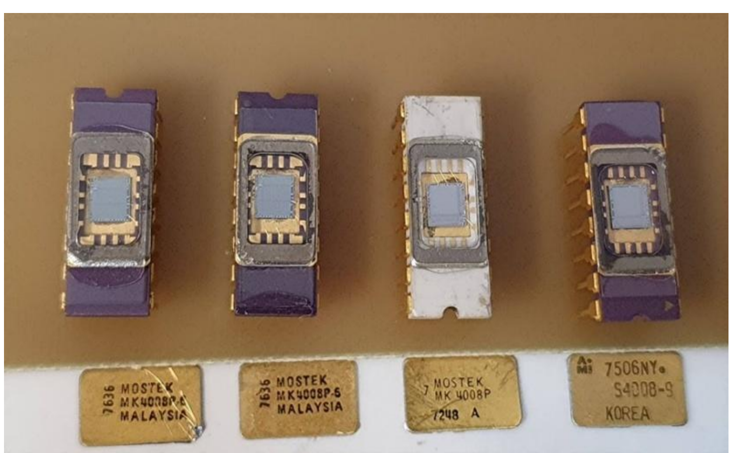


Also it was not difficult to find a D mount 16mm camera lens, there were a number of options on ebay.


The photo below shows the internal construction of the camera. The white plastic plate between the Lens and front board containing the Sensor IC, helps reflect the LED bias lights evenly onto the sensor IC. The Bias lights lift the baseline sensitivity of the image sensor.



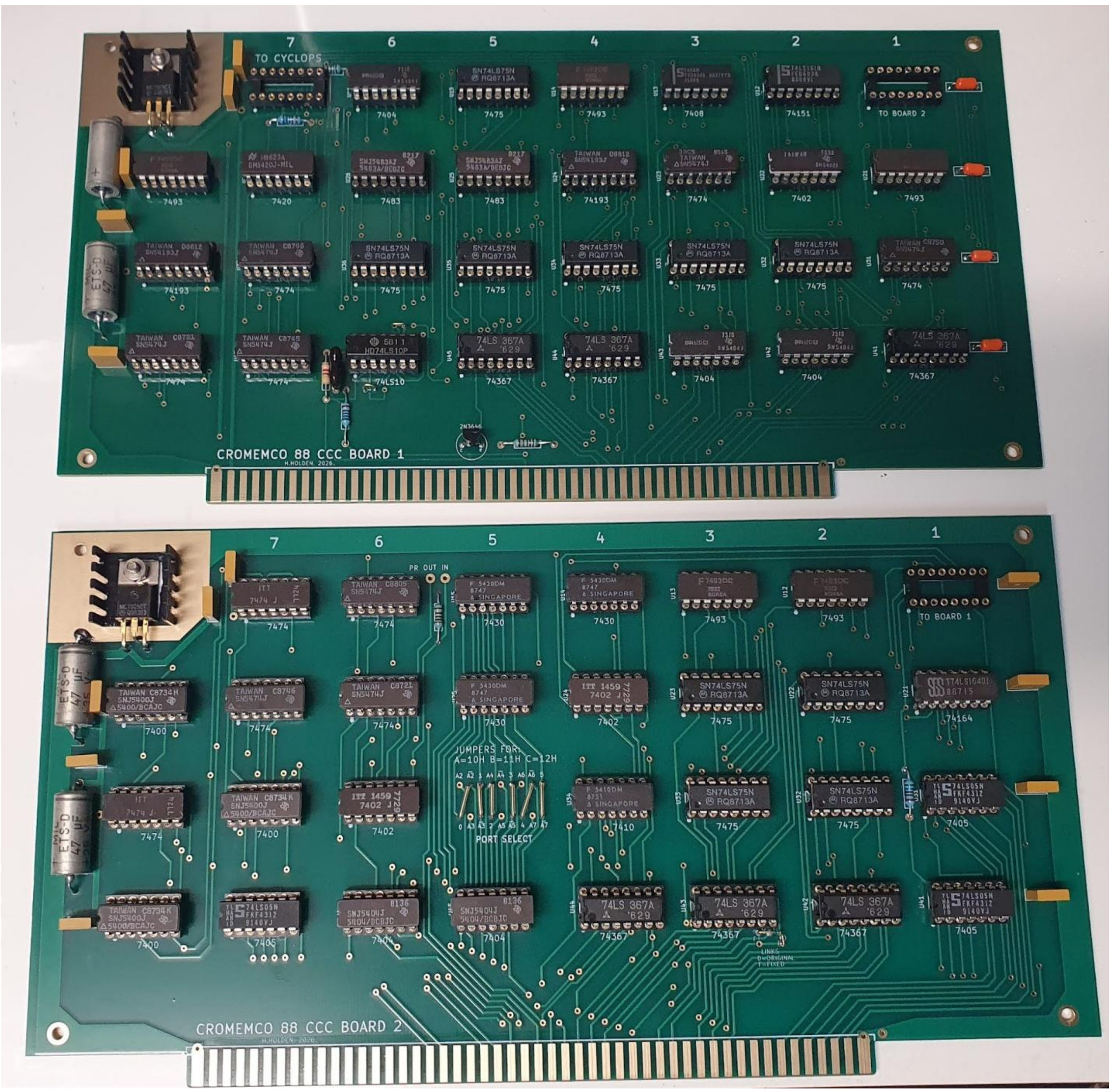
Another task in building the camera head itself involved removing the tops off some vintage Mostek Memory IC’s, and fitting a glass cover. After I had removed 4 or so tops from this sort of IC my skill at doing it improved. After the tops are removed, to protect the die it is best to place a glass cover. For this I initially used some 18mm x 18mm microscope glass cover slides which are about 0.14mm thick, and cut them in half. That was not an easy task. Later I found some excellent pre-made 17mm x 7mm x 0.5mm quartz glass windows and moved to those. The photo below shows some de-roofed memory IC’s:



The original part used by the designer Terry Walker was the S-4006. Therefore I acquired some of those for de-capping and testing too.

The three small camera PCB's covered by Cromemco's 88-ACC document had been recreated in the past. The two main S-100 boards for the computer (shown below) covered by Cromemco's 88-CCC document had been replicated in the past and were available on Github. Unfortunately they had some issues and did not work. They had never been tested. They were created from the schematic capture, not the actual original Cromemco PCB's. There were drawing mistakes in Cromemco's documented schematics. The worst errors were on board 1 and it is a complex logic circuit. I had to recreate the boards from scratch.

My hand created replica boards based on Cromemco's board foil patterns:



I found that the only way to recover the correct schematics was to reverse engineer them from the two original board's foil patterns. However these boards are quite rare now. I had to work with the available historical online photos. Fortunately there was enough information to be able to regenerate the schematics from those and compare those to Cromemco's published schematics and apply the corrections. This made it possible to regenerate a pair of working PCB's shown above. These were hand drawn into KiCad.

In the process of doing this I found a PCB error I was not expecting. I discovered there was a track layout error on Cromemco's original board 2 pcb. It resulted in the two LSB's of the pixel intensity data being accidentally swapped around. The effect of this was not noticed at the time, because a small disordering of the lower level grey scale was not easily seen in the image due to the coarse nature of it.

In addition, the firmware program, provided by Cromemco, to run the Cyclops in the computer was defective. I found after studying it and writing out the sequential addresses, that there was an incorrect jump to an immediate data location rather than an Opcode. This problem was corrected by the computer scientist Mr. David Roberts in the UK (a member of the Vintage Computer Forum) who helped with this problem. He worked out where the correct location of the jump should have been. It was basically a typo in Cromemco's Octal code where the jump was specified to the octal address 211, where it should have been address 221. David also found a faulty bit mask value. He corrected that too and was also able to comment the entire program, so as to help others to understand how it works.

**Running the Dazzler & the Camera Controller cards in the SOL-20 computer:**

This computer has a limited number of S-100 card Slots, only 5. There is a $6^{th}$ slot pointed vertically which is inconvenient. For the system to run requires the two Dazzler boards, the two 88-CCC camera controller boards and the required static memory board. This uses up the 5 slots, so that I could not run any ROM boards or Disk controller cards which hold programs.

Instead I formatted the driver software as a .ENT file, to send to the SOL-20 via Tera-Term and its serial port. The .ENT file acts as though the bytes are being manually typed in by an operator into memory. To put the SOL into a receiving mode it is just a matter of typing SET I=1.

**How good is a 32 x 32, 16 shades of grey pixel image?**

Prior to running and testing the full capabilities of the Cyclops camera itself, what I was interested in was:  What would a monochrome image at best, with 16 shades of grey, and 32 x

32 pixels would look like on a VDU. This is a Dazzler picture format option I had not produced an image for in the past. In this format the image is encoded as 512 bytes. The upper and lower adjacent nibbles each control the 16 shades of grey (counting 0 as black and F as peak white) of adjacent pixels. The first pixel starting out at the top left of the display area and the last pixel at the bottom right.

Therefore I started with a high resolution monochrome .BMP image and compressed it to 32x32 pixels in a photo editor. In a Hex Editor program I stripped off the 54 byte leader. Then it required reversal of the byte order. This is because a .BMP image starts at the bottom right and ends at the top left. Then it required removal of every second and third byte. This is because in a monochrome .BMP file, the RGB levels are defined by matching byte triplets.

After that the byte values were examined for magnitude, assigning bytes with a decimal value 0 to 15 a value of 00. And a value of 16 to 31 a value of 01 and so on and so on until complete.

The file then consisting 1024 bytes in the range 00 to 0F corresponding to 16 shades of grey. The finally the byte pairs were combined, for example if the first byte in a pair was 0A and the second byte in the pair 06, a new byte with a value of A6 replaced that byte pair.

Now the file was in a suitable form to display on the Dazzler's 32 x 32 image mode. The result is shown below, again note: ***This is NOT an image produced by the Cyclops Camera***, but a "test image" to see what this format image would look like, if some camera system had excellent performance in producing an image in this 32 x 32, 16 grey scale format:

**Operating Theory of the Cyclops Camera:**

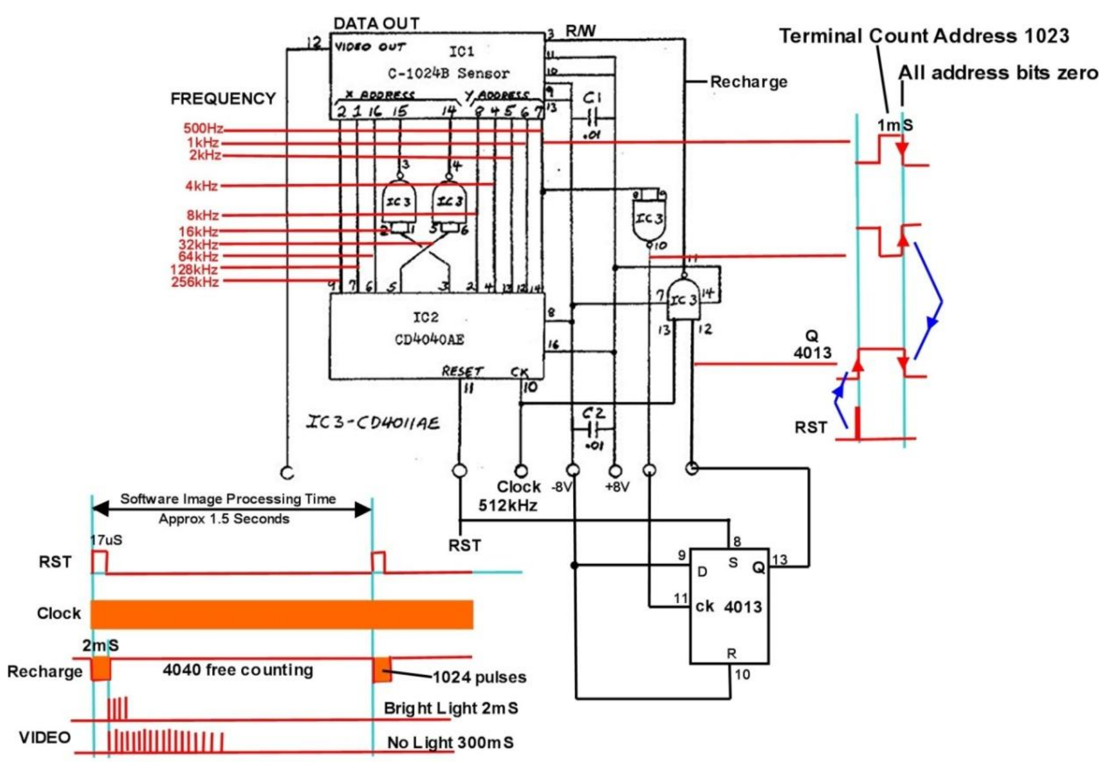
How is this camera able to create 16 shades of grey, from a DRAM IC acting as an image sensor, which only has a binary or on-off output for each of its memory cells?

The S- 4006 DRAM IC (and others such as the MK4008) stores its data a charge on a capacitor which is essentially a p Channel CMOS gate. The entire array of memory "cells" is addressed with a write cycle which pre-charges all of the capacitors in the array setting the bit value to logic 1 for each cell.

This means every cell in the 1024 cell array has a logic value of 1 immediately after the pre-charge process. That corresponds to a situation where the pixels in this case, corresponding to the high logic at each address in the array are interpreted by the electronics in the camera controller as having a value of Black or logic zero. This is because the output of the camera sensor is inverted by a transistor stage to send down the camera cable to the 88-CCC control boards.

In the normal case, using the chip as the memory device it was designed for and not a sensor, the situation is such that reads of the data are not destructive, however writes or refresh cycles must be performed every 2mS to recharge the cell's capacitors, otherwise they self discharge and the data is lost and the value in the cell is logic 0.

During this self discharge period, if light falls on the array of cells, it accelerates the discharge of the capacitive elements. Light drives the state of charge to zero at a higher rate than the self discharge. Dynamically though, depending on the exact light level falling on a cell from a focussed image on it, some cells are discharging faster than others.



The circuit above modified from the Cromemco 88-ACC document shows the sensor & address generator arrangement. The clock signal is derived for the host computer's approximate 2MHz signal by dividing it down with two flip flops on the 88-Controller boards. In my SOL-20 computer this divided clock frequency came out close to 512kHz.

A reset pulse (RST) is sent from the controller board initiates the data acquisition. At that time a full address count is initiated so that each memory cell (corresponding to each of the 1024 pixels) is charged and each cell has a logic state of 1, a recharge cycle if you like.

In normal use, as a memory IC, the entire array of cells, whatever logic value they have, would be refreshed with either a write or a refresh cycle every 2mS or less, recommended by the manufacturer, to avoid the data fading away. In this case that does not happen and the system to create the 16 shades of grey relies on the data ultimately fading away in all cells to zero.

At the end of the generated 0 to 1023 addresses by the 4040 ripple counter, immediately following the RST, at the terminal count of 1023, the MSB is high as are all the address bits and and on the next clock pulse goes low, clocking IC4 and terminating the 2mS recharge time. This corresponds to 1024 total address locations being pulsed by the clock to a logic 1.

After that 2mS recharge time initiated by RST, the 4040 counter simply keeps counting and counting in the range of 0 to 1023, generating addresses and rolling over. The sensor IC is in a fixed read mode and the data which comes out of it will be either a 1 or a 0 from each cell. Initially, it is all 1's but since there is no refreshing going on, the charges are decaying away and ultimately all cells will go to zero.

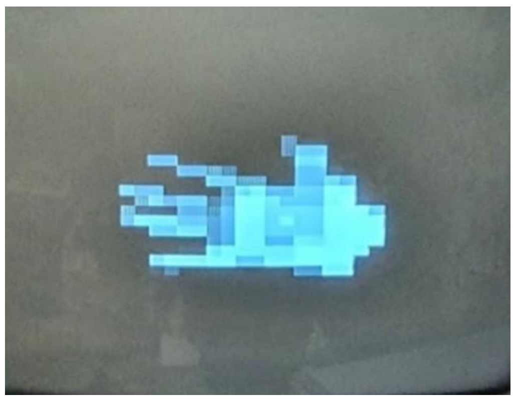
For example in the dark, with a typical image sensor IC it takes around 300mS to 400mS for all of the cells to discharge on their own. Shining a light on these cells accelerates the discharge with a photo diode effect. Applying a very bright light to the the chip's die shortens the discharge time for a cell to around 2mS. At intermediate light levels, it takes varying amounts of time in between to get the cell to change its logic state. This is how grey levels are created from from a time domain threshold effect to transition the logic state.

As noted above, the actual data sent to the camera controller board is an inverted version of what comes out of the sensor IC, so at the controller end, a logic low represents a charged pixel and black level. A logic 1 represents white. The software controlling the camera control circuitry, over the time that the charges in the array are fading away in a light sensitive and time variable manner, examines the logic value at each of the 1024 addresses.

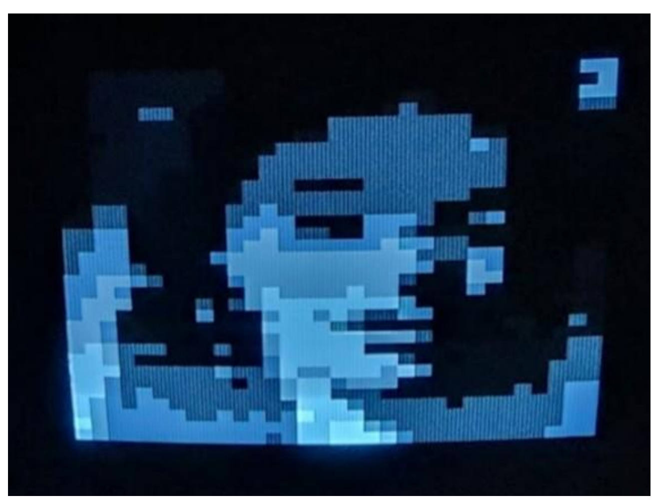
The entire array of 1024 cells (address locations of those cells and their binary value) is scanned up to 15 times or 15 fields are used to create the one image frame. The demo program created by Cromemco uses 12 fields to make the displayed frame. As noted, the Dazzler's 32 x32 image format is arranged so that one byte controls two adjacent pixels. Each nibble of the byte has a value 0 to F which encoded for black and the 15 shades above that to white. The Dazzlers image memory therefore uses up 512 bytes of the host computer's RAM to store the data values for 1024 pixels.

Testing of the sensor IC's showed that there was quite a lot of inter-cell variability in some types and variability between types. To select the better IC specimens for the application, I created a diffuse light source to perfectly illuminate all of the cells equally, and picked the IC with the lowest inter-cell variability.

Sample images produced by the camera are shown below. The one of my hand was taken with the hand about 6 feet from the camera with a 60W tungsten light source. The Sensor IC is more sensitive in the infra red area and does better with a Tungsten lamp than other types.



The two images below: One on the left is a source image displayed on a computer VDU with the camera pointing at the VDU and the one next to it is the image on the right that the Cyclops Camera system created from "looking at it"



One feature noticed here, for all the image sensor IC's I tried, there is a tendency for the image to brighten up toward the end of the frame, toward the bottom of the image. This is likely due to the data processing time being a little longer as the frame is constructed from top to bottom and that resulting in a little more discharge of the memory cells as the frame assembly proceeds. When using the camera to view a VDU image, the VDU's brightness has to be turned up to full because the image sensor is not very light sensitive.

Attempting to use the Camera to view printed images on paper was interesting. With a black and white image created on a piece of A4 paper with an Inkjet printer and illuminating that with a 60W Tungsten lamp, the camera "sees" significant light energy reflected from the areas which look jet Black to the Human Eye. Therefore the contrast is very poor. Possibly that would improve with a Laser printer image. In the case of the Camera looking at a black and white image on a VDU screen instead, there is no light emanating from the black areas, so there is no response from the camera and the image contrast is much better.

Again the reproduced image is stretched somewhat from side to side because the pixel elements and sensor array is square, but the 4 x 3 aspect of the VDU stretches them on the horizontal axis.

**Summary:**

The Cyclops camera is one of the World's first digital video cameras, one that a home enthusiast could make for themselves and explore the idea of creating images digitally, with a Silicon Sensor IC, rather than an image tube such as a Vidicon tube.

And while its 32 x 32 pixel pixel image is low resolution and only 16 grey levels, the project was more or less a proof of concept design; if the notion that a solid state sensor could work.

It was a very creative idea to use an existing DRAM memory chip to do it, rather than having an expensive precision higher resolution sensor made. On top of this, to be able to create a grey scale from a binary output of the sensor IC, using the temporal behaviour of the DRAM memory cells, was creative, both in the hardware and system software. It is difficult there to know which is the cleverer of the two, because they are equally impressive.

Of course as the years rolled by nearly all camera sensors became solid state and now people carry around mobile phone cameras capable of astonishing color super high resolution electronic images. These are so familiar now that most likely they are taken for granted. However, looking back in history, it is always very interesting to see how it all got started.

*****************************************************************************