THE MICROCOMPUTER

The WF/PC uses an 8-bit, RCA 1801 microprocessor, which is rather slow compared to today's microprocessing speeds. Before I got involved with space instruments, I always asked myself what would people do if some part of the memory fails, or what if they decided to change the program while the space craft is way up there in space?

As I will explain shortly, the way WF/PC does this is by setting break points at different parts of the program. The way a microprocessor works is like someone following your instructions, written on pages of a book called memory. In the WF/PC processor as well as your IBM-XThome computer, these instructions are written as an eight digit binary number (or one byte) such as 1100,0000, or 0111,0001, etc. These numbers can be easily represented in the form "CO", or "71", known as a hexadecimal representation.

The book that memorizes all this even when the computer is off, is called ROM or read only memory. The 6K ROM in WF/PC is like a book with 6144 pages. There is also a 4k RAM or random access memory, which is like a diary book with 4096 pages (only half of the available RAM is normally used). The diary book can be erased by either writing over it or by shutting off the computer.

The so called computer brain refers to the guy reading the cards (Fig. 10), and doing exactly what they say. For example, "CO" means to jump to some other page in the book, and do what it says in that page, "F8" means to read from some location of memory. When the computer is turned on, it starts reading location 0000 (Fig. 10, under ROM), and executes the program one by one. To do this, he uses a chalk board (REGISTERS), a hand calculator (Arithmetic and Logic Unit), and a place on the board where the results from ALU are displayed (D register).

In home computers, programs are loaded into RAM, and changed if necessary. But in spacecraft applications, the programs are usually executed from the ROM book, and no new page can be added to the book, or taken out. A way out of this is to divide the book into chapters (Fig. 10), and at the end of each chapter (break point), tell the processor to jump to a page in the RAM book (the first break point is at ROM location 010B). This page of RAM (for the first break point, this would be RAM location 4100) would simply tell the processor to go back to the beginning of next chapter in ROM, and continue with the program until the next break point (ROM location 012D, etc.).

The advantage of this is that if you decide to change the program within any of the chapters, you can change the page in RAM to tell the processor to skip any chapter, and continue with a new program loaded in RAM, For example, if you decide to do some changes in the program that resides within ROM locations 012E and O 17A, a substitute program can be transmitted to the spacecraft, and written to any RAM location between 4800, and 4FFF (spare RAM).

When the processor reaches the second break point, it jumps to RAM location 4103, and then jumps to the beginning of the new program in RAM, and continues the rest of the program in ROM (pages 4104 and 4105 have a new jump address). So instead of having to transmit an entire new program, you have the convenience of changing only a part of it. Of course, if there is a short power failure in the space craft, the RAM contents are destroyed, and the computer starts running the same old program stored in ROM, and you have to upgrade the RAM once again.

In the real world, a single memory chip is not big enough to contain all the program. Each pair of WF/PC's ROM chips (books) contains 256 pages, making an array of 48 chips. If any of the memory chips would fail, break points are useful to go around bad ROM locations. In case of primary RAM failure, the microcomputer uses the spare RAM.

THE CHARGE COUPLED DEVICE (CCD)

Next time you have dinner, pick up three forks, and orient them at 90 degree angles on the table. Now slide the two forks that are facing each other so that their teeth interlace together, then place the third one on top of the next two (Fig. 11), Imagine connecting any of the two perpendicular forks to a negative potential (let's say -8 volts), and the third one to a positive (or less negative) potential (let's say +2 volts). You have just constructed a model for a two phase CCD device with 12 pixels!

The way a CCD works, is by trapping tiny elements of an image in a matrix array of storing elements called pixels. In simple electronic terms, projecting an image on a CCD is like pouring a hand full of electrons on the square area interlaced by the three forks: The electrons will be collected only on nine different locations, with the +2V potential.

The idea is that each pixel encloses electrons in both vertical and horizontal directions by the two perpendicular forks connected to -8 volts. The electrons, having a negative charge, will only be attracted to the teeth of the fork connected to +2 volts.

The number of electrons in each pixel would describe the brightness in that area of the image: The more the number of electrons, the brighter the image, and vice versa. CCD's then, behave similar to photographic film, they require a shutter, and should get the right amount of exposure. If over exposed, the electron capacity of the pixels are exceeded, and the electrons would overflow to the neighboring pixels. But the light itself consists of photons, how is a "photon image" converted to an "electron image"?

The way this works is like how photo transistors convert light into an electronic signal. Inside a photo transistor is a layer of silicon, and when a photon enters this layer, it generates one or more electrons, provided that it is powerful enough to release this electron from the outer layer of Silicon atoms. The energy of this photon is determined by its wave length. In the ultra-violet region, photons have a much higher energy, and

interact immediately after entering the Silicon layer. In the infrared region, photons have less energy, and penetrate deeper into the Silicon to release an electron.

Figure 13 shows the schematic of a WF/PC-II CCD, with the number of pixels reduced to an array of 4 (vertical) by 9 (horizontal). The actual CCD contains an 800 x 800 pixel array, produced by four interlacing structures that resemble forks with 800 teeth! These CCD's come in a 40-pin package. You can think of pins 10, 11, and 12 as the handles of the forks sticking out of the chip (this is a 3-phase CCD device)!

The teeth of these three forks run horizontally across the active area of the device, and are labeled 1, 2, and 3. For example, if you trace the wire from pin 10 under a microscope, you '11 reach a vertical bar, which is connected to every other three horizontal running traces (labeled "2" in Fig. 13). All the horizontal traces 1, 2, and 3 are separated by running over a series of vertical barriers, labeled "B". In CCD terminology, forks are called phases (i.e. PI, P2, and P3), and the charge Barriers (labeled "B'') are equivalent to the vertical fork in Fig. 11.

In reality, these fork structures are thin implants on the surface of a substrate, which is a stack of several Silicon layers. When the light strikes the Silicon layer, it releases electrons, which are then collected in the pixels. During the exposure, phases 1 and 3 are held low by applying -9 volts to pins 12 and 13, and Phase 2 is held high by applying +3.5 volts to pin 14. This makes phase 2 to be the electron collecting phase of the CCD, creating a total of 36 "charge buckets" (640,000 pixels in the real device).

Let's say an star image falls on any given area of the CCD, and forms an image the size of a pixel. The light photons would penetrate into the silicon substrate, and release a number of electrons depending on the brightness of the star (and its wave length). The generated electrons would be attracted to the closest phase with the highest potential, or the closest phase 2. Each pixel in WF/PC CCD's has an area of 15X 15 microns (1 mic. = 1 millionth of a meter), and can collect up to 50,000 electrons. The exposure time must be chosen such that the brightest area of interest would not generate over 50,000 electrons.

READING OUT THE CCD IMAGE

Figure 12 shows the basic read out scheme ofa CCD, using parallel and serial shifting of the pixels. A single parallel shift, dumps every row of the CCD to its lower row, and dumps the very bottom row to a horizontal shift register, The horizontal array is then" read out one by one, and the number of electrons in each pixel is measured (six horizontal shifts for every one vertical shift).

To understand the shifting operation, consider switching the voltages between phases 2 and 3 in figure 13. For the same reason the charges were collected under phase 1, they will now migrate to phase 3. Now switch the voltages between phases 3 and 1: The charges would be collected under phase 1. Finally, by switching the voltages between 1 and 2, the charges end up on a new phase 2, at a higher row.

To visualize this, consider a series of mechanical pistons that go up and down by a common crank shaft (Fig. 14), which moves each two neighboring pistons 120 degrees out of phase. Similar to the CCD, phase 1 is lower than the rest of the pistons, so all the electrons end up in phase 1. Switching the voltages between phases 1 and 2, is like turning the crank shaft 120 degrees (Fig. 14).

The barrier columns separate the vertical pixels in the horizontal direction (Fig. 13, 16). With a complete rotation of the crank shaft, the electrons from the bottom pixels are fed to a serial register (the electrons from the parallel registers can either go to phase 2 or 3 of the serial register). Before the serial register can start shifting out the electrons, it has to be first isolated from the parallel (or vertical) registers. This is the job of the transfer register (Fig. 13, 16, 17).

The serial shifting operation is the same as the parallel register, but it is made longer to provide more information about the exposure, I will explain this later in more detail. The job of the output well is similar to the role of the transfer phase (Fig. 17). The contents of each pixel are dumped into a measuring system, which measures the voltage generated by the number of electrons. The PC (pre charge) is synchronized with the serial shifting register, so that it dumps out the previous content of the measuring device, while the serial register is shifting the contents of the next pixel (Fig. 17). The output of the measuring device is the video signal (a single voltage value between .003 to 12.228 millivolts).

If you now go back to the CCD schematic (Fig, 13), you know the job of every pin in the device. There are in fact two ways to read this CCD, from the top, or from the button. This is similar to rotating the crank shaft in the opposite direction. The pin names followed by letter "L''correspond to control pins for the Lower register, and those followed by "U" correspond to the lower register.

CONVERTING THE NUMBER OF ELECTRONS TO A DIGITALNUMBER

Considering the possible number of electrons in a given pixel, we may have any number of electrons ranging from zero to 50,000 (Fig. 15). In order to transmit this information to a ground station through a radio antenna, it first has to be converted to a digital number. The WF/PC analog to digital converter does this conversion with 12 bits of resolution. This translates to 7.5 electrons per digital number, or 4094 different shades of gray.

In simpler terms, ifa pixel is empty, the A/D converter would assign a number to it which would read 0000,0000,0000 (Fig, 15). If the pixel is full, the A/D converter would assign the value 1111,1111,1111 to it, and there are 4096 different possible combinations in between. Dividing 50,000 by 4095, you'll get 7.5 electrons. Increasing the resolution of the A/D converter from 12 bits to 16 bits means going from a 35mm camera to a Hasselblad!

SENDING DOWN THE IMAGES

The camera heads are read out one at a time, and it takes about 20 seconds to read one CCD. Since the instrument has only 2K memory, it transmits every pixel immediately after the A/D conversion (except the 17th pixel in every line). Because the logic board sends the data out in a 16 bit format (Fig. 18), the 12 bits out of the A/D converter are copied twice in the middle of the 16 bit slot (bits 4,5,6, and 7 are the same as 8,9,10, and 11).

In addition to the video signal, there are many engineering information, such as which camera took the picture, the temperature of the CCD during the time of exposure, the shutter speed, what filters where used, and hundreds of other information (Fig. 18).

The result is a block of data 800 by 813, and 16 bits deep. This is only the WF/PC data. There is also the Standard Header Packet (SHP), which contains the time, date, user scientist's name, pointing and orientation of the telescope, and the status of the rest of its many instruments, and so on. The final raw data stored for a single image (four CCDS) is over six Megabytes. Three of Hubble images would be enough data to fillthe hard disc capacity inside an ordinary PC!

The telescope images are relayed through the TDRS (Tracking and Data Relay Satellite) positioned in a geosynchronous orbit, to White Sands radio telescopes, and then to the Goddard Space Flight Center via another communication satellite (Fig. 19). The image is finally sent to the STSI (Space Telescope Science Institute) at maryland. All this causes less than a 3 second delay,

I like to express my thanks to many people at JPL, including Lloyd Adams, for answering many of my questions on the opto-mechanical issues concerning the WF/PC design. I also wish to thank Tom Bickler, who patiently helped me with the electronics, and the article, The CCD part would not have been possible without the help of Patrick Sullivan, and Arsham Dingizian. The software portion was clearly explained to me by Bobbie Woo, and Neil Pignatano. I also owe many thanks to Peter Kobzeff, Charlie Davis, Ed Kirchhof, Bill Hornaday, Mike Sucy, Gail Watson, Margaret Dietrich, Maryjon Reedall, Donna Beckert, Dave Thiessen, John Bousman, Ramona Ynigez, Marry Reeves, Juanita Reyes, Percy Moore, Olga Paredes, and many others.

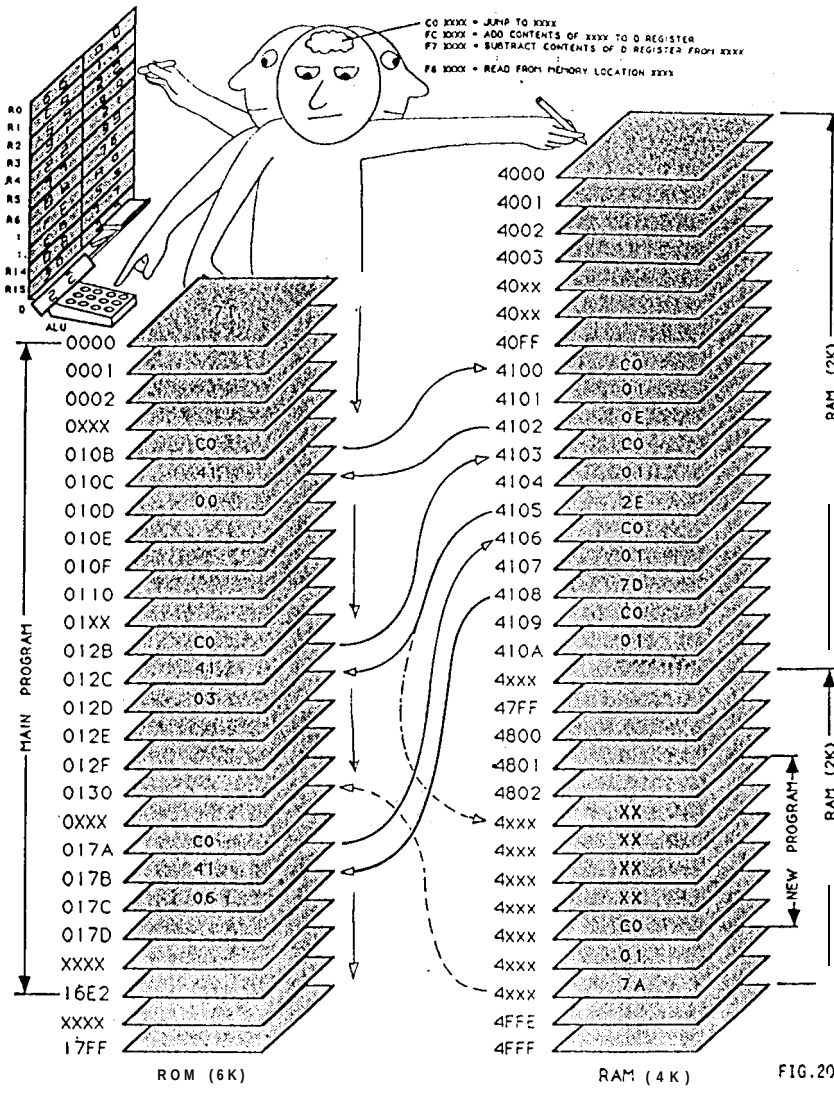

FIG.20

F19.1

THE MICRO PROCESSOR

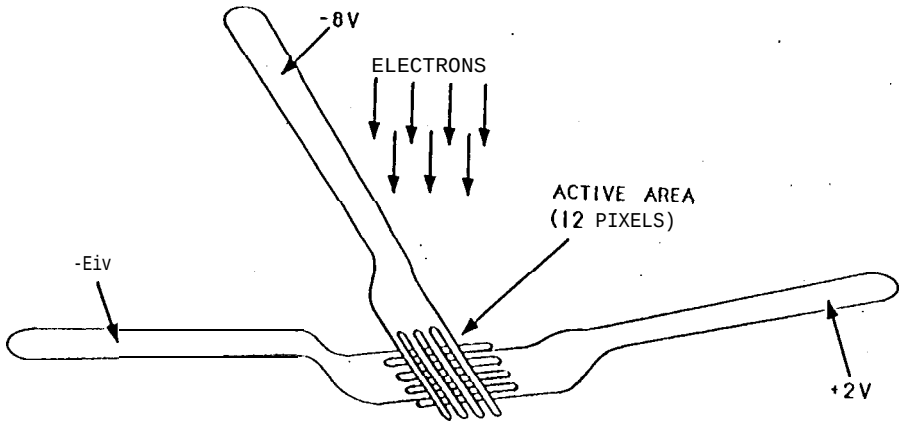

FIG.21

Fig. 11

The charge coupled Device (CCD)

FIG. 13

FIG.23 CCD IMAGE READOUT

FIG. 12

FIG. 24

FIG. 14

FIG. 27

FIG. 15

FIG. 17

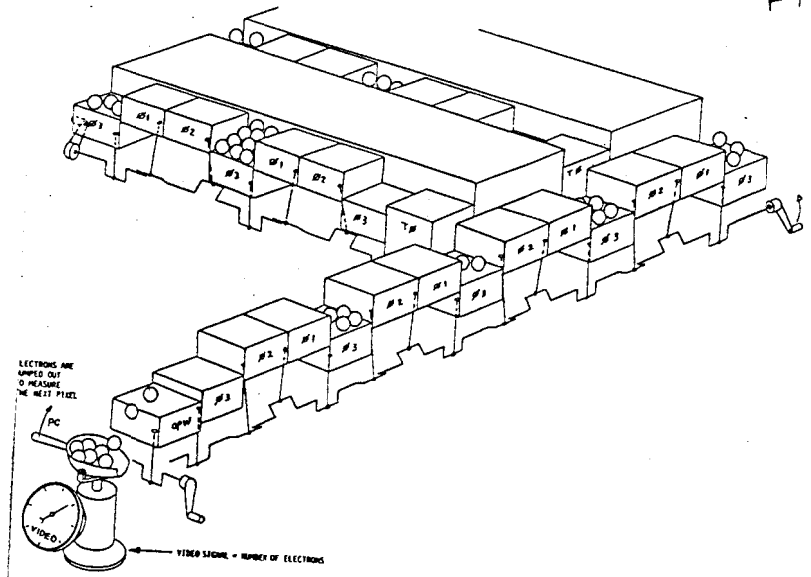

FIG. 26

FIG. 8

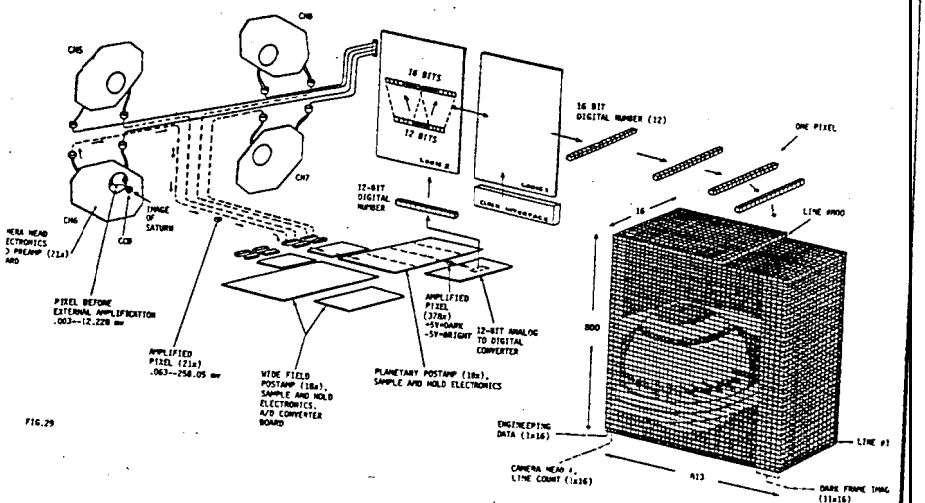

FIG. 29

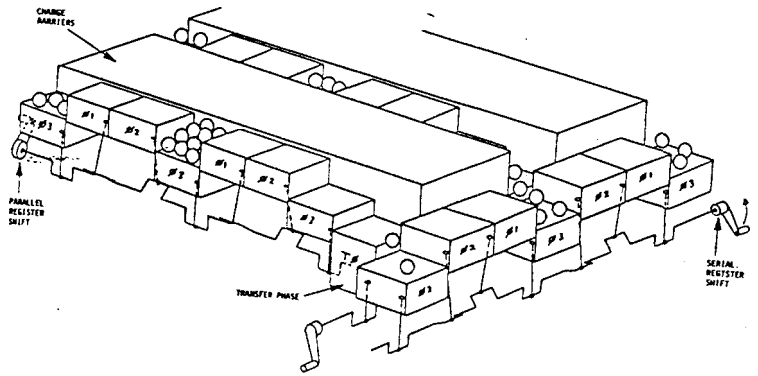

FIG. 25

FIG. 16

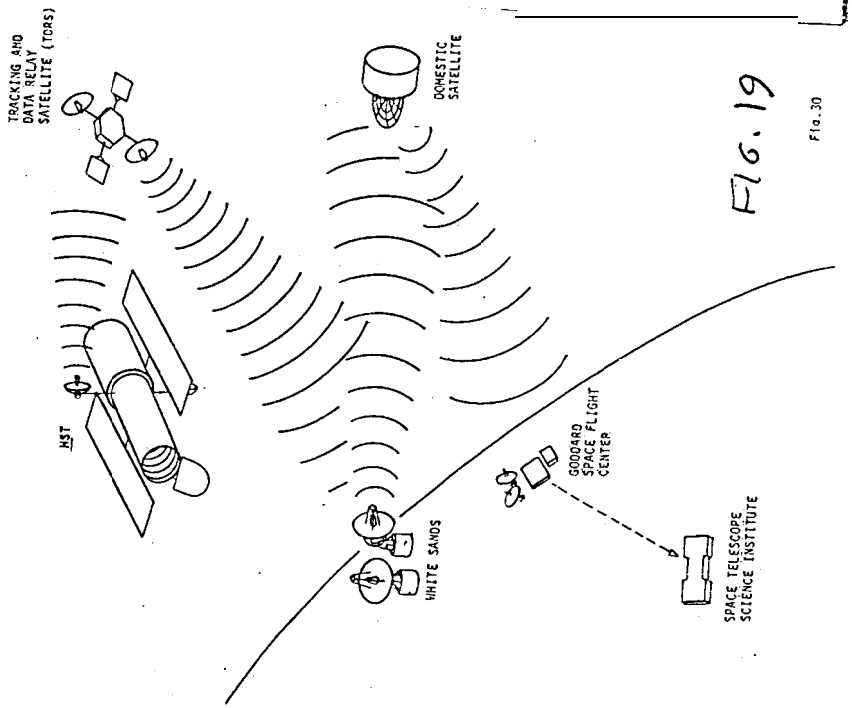

FIG. 19

FIG. 30