DIGITAL PROCESSING OF RADAR AND SATELLITE IMAGES

By

© KENNETH WALTER SMITH , B.ENG.

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Engineering

McMaster University
May 1980

DIGITAL PROCESSING OF RADAR AND SATELLITE IMAGES

MASTER OF ENGINEERING (1980)
(Electrical)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: Digital Processing of Radar and Satellite Images

AUTHOR: Kenneth Walter Smith, B.Eng. (McMaster University)

SUPERVISOR: Dr. C. R. Carter

NUMBER OF PAGES: x , 267

ABSTRACT

A microcomputer based image processing and display system was constructed, consisting of an Intel 8080 microcomputer development system, a 128 kilobyte memory, a colour television monitor, and an image display controller and interface. This display system was used to compare images from airport surveillance radar and meteorological satellites. The main purpose of the comparison was to detect and classify hazards to aircraft, particularly weather and migratory birds. The image display system is a relatively inexpensive means of performing basic processing of images for a variety of applications.

Data from the Winnipeg airport radar and from the NOAA 5 meteorological satellite, of the same area, was simultaneously obtained during May 1978, the time of the bird migration. The two images were geometrically transformed into the same map projection and then compared. It was found that some clouds were common to both images, but many were not. Birds, on the other hand, are only detected by the radar. More data needs to be collected and analyzed, under different weather conditions and the radar optimized for detecting clouds.

ACKNOWLEDGEMENTS

The author would like to express his thanks and appeciation for the assistance from the following people:

The National Research Council of Canada

Mr. John Lewis and Mr. Charles Taggart of Atmospheric Environment Services

Mr. Glen Leinweber, Department of Physics, McMaster University

Mrs. S. El-Sagir, Communications Research Laboratory, McMaster University

Staff and students of the Communications Research Laboratory, McMaster University

and Dr. C. R. Carter, supervisor

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF TABLES

APPENDICES

CHAPTER 1

1.1 PROBLEM

Airport surveillance radar is a major instrument for monitoring and controlling air traffic in the vicinity of airports. The most serious hazard to aircraft is one of collisions with other aircraft. The major task surveillance radar performs is to detect aircraft and plot their positions in real time. Navigational equipment, other than radar, has been developed for determining positions of aircraft for guiding the aircraft to its destination and avoiding collision with other aircraft, particularly important when the aircraft is out of range of radar.

Other hazards to aircraft are weather and migratory birds. These hazards can also be detected by radar, although they are often not detected. When they are detected, the hazard may not be recognized. Identifying these targets is often difficult since their statistical nature has a large degree of variability. The problem is further complicated by the fact that airport radar is optimized for detecting aircraft and suppressing other objects such as weather and birds.

The problem has been generally addressed with radar signal processing techniques. Another approach to the problem is through the methods of remote sensing. Remote sensing, of which radar is one technique, is the detection, classification, identification and measurement of objects generally through the use of electromagnetic radiation. Radar uses microwave radiation in the active mode for observation of the air-space around an airport. The success of many remote sensing projects and investigations (such as earth resources satellites) is largely due to the "multispectral" aspect. Imaging sensors operate on more than one spectral region, since observing several spectral bands contains more information than one. Due to the availability of meteorological satellite data, images were compared to corresponding radar images. It is hoped that they will supply information, not found in the radar signals, which will aid in identifying radar echoes.

1.2 CURRENT TECHNIQUES IN RADAR

Airport surveillance radar installations generally use conventional processing and display techniques. A plan position indicator (PPI) is a large, long-persistence CRT which plots the radar signals in a map form, where azimuth and range of the target can be determined. Since aircraft are the primary targets, the received signal is processed to enhance the aircraft signals. This is done largely through a moving target indicator (MTI) filter which passes signals with a Doppler shift, eliminating stationary targets such as ground terrain. The major problem with MTI is its blind spots, where a fast moving target in a tangental direction to the radar beam has no radial velocity and hence is eliminated by the MTI filter. There is a growing interest in weather and bird clutter, both hazards to aircraft. These types of targets are difficult to detect and identify in comparison to aircraft since they are slower, weaker and less defined targets than aircraft.

Weather and bird clutter are often indistinguishable from ground clutter and are often filtered out by the MTI. These hazards are more important at close range where unfortunately the ground clutter is the worst.

On low dynamic range displays, such as PPI displays, weaker and less reflective weather and bird targets often appear dim, despite clipping and logarithmic amplitude compression. The only advantage a distributed target, such as weather, has over point targets, such as aircraft, is that the received signal from a distributed target has a power which is proportional to $1/R^2$. Point targets conform to a $1/R^4$ relationship.

With increasing emphasis on these hazards, many efforts are directed towards radar signal processing in order to analyze the signals and determine if they contains such targets. Signals known to contain various classes of targets must be analyzed in order to determine the characteristics of each type of target that separate and identify it from the rest. Haykin and Carter [3] summarize known characteristics of bird and weather clutter. One important amplitude signature of birds is the "wingbeat" modulation. Bird echoes often are non-uniform spatially, having a granular pattern. Spectral analysis of the baseband signals, using the Maximum Entropy Method, may offer success. Radar signal processing is difficult due to the high sampling rate required and the fact that the characteristics of the targets to be detected are not well defined.

1.3 CAPABILITIES OF SATELLITE REMOTE SENSING

Earth satellites probably have had the greatest impact on the science of remote sensing, particularly since the Earth Resources

Technology Satellite (ERTS-I) was launched in 1972. Orbiting imaging satellites have proven to have considerable advantages over aircraft. Satellites offer a continuously operating system which can cover the entire earth in a little as half a day, where the satellite can scan the earth in sections up to 3200 km. wide. Imaging satellites have a wide range of ground resolution available, from a resolution of 10 km. in global meteorological satellites to a few meters in military surveillance satellites. The main disadvantage of satellites is their cost.

The science of remote sensing makes the use of electromagnetic radiation emitted from objects of interest in order to detect them and gather information about them. In the passive mode, radiation is emitted or reflected where the original source of energy is the sun. In the active mode, the source of energy is the satellite, such as side-looking radar satellites. A problem with remote sensing system is the great amount of data available. Although a scene may have an almost infinite number of resolution cells, the resolution is limited to suit the requirements of the applicaation and to allow a reasonable area of coverage. In general, higher resolution scenes have a smaller region of coverage. Hence the principle of the "close-up" is employed. In addition, each resolution cell has a characteristic wavelength spectrum. It is the spectral characteristic which provides a great deal of information and forms most of the basis of remote sensing. The colour of an object only represents the visible spectrum. To fulfill the need for spectral analysis, satellites have multichannel imaging sensors, which produce images with different spectral bands. Again, the information has been limited to a practical amount. The sensors only sample small regions of a large spectrum or they can sample the spectrum with wide sensor windows. A typical earth resource satellite contains five sensors in different spectral regions, usually in the visible and infrared regions.

Of particular interest are the meteorological satellites. The use of these satellites is the most widespread of any remote sensing systems. Meteorological satellites have been in use since 1960 on an experimental basis and since 1965 on a daily operational basis. Continuous operational systems are important since weather systems are very dynamic in contrast to natural land and water resources which change with the seasons. ERTS-I, primarily concerned with natural resources, provided coverage of a given area on the earth's surface every 18 days. ERTS images are 185 km. square with a resolution of 100m. Meteorological satellites cover a much larger area, but with a lower resolution of the order of 1 km. These satellites are primarily concerned with weather systems. Since weather systems are very dynamic with mechanisms too complex for accurate analysis and prediction, high resolution data provides little useful information on a day-to-day basis.

Two spectral channels are of prime concern; the visible and infrared channels. The visible image simply shows the physical cloud structure. The infrared image allows temperature measurement of ground and cloud temperature. Infrared "light" allows night-time visibility since temperature is correlated with optical reflectivity. Clouds with a much higher brightness than land are also colder. If the infrared image represents temperature with grey levels, with white corresponding to cold and black corresponding to warm temperatures, the infrared image has a very close resemblance to the visual image.

Recent developments in remote sensing have expanded experimental and operational satellite observation of other aspects of meteorology [4]. These include vertical temperature profiles, atmospheric composition of carbon dioxide, ozone and water vapour, wind fields, precipitation estimation, and ice pack structure. With respect to radar, important aspects of remote sensing are precipitation detection and measurement, moisture and water vapour, vertical temperature profiles and atmospheric turbulence measurement.

1.4 RADAR AND SATELLITE IMAGERY COMPARISONS IN METEOROLOGY

There have been many investigations involving satellite imagery and radar. The purpose of most of these investigations was to attempt to estimate rainfall from satellite imagery, particularly in thunderstorms.

Woodley and Sancho [5] used 10 cm. meteorological radar to measure rainfall rate. A previous study determined the radar reflectivity-rainfall equation. The precipitation echoes were superimposed on ATS-3 satellite photographs of the southern Florida area. ATS-3 (Appplications Technology Satellite-3) is a geosynchronous satellite in orbit over eastern South Americawith a spin-scan imaging system, producing visual full-disc images of the earth (Western hemisphere). The processing of the images was largely optical. The radar display presented contoured echoes. The satellite photograph was brightness-contoured with a colour densitometer. It is interesting to note that these optical processing methods were considered superior to the laborious and expensive computer enhancement procedures, at the time this study was done (1970). Slides of the densitometer scope were then projected over a map and the projection was adjusted for best fit. Despite problems of calibration and brightness normalization, the results were encouraging on at least a qualitative basis. Radar cloud echoes corresponded to brighter cloud masses. Bright cloud masses produced echoes since the water vapour density is high, often producing precipitation. In general, small, bright cloud masses (young active thunderstorm systems) had the same area as the radar echoes from the same clouds. In the case of extensive cloud masses (decaying storms systems with cirrus cloud canopy), the echoes were much less correlated.

Martin and Suomi [6] performed a similar study. They also confirmed that bright cloud masses, with deep convection, corresponded to radar echoes, particularly large echoes. Small echoes corresponded with small, isolated clouds to such a high degree that they were used for radar and satellite image alignment. A major problem was distortion and scaling errors in the optical processing, and also ATS navigation error and radar azimuth error. It was also found that timing differences between the radar and satellite data collection greater than 5 minutes produced sizeable changes in the cloud structure, particularly active ones. Despite distortions, radar echoes and clouds were compared over limited sections of the image.

Blackmer [9] conducted a study using Defense Meteorological Satellites (DMSP) and weather radar. Again the purpose of the study was to show how visual cloud brightness and thermal IR radiance indicated the presence and intensity of precipitation. DMSP satellites provided a much higher resolution (.5 km.) than ATS. Hand analysis and computer processing was performed on the digitized radar and satellite data. A major problem was determining the geographical location of image points. The data was printed out as a matrix of intensity values.

From these, cloud contours are drawn. B-scan radar data was used since it was felt that scan conversions introduced distortion in the radar data. To minimize co-ordinate transformations, the satellite data was averaged over neighbouring points. This study showed a fairly good relationship between cloud radiance and brightness , where radiance is directly proportional to brightness. High radiance values correspond to low temperatures. Because of this relationship, IR images can well approximate visual images, particularly for night-time purposes. It must be recognized that extensive high cirrus clouds associated with a storm complex may obscure the underlying precipitating clouds. It was concluded that cloud brightness and radiance do provide information on precipitation distribution although there are exceptions to the rule. Precipitation intensity generally increases with cloud size up to a limit. In very large complexes, cloud brightness and radiance may be constant but the precipitation may be more localized. Another problem was that satellite data was only available in the morning, when storm systems were not developed. The study recommended that three-dimensional radar data and satellite microwave radiometers would give a much better indication of cloud water content.

Griffith [10] made a study to estimate rainfall from geosynchronous satellite imagery, again using ATS-3. Although raingauges would give an accurate measurement of rainfall, radar (S-band) was used since it can give a fairly accurate measurement of rainfall. The radar was calibrated from a previous study relating echo intensity with raingauge measurements. Again, convective systems were studied. A densitometer was used to digitize photographs. A TV camera is aimed at the photograph and video processing translates photograph density into a colour coding scheme. Chosing a cloud brightness threshold value above which produced radar echoes was a problem, although this study showed that the greater the maximum cloud brightness was, a greater percentage of clouds had echoes. This relationship was found to be time dependent, in the life cycle of cloud systems. Maximum cloud size occured after maximum echo size and the echoes were smaller and lasted a shorter period than what was visible from the satellite. The results were favourable, with satellite measurements over at least 5 hours were within a factor of 1.5 of radar and raingauge measurements.

1.5 A NEW APPROACH

The major problem in previous approaches was that the amount of data was very large and much of it was not handled through computer techniques but rather stand alone densitometers were used for enchancement. If computers were used, a major problem was the lack of convenient display devices. The ideal way of processing the data is to use digital techniques throughout, from the raw satellite and radar data to final figures and displays. Computer processing allows various techniques to be applied to large amounts of data not possible by hand. Although computer techniques and image displays are available, they are generally limited to larger, expensive and less accessible systems.

A small system was developed which included a microcomputer and an image display. This allowed extreme flexibility in processing and displaying image data, yet could be used for a task previously not possible due to cost restraints.

CHAPTER 2

2.1 CURRENT TECHNOLOGY IN IMAGE DISPLAYS

2.1.1 Commercial Display Systems

In 1978, there was a very substantial increase in commercially available image display systems, utilizing a standard colour or black and white television monitor and a large (32 kilobytes or larger) memory for screen refresh. Image display systems are becoming popular due to the large reduction in solid state memory cost and increasing density and speed of RAM devices. Manufacturers offering such display systems include Ramtek Corp., Lexidata Corp., and Norpak Ltd.

Manufacturers have a wide variety of features and options for their display systems. A very large percentage of image and graphic display systems are microprocessor-based. Although these systems are often interfaced with a host minicomputer, the internal microprocessor performs various functions in manipulating the image data. Image displays are available in a wide variety of picture element density and number of bits (colours) per element, depending on the application and allowable cost. The range of application for image display systems is unlimited, the more notable applications are in the areas of medicine (e.g. tomography) and remote sensing (e.g. satellite imagery). Image display and graphic display systems must be differentiated.

Graphic displays are more widespread in use, particularly for computer outputs. Typically, graphic displays are used for drawing single colour lines (straight or curved) with overlay characters and symbols on a colour background. A given line, character and background may be one of a set of available colours. Since each display element has few number of bits assigned to it, the total number of elements is usually large. A good density (resolution) would be 1024 x 1024 horizontal and vertical elements. A graphics display is more complicated than just a simple array of picture elements, each with a certain number of data bits. Instead, the data for a graphics display is usually "compressed". For example, a line is specified by its colour and its start and end points (i.e. a vector) rather than being specified element by element. An entire character consisting of many elements only requires colour, position, size and character code to be specified. In other words, the information for a graphics display is coded. Hence a graphics display requires considerably less memory capacity than an image display of the same element density. The simplest display is a basic matrix of picture elements, essentially a one bit per element image display.

Image displays require much more memory due to the random nature of the data and many colours or levels per element. A minimum element density is generally considered to be 256 x 256 elements, with 4 bits per element or 16 levels or colours. At a density of 512 x 512, the "digitization effect" is almost invisible. 1024 x 1024 elements produce a superb quality image with up to 4096 colours.

Typical display functions available include vertical and horizontal scrolling and scaling, programmable grey level and colour translation tables, character and symbol generation, vector drawing and cursor functions. A user can control these functions using high-level language type commands for interactive processing. Standard Fortran programs for the host computer used for data acquisition and analysis can be linked to the display controller for output purposes. Although some displays are a sequential write-only device, many have randomly accessible memory. The memory used for display refresh can also be used as read/write data memory for processing purposes. Data may be transferred sequentially using a serial interface or DMA (direct memory access).

Since many data bits are required to represent the large number of colours distinguishable by the human eye, display systems often have a restricted set of displayable colours, but this set of colours may be reprogrammed using colour and level translation tables. For example, a display system may offer 4096 colours (Lexidata 6400) per element, but these 4096 colours are chosen as a subset of $2^{24} - 1$ possible colours. A colour translation table inputs a given colour code word (as an address) and produces 3 data output words, each of which produce a red, green, or blue video signal through a digital to analogue converter. Hence, each colour code word is "mapped" onto a corresponding colour in an arbitrary fashion defined by the translation table.

2.1.2 Special Systems

Due to the wide variety of image processing and display systems, there are many specially designed systems in use, of varying degrees of complexity. A problem in image processing is the large amount of data to be processed, sometimes by complex algorithms. When the constraint of real time processing is included, very powerful and fast processors are required. For this purpose, array and parallel processors have been developed. Consider an image with an array of 512 x 512 elements with a refresh rate of 30 Hz. The real time data rate is about 8 MHz. As a result, most image processing is not done in real time. Early efforts in image processing, even using large computers, required hours or days of computer time. A desirable yet achievable approach is that of interactive processing, where an operator controls the processing and makes judgements on what is to be done next, on the basis of previous results. A maximum turn-around time for interactive image processing is considered to be about 1 minute. One example of a facility is at the U.S. Army Engineer Topographic Laboratories. Here, the Digital Image Analysis Laboratory consists of a PDP 11/50 minicomputer, a CDC 6400 extended core computer and a PDP 11/20 interfaced with a Staran array processor. In addition, there is a wide variety of peripherals such as tape drives, terminals, displays, disks, etc.

The Staran array processor is designed for extremely fast image processing, being at least an order of magnitude faster than a CDC 6400 computer. It consists of several array modules and a main control unit and memory. Each array module has a multi-dimensional memory (256 of 256-bit words), a permutation network and 256 processing elements. An N-point fast Fourier transform can be performed in $\log_2 N$ steps. What the Staran can process in 2 seconds requires 1552 seconds for the HP3000 computer. The Staran performs basic image processing algorithms such as spatial warp, magnification, convolution, spatial filtering; all basic enhancement techniques as well as pattern recognition and classification.

For example, if an image is to be deblurred, the operator can try various filters based on previous experience. He can try several filters on an interactive basis, with a fast turn-around rate.

Another application of image processing is in astronomy, using similar facilities described previously. One rather unique problem in astronomy is the enormous dynamic range of modern photon detectors with image intensifiers, requiring more than 12 bits of data for each sample. 18 bits are required for representation in linear form. The raw data is input in 20-bit words format. Filtering operations produce more bits of significance than the input data and the precision must be preserved. In typical applications, a 20-bit 256-square RAM is used. One notable application of processing is restoration and deblurring of images which were blurred by atmospheric turbulence.

Many scientific image processing requirements are very demanding of processor performance. Very large scale integration will make large improvements in memory devices and make array processors more powerful and less expensive and hence available to a larger number of applications and users.

2.2 GENERAL SYSTEM OVERVIEW

The basis of the system used in the satellite radar study was an Intellec MDS-800 development system, which uses the 8080 microprocessor for the CPU. The MDS-800 unit is a microcomputer, complete with CPU, 16 kilobytes of dynamic RAM, a 2 kilobyte ROM monitor, peripheral interfaces and power supply. The Intellec MDS chassis and motherboard provide the housing and interconnection for up to 18 circuit cards. The circuit cards plug into the motherboard using 86-pin connectors, which supplies DC power and signal lines. The Intellec MDS is capable of considerable expansion. As well as additional RAM, ROM and I/O, the system is capable of supporting multi-processors and a 16-bit data bus. Fig. 2.1 shows the system block diagram.

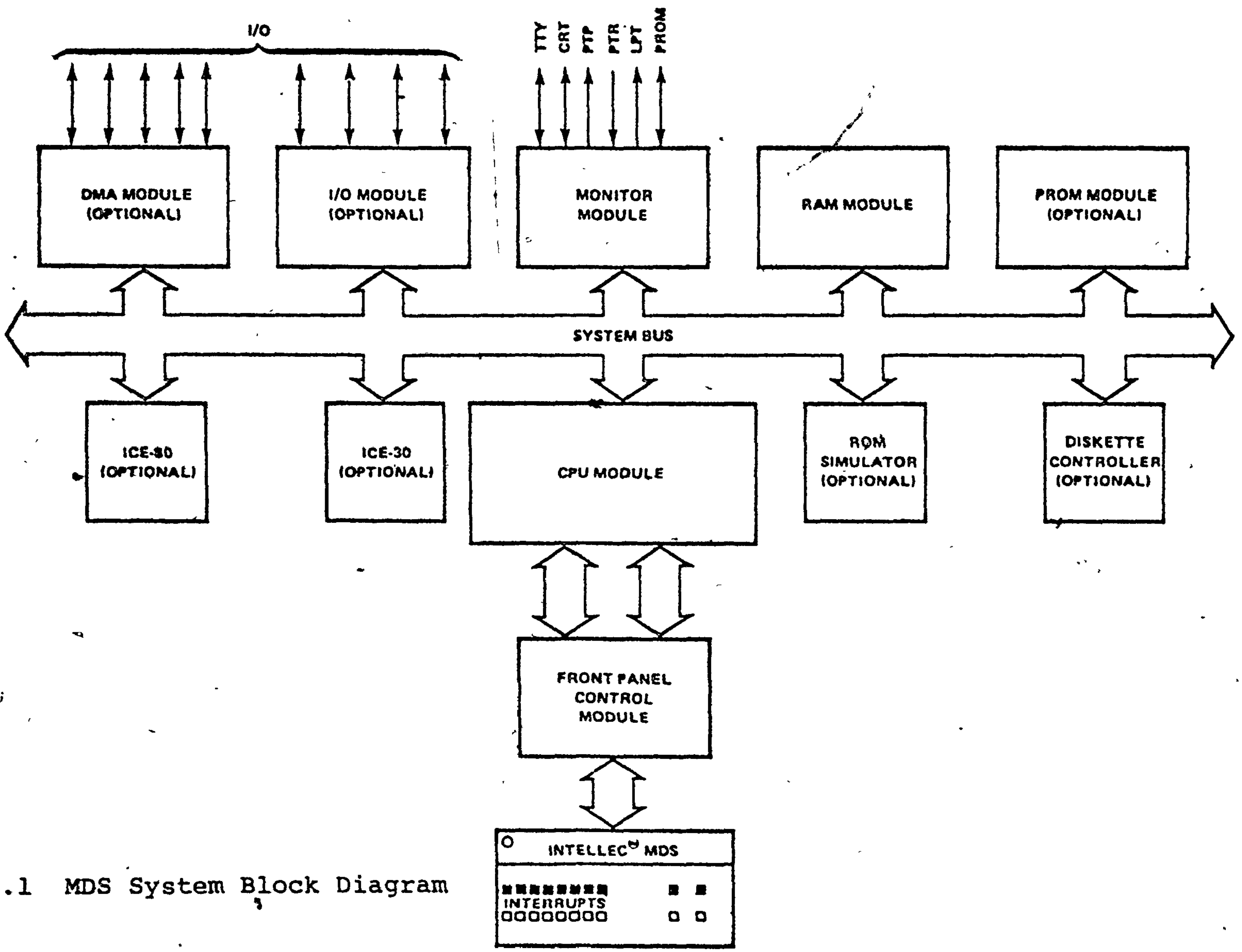


Fig. 2.1 MDS System Block Diagram

The present Intellec MDS system uses an ASR-33 Teletype as the main console device for command and program entry. Programs (both system and user-orientated) are stored on paper tape, which are read in using the high-speed paper tape reader. The remaining basic peripheral unit is a Universal Prom Programmer, which can program a variety of PROM's and EPROM's.

Software development is done mainly in 8080 assembly language. Although Intel-supplied text editor and assembler are available, a much more useful package is the IMSAI 8080 Self-contained system. This is a complete text editor and assembler package which occupies only 4 kilobytes of RAM for object code. Although it only has simple commands, its usefulness and power is due to the fact that programs can be interactively written and debugged in assembly language very quickly. The program is entered and edited using the text editor. The text editor stores the source text in a RAM buffer. The program is then assembled into object code. The assembler read the source code from the text buffer and converts it into absolute object code which is written into RAM with the address specified by the user. Due to this "self-contained" concept, the assembly is essentially instantaneous. The program can then be run and tested, normally using the Intel ROM monitor. Errors are quickly corrected by text editing and re-assembly. The main disadvantage of this type of operating system is that program size is limited by available RAM and is very vulnerable to program "crashes". Hence a new program should be checked-out very carefully the first time it is run, particularly with respect to memory write instructions, stack operation balancing, and iterative loops. For typical user programs, 16-K of RAM is normally sufficient.

For programs numerically orientated, which do not require fast execution, a Basic interpretive language is available. It is simple to use and is an ideal high level language for microcomputers, also being extremely popular. In many respects, Basic is superior to Fortran. The particular Basic language used on this microcomputer system can access particular memory locations and I/O ports directly as program statements. The main disadvantage of Basic is its slow execution speed. Appendix 1 contains a description of operating the assembler/editor and Appendix 2 describes the use of Basic.

The Intellec MDS system has been expanded to include an image display system as shown in Fig. 2.2. The interface and display board has two main functions. First, it interfaces the 128 kilobyte image memory to the colour television monitor by providing screen refresh data from the image memory. Second, it allows the microcomputer to access the image memory. The colour display has a 256 square matrix of elements, with each element having 16 discrete grey levels or colours. The remaining 3/4 image memory capacity is not used for direct display purposes but is used for workspace and image storage, as so many image processing algorithms require additional storage beyond the image itself.

The remaining peripherals and interfaces deal with data acquisition and mass storage. For mass storage purposes, magnetic tape is used. Although cassette recorder interfaces are common with microcomputers, they are too slow for storing image data. Instead, a medium or high performance audio open reel recorder is used. The data transfer rate is 9600 baud, which provides a minimum reasonable data transfer rate of 873 bytes/sec. At this rate, a simple 256 x 256 4-bit display

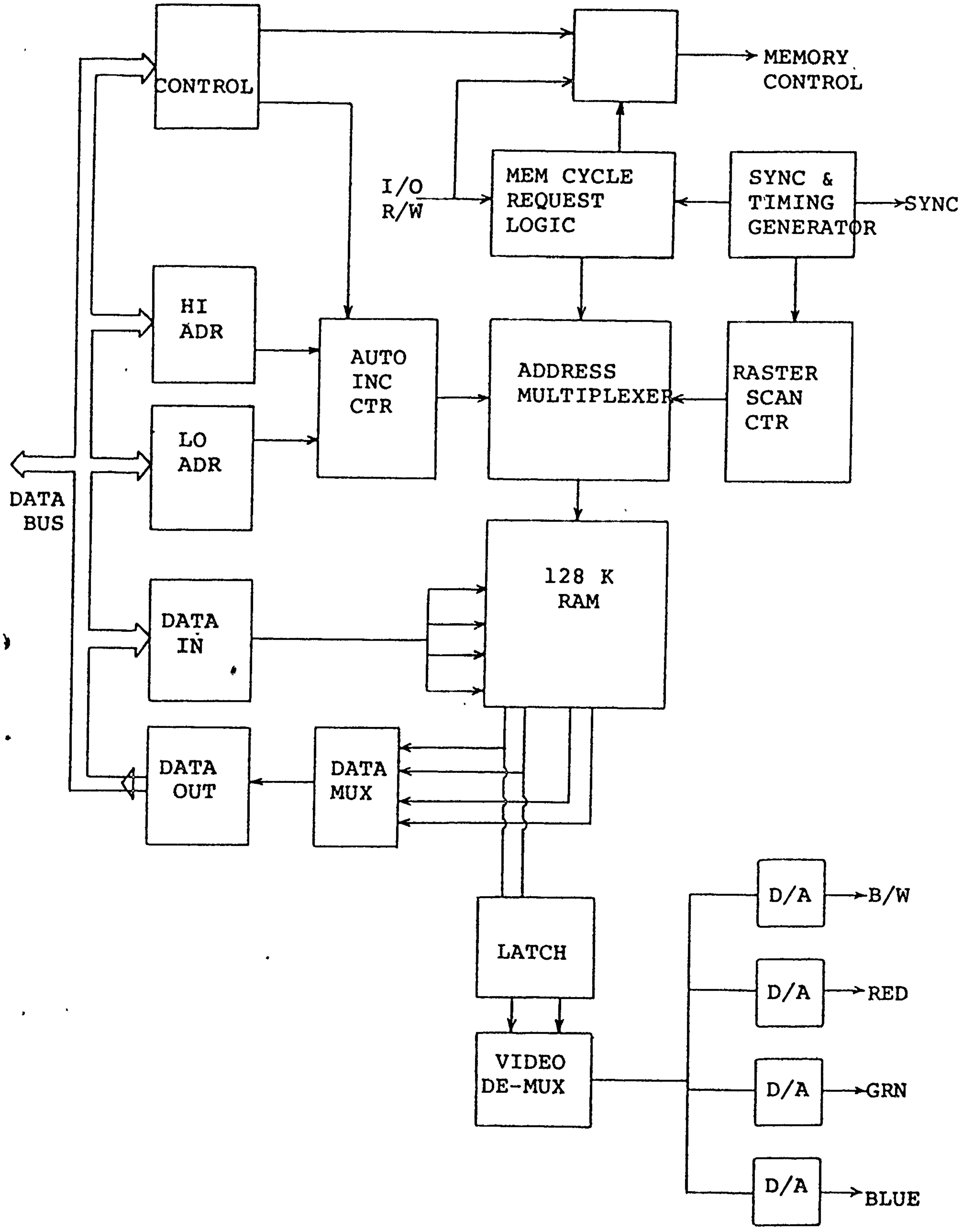


Fig. 2.2 Image Display System Block Diagram

image can be read and written in 38 sec., while the entire 128-K RAM can be transferred in 150 sec. The data format is 9600 baud asynchronous serial data, with 8 data bits and two stop bits. The recording technique is phase-encoding, a very widely used commercial technique. It is reliable and simple to implement. At 9600 baud, the bandwidth requirement is 19.2 KHz., easily achievable on an open reel recorder. A significant advantage of phase-encoding over other standard techniques (such as NRZI) is that no modifications are required to the recorder. The phase-encoded signals are routed through the normal audio inputs and outputs. In contrast to commercial saturation recording, normal analogue recording is used, which achieves sufficient performance and reliability. On a good quality recorder, no pre- or post-equalization is required. Although phase-encoding is a synchronous, self-clocking technique, the data format is asynchronous in order to maintain compatibility with standard serial interfaces, such as the one in the Intellec MDS system. Here a UART is used, which is controlled, written and read by the CPU. Without hardware changes, the 2400 baud CRT serial interface in the MDS system is changed, under software control, to a 9600 baud serial rate. The serial data lines are connected to the recorder via the tape interference unit (it may be conceptually considered as a modem).

Radar data is recorded in two forms; on video magnetic tape and on PPI display photographs. The image processing system can acquire data from both sources. The real-time rate analogue data from the video recorder is too fast for the CPU to handle directly. Instead, the digitized video data is transferred to the image memory using DMA. The memory control signals and memory address are supplied by the video recorder interface unit. The video recorder has available video, ACP, ARP, and system trigger signals. The memory data is supplied by an A/D converter. The present format consists of an entire radar sweep, with 2048 scans and 128 samples per scan. Each sample has a 4-bit code and hence there are 262,144 samples taken for one sweep, filling the entire image memory.

Data in the form of photographs can be digitized using an opto-mechanical scanner, often referred to as a facsimile transmitter. This device has a rotating drum (180 RPM), to which a photograph is attached. The drum is slowly, laterally-fed, which produces a raster scan on the photograph. A high-intensity lamp with beam forming optics, focuses a bright spot of light on the drum. The reflected light is focused onto a pin-hole aperture which selects a small resolution cell to be "seen" by a phototube. The video from the phototube is amplified and low-pass filtered before A/D conversion. The data is then read in by the computer. Normally a photograph is digitized to 256 x 256 elements, each element with a 4-bit code.

2.3 IMAGE DISPLAY HARDWARE

2.3.1 Introduction

The display format chosen was a 256 x 256 element matrix, with each element having a 4-bit code. This format is generally considered a minimum resolution display for image purposes. This was found sufficient for most purposes. In order to present a flicker-free, stable image, the TV screen must be continuously updated or refreshed. In accordance with television standards, the entire display memory (65,536 elements) must be scanned through 60 times per second. This means that consecutive memory locations must be read every 250 n sec. Since the minimum cycle time of the memory unit is 450 n sec., two picture elements (pixels) are read at a time. An 8-bit memory byte will contain two pixels. The 8-bit data is de-multiplexed into two 4-bit words. The first memory location is "mapped" onto the upper-left corner of the display and the last memory location corresponds to the lower-right-corner.

To display the image in black and white mode, the 4-bit pixels produce the TV video signal by a 4-bit D/A converter. A binary code of 0 produces a fully black display pixel and a code of 15 produces a full white pixel, with equal increments of intensity in between. When the TV display is switched to colour mode, it requires 3 video signals corresponding to the 3 primary colours: red, green, and blue. Here, three D/A converters are used. Almost any colour, as perceived by the human eye, can be produced by appropriate combinations of red, green, and blue video levels. White, for example, is produced by all 3 video signals at maximum level. It is important to realize that, to the human eye, two colours may appear identical even though the wavelength spectra of the two sources may be completely different. To the human eye, a source appears "white" whether it is a continuous wavelength source ("flat" spectrum) or three monochromatic red, green and blue sources adding together.

Each colour has 3 dimensions; luminance (brightness), hue (tint), and saturation (purity). In order to display an image in black and white format, at least 16 levels of brightness are required so that the quantization or contouring effect is not objectionable. In colour format, the minimum number of bits in a code word is considerably greater, at least 12. Since the image display system is not designed for high-quality colour viewing but rather pseudo-colour display of remote-sensing data of limited resolution, 4 bits per pixel or 16 colours is sufficient. The colour set chosen to correspond to the 16 binary codes is shown in Fig. 2.3. It consists mainly of fully saturated colours (with the exception of pink) in addition to black and white. The colours are arranged in mainly spectral order as shown in Table 2.1. The colours make a fairly smooth transition from one end of the scale to the other, yet are far apart if they are plotted on a CIE Chromaticy Diagram. In other words, each colour is easily distinguishable from the next yet the human observer can easily associate a colour with, for example, a signal intensity, temperature, or other physical quantities. One advantage of using colours which are fully saturated is that they are easier to photograph, being less critical to exposure time, aperture, or film qualities.

Binary colour code	Display colour
0000	black
0001	purple
0010	violet
0011	dark blue
0100	blue
0101	cyan
0110	green
0111	green-yellow
1000	yellow-green
1001	yellow
1010	orange
1011	red-orange
1100	red
1101	magenta
1110	light-pink
1111	white

Table 2.1 Display colour-coding

2.3.2 Address and Data Formatting

For image processing applications, a major limitation of most 8-bit microprocessors (and many minicomputers) is their 64 K-byte address space, as the case for the 8080. A simple, straightforward addressing scheme was chosen, common to many commercially available image display systems. The x,y image co-ordinate is selected through 2, 8-bit output ports, a total of 16 bits or 64 K-bytes. The ordinary CPU address bus (16 bits as well) is used for its normal purpose of addressing internal RAM and ROM. It must be remembered that the image RAM is external to the microcomputer (a peripheral device) and is not used for storing or executing program code. The two port addressing scheme is ideal for matrix or image operations using a 256 by 256 array, although is much slower than using the normal register-indirect addressing scheme. If the data is not of a two dimensional nature or does not require much storage (e.g. lookup tables), the internal RAM is more desirable.

The image display RAM is actually two independent 64-K byte RAM units, which will be called channels (channel 0 and channel 1). For simplicity, the address lines have been made common. However, the data lines are separate. In order for the microcomputer to read or write to memory, after setting-up the address, an input and output port is used, transferring 8 bit data. In order to select which channel is used, a fourth output port is used. This serves as a control port, providing several functions, as shown in Table 2.2. Table 2.3 is a simple assembly language program for filling the entire memory with a constant.

Port		Function
80H	output b0	unassigned
	b1	unassigned
	b2	0 = chan 0, 1 = chan 1
	b3	1 = select chan 0 after write
	b4	1 = select chan 1 after read
	b5	1 = increment address after write
	b6	1 = increment address after read
	b7	unassigned
81	output	High-order address (Y co-ordinate or row #)
82	output	Lo-order address (X co-ordinate or column #)
83	input and output	8-bit data read and write

Table 2.2 I/O Port Functions

Since the output ports are latched, the control and address lines remain constant until a new output instruction is executed. Hence address or control output is only necessary when changes are required.

The image display uses the lower 4 bits (b_0-b_3) of memory data in channel 0. The upper 4 bits of channel 0 and all of channel 1 are "invisible". Two images may be stored in channel 0, each occupying lower and upper byte halves, one being displayed and the other not.

2.3.3 Memory Cycle Sharing

The memory is cycled near its maximum rate (450 n sec.). If the microcomputer requires a memory access, it "steals" a memory cycle from the display refresh operation, although the display address counter for the memory remains running. The memory address switches from the counter to the two output ports of the microcomputer for this one cycle.

```
3000                0000 *PR0GRAM T0 FILL RAM WITH C0NSTANT
3000                0001 *AND DETECT MEM0RY READ/WRITE ERR0RS
0000                0010 C0NST  EQU 0       ;CLEAR MEM0RY
0080                0020 CTRL   EQU 80H     ;C0NTR0L P0RT
0083                0030 DATA   EQU 83H     ;DATA I/0 P0RT
0081                0040 HIADR  EQU 81H     ;HI (Y) ADDRESS
0082                0050 L0ADR  EQU 82H     ;L0 (X) ADDRESS
0006                0055 SP     EQU 6
3000                0060        0RG 3000H          ;SET 0BJECT C0DE ADDRESS.
3000 31803E         0100        LXI SP,3E80H       ;SET STACK
3003 210000         0110        LXI H,0            ;SET Y,X=0,0
3006 0E00           0120        MVI C,C0NST        ;SET C=C0NST
3008 7C             0130 0UTER:M0V A,H             ;SET HI ADDRESS
3009 D381           0140        0UT HIADR
300B 7D             0150 INNER:M0V A,L             ;SET L0 ADDRESS
300C D382           0160        0UT L0ADR
300E 3E00           0170        MVI A,0            ;CHAN 0 M0DE
3010 D380           0171        0UT CTRL
3012 CD2730         0172        CALL SUB
3015 3E04           0230        MVI A,4            ;CHAN 1 M0DE
3017 D380           0240        0UT CTRL
3019 CD2730         0250        CALL SUB
301C 2C             0260        INR L              ;X=X+1
301D C20B30         0270        JNZ INNER          ;TEST F0R END 0F LINE
3020 24             0280        INR H              ;NEXT R0W
3021 C20830         0290        JNZ 0UTER          ;MUST SET L0&HI ADDRESS
3024 CD0000         0300        CALL 0             ;RETURN T0 R0M M0NIT0R
3027                0310 *SUBR0UTINE
3027 79             0320 SUB:   M0V A,C            ;WRITE C0NST T0 RAM
3028 D383           0330        0UT DATA
302A DB83           0340        IN DATA            ;READ BACK
302C B9             0350        CMP C              ;READ SAME AS WRITTEN?
302D C8             0360        RZ                 ;RETURN IF 0K
302E 76             0370        HLT                ;HALT 0N ERR0R
```

Table 2.3   Example Assembly Language Program

2.3.4 Image Memory Interfacing

The image memory is a 128 kilobyte static RAM unit manufactured by Electronic Memories and Magnetics Corporation; the Microram 3400N. This is a self-contained unit with a housing, power supply and cooling fans. The memory capacity can be expanded to 256 kilobytes by adding two more circuit cards. It utilizes 5220-B 4K x 1 static RAM's. For maximum flexibility, the memory is arranged as two independent 64 kilobyte units or channels. Each channel is arranged as 32,384 words by 18 bits. In normal computer memory applications, only 16 bits are used for data and the remaining bits are used for parity and memory write protect indicator. For the image memory system, the remaining 2 bits are unused which has proven useful for correcting failures. The unused RAM IC's can be moved into a socket where a faulty RAM was located. Since the memory is arranged as 32 K-words, there are 15 address lines. All memory read operations result in a word being read to the 16 (or 18) data output lines. In order to read a byte (8 bits), the two halves of a word must be multiplexed into one byte. Hence, the multiplexer select input serves as the 16th address bit when addressing the memory as 64 kilobytes. In the write mode, the two BCL inputs allow data to be written in either or both halves of the data word. Hence, each half of the word can operate independently of the other.

In order to maintain compatibility with TTL logic, an interface unit is used between the memory unit and the main processor. Connectors are mounted on the front panel for signal lines. 5 volt DC power is supplied by the memory unit. In order to connect the interface unit to the processor, the use of flat ribbon cable is recommended, with .05 inch conductor spacing. Signal lines should alternate with ground lines. This allows good transmission of high speed logic signals. The interface unit connect pin list and front panel diagram is given in Appendix 3. Also, a description of signal line functions is given. Refer to the Microram manual for information on the memory unit.

2.3.5 Hardware Description

Fig. 2.3 shows the block diagram for the display and computer interface hardware. The sync and timing generator provide synchronization for the TV monitor, memory cycle pulses and refresh address update. For computer interfacing, four output ports and one imput port is used. The I/O port hardware was duplicated from part of the Intel I/O module logic, although the Intel I/O module offers four input and four output ports. A description is given in the Hardware Reference Manual, Chapter 9. Figs. 2.4, 2.5 and 2.6 show the I/O timing diagrams and block diagram. The Intel Multibus has a constant clock, CCLK, signal and an acknowledge line, XACK, independent of the normal CPU signals to allow flexible asynchronous data transfer. Fig. 2.7 is the I/O port decoding logic and data transfer acknowledge generation. The Multibus signals are buffered with low power Schottky logic. The address decoder D1 generates port select signals SEL1 through SEL31 which enable I/O ports 80 through 83 (hexadecimal). Table 2.4 describes the Multibus signal functions. Fig. 2.8 shows P80-0, the control port and P83, the memory data output port (for memory write). P83 drives the 32 memory data lines of both channels.

Fig. 2.9 is the logic for reading data from the memory. The 32 memory data output lines are multiplexed to 8 lines for the input port P83. M1 through M4 select memory channel and byte half of the memory

	(COMPONENT SIDE)			(CIRCUIT SIDE)		
	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	GND	Signal GND	2	GND	Signal GND
	3	VCC	+ 5 VDC	4	VCC	+ 5 VDC
	5	VCC	+ 5 VDC	6	VCC	+ 5 VDC
	7	VDD	+ 12 VDC	8	VDD	+ 12 VDC
	9	VXI	Supply Spare 1	10	VXI	Supply Spare 1
	11	GND	Signal GND	12	GND	Signal GND
BUS CONTROLS	13	BCLK/	Bus Clock	14	INIT/	Initialize
	15	BPRN/	Bus Pri. In	16	BPRO/	Bus Pri. Out
	17	BUSY/	Bus Busy	18	BREQ/	Bus Request
	19	MRDC/	Mem Read Cmd	20	MWTC/	Mem Write Cmd
	21	IORC/	I/O Read Cmd	22	IOWC/	I/O Write Cmd
	23	XACK/	XFER Acknow	24	INH1/	Inhibit 1 disable RAM
				26	INH2/	Inhibit 2 disable PROM or ROM
SPARES	25	AACK/	Special	26		
	27			28		
	29			30		
	31	CCLK/	Constant Clock	32		
	33	INTR/	Direct Int	34		
INTERRUPTS	35	INT6/	Parallel	36	INT7/	Parallel
	37	INT4/	Interrupt	38	INT5/	Interrupt
	39	INT2/	Requests	40	INT3/	Requests
	41	INT0/		42	INT1/	
ADDRESS	43	ADRE/		44	ADRF/	
	45	ADRC/		46	ADRD/	
	47	ADRA/	Address	48	ADRB/	Address
	49	ADR8/	Bus	50	ADR9/	Bus
	51	ADR6/		52	ADR7/	
	53	ADR4/		54	ADR5/	
	55	ADR2/		56	ADR3/	
	57	ADR0/		58	ADR1/	
DATA	59	DATE/		60	DATF/	
	61	DATC/		62	DATD/	
	63	DATA/	Data	64	DATB/	Data
	65	DAT8/	Bus	66	DAT9/	Bus
	67	DAT6/		68	DAT7/	
	69	DAT4/		70	DAT5/	
	71	DAT2/		72	DAT3/	
	73	DAT0/		74	DAT1/	
POWER SUPPLIES	75	GND	Signal GND	76	GND	Signal GND
	77	VBB	-10 VDC	78	VBB	-10 VDC
	79	VX2	-12 VDC	80	VX2	-12 VDC
	81	VCC	+5 VDC	82	VCC	+5 VDC
	83	VCC	+5 VDC	84	VCC	+5 VDC
	85	GND	Signal GND	86	GND	Signal GND

Table 2.4 MULTIBUS Signal Functions

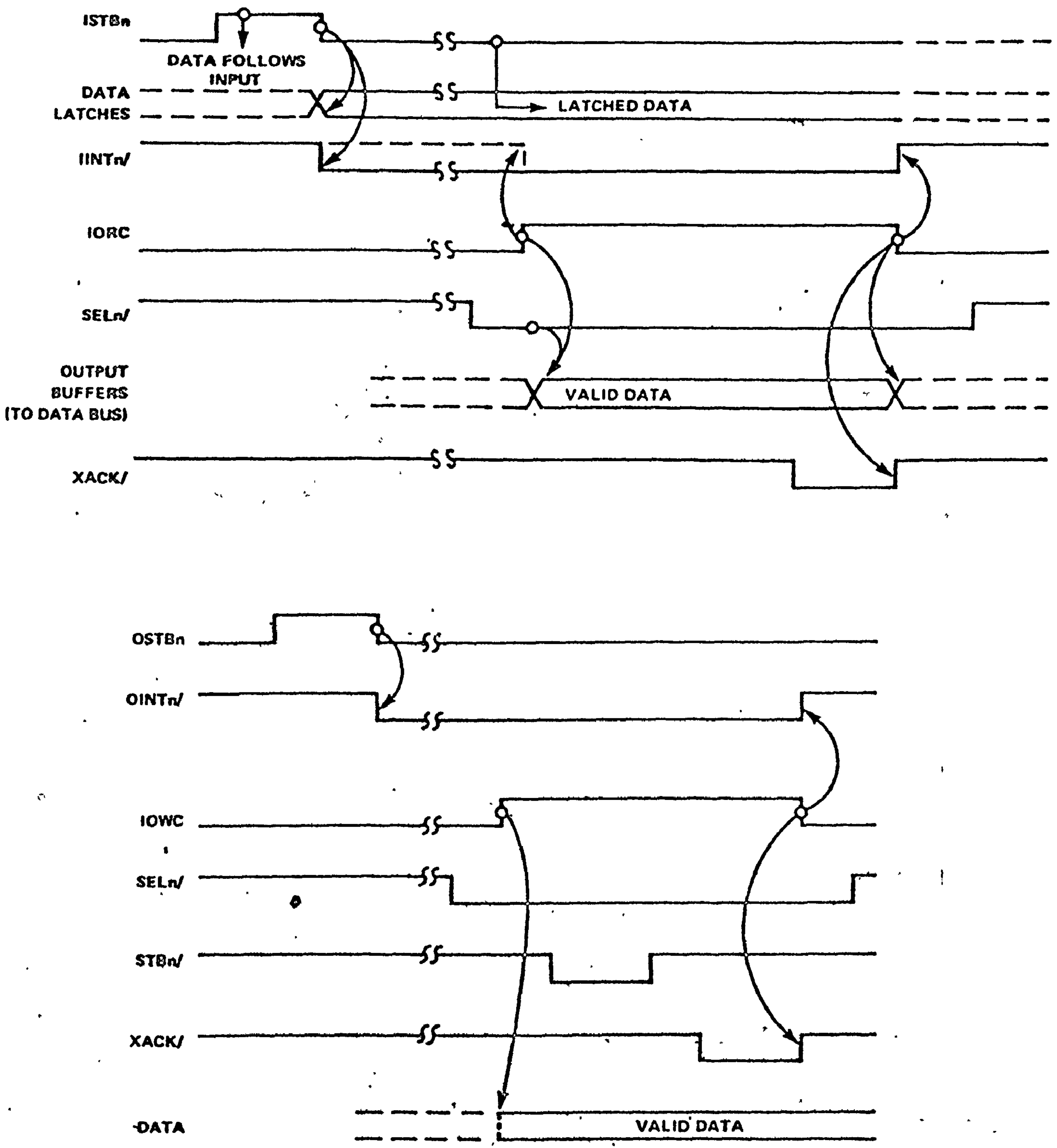
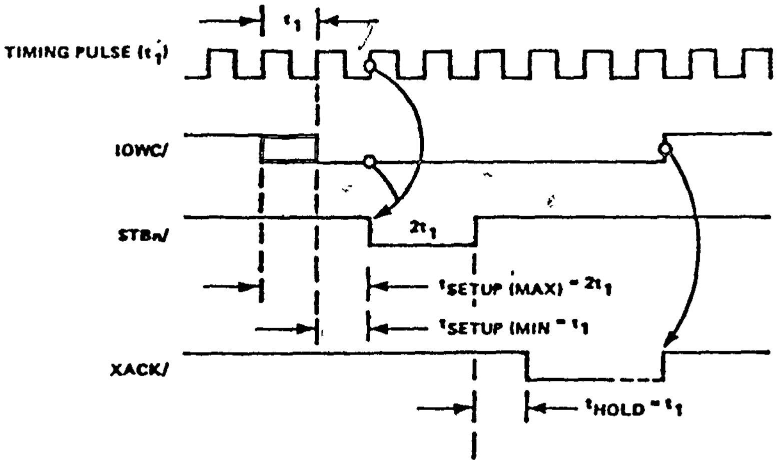
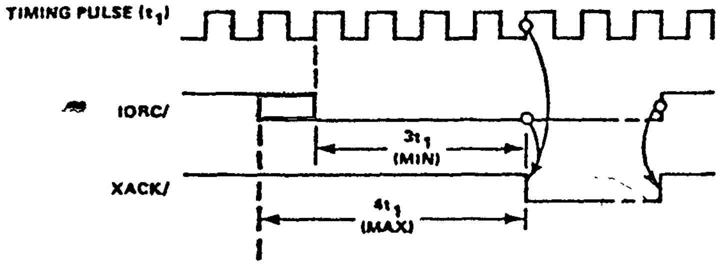


Fig. 2.4 Input and Output Port Timing Diagram



OUTPUT CYCLE



INPUT CYCLE

Fig. 2.5 XACK generation for MULTIBUS

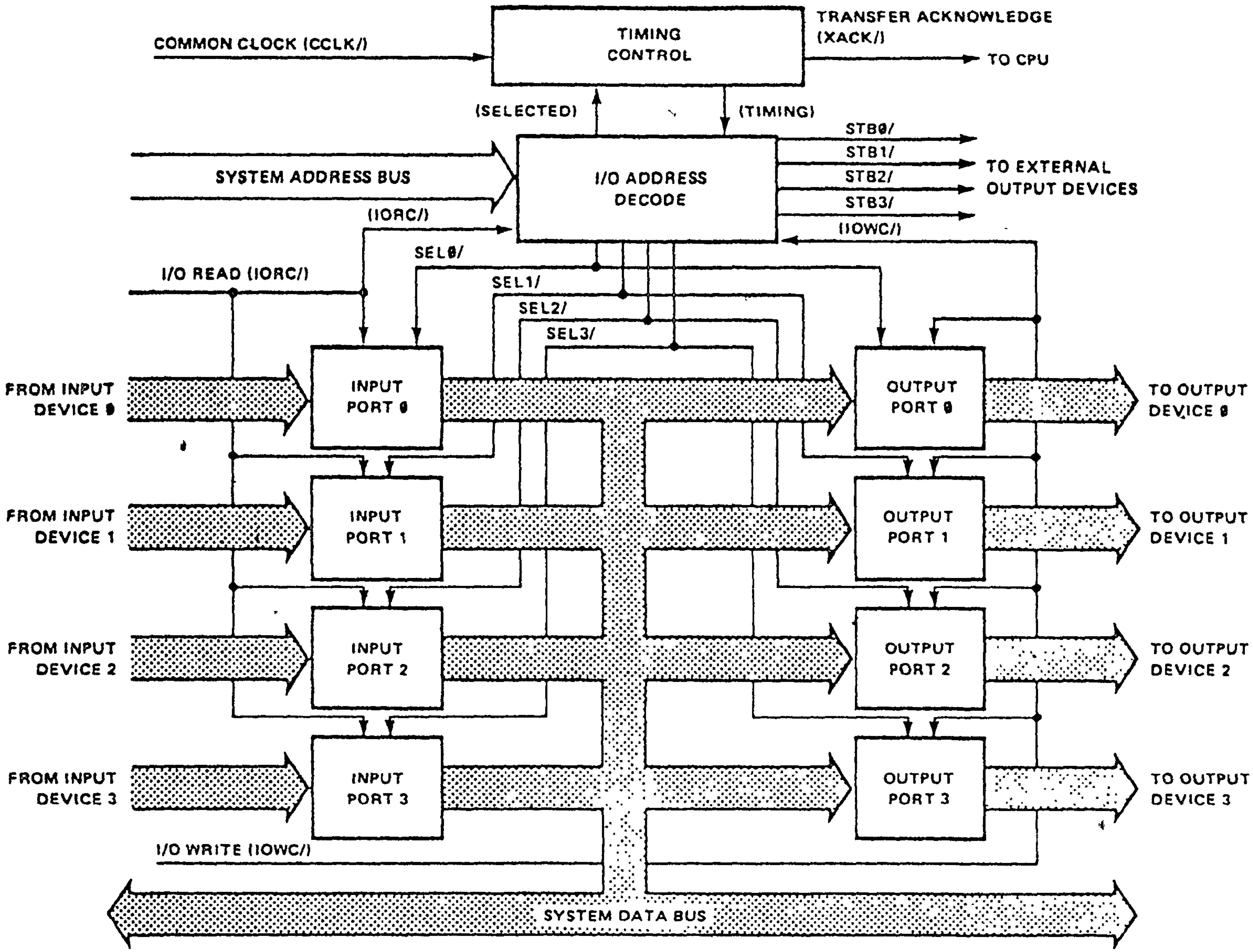


Fig. 2.6 I/O Port Block Diagram

word since all memory reads produce a 16-bit data word. Data is strobed into P83-I when a read is requested (by an input instruction) and when the data is valid from the memory. The computer must wait until valid data from the memory is acknowledged.

Fig. 2.10 shows the circuit for the timing generation. A 10 MHz crystal-controlled clock is divided by FF7 and C3 to produce the 5 MHz scan clock and 2 MHz clock for the MM5321 TV sync generator. The TV sync generator produces the complex sync, blanking, horizontal and vertical drive signals conforming to commercial TV standards. FF2 generates the 2.5 MHz (400 n sec.) memory cycle pulses.

Fig. 2.11 is the circuit for the control functions. As seen in Fig. 2.18, the channel bit C2 is clocked into FF3 when an output to port 80 is executed. If C4 is high, enabling M2 to be triggered, FF3 (and hence CHAN) is set after an input from port 83 (memory read). In a similar manner, FF3 is reset if C3 is high after an output to port 83 (write). Hence the channel can be automatically switched after read and write operations.

Fig. 2.19 shows how the auto-increment signals are generated. If bit-C6 is high, the address contained in C4,5,6 and 7 in Fig. 2.13 is incremented after a read from port 83 and if bit-C5 is high, the address is incremented after a write to port 83. C4 and C5 are loaded with the low order address after a write to port 82 and C6 and C7 are loaded with the high order address after a write to port 81. G15 generates a clock signal from INC or when P82-0 or P81-0 is written to. C4,5,6 and 8 load a preset count when LD is low and CK clocked. If C4 and C5 are being loaded, C6 and C7 are disabled from being clocked using P. The resistor and capacitor on the outputs of G16 and G17 generate a delay (200 n sec) to prevent "race" conditions. Note that all the I/O ports and many counters and flip-flops are reset when the computer is reset from the front panel switch.

In order for the display screen to be refreshed, the image memory is continuously read (or cycled) every 400 n sec. When a computer reads or writes, a memory cycle is "stolen" from the screen refresh. C8,9,10 and 11 in Fig. 2.14 contain the refresh address. These counters are clocked independent of computer memory access so that refresh scanning resumes after the computer steals a cycle with the same address if the computer had not made an access. Since the stolen cycle disrupts the stream of video data, each computer access causes a "glitch" on the display. Multiplexers X5,6,7 and 8 switch the memory address between the refresh counter address and the computer address with the MEM BUSY line. Fig. 2.20 shows the timing for memory cycle timing.

FF4 is set if the computer requests a memory access. (Start of a read or write operation) G40 generates the actual memory cycle pulses, RP^1 and RP^2 from SCAN CLK/ and Q2.5 signals. For refresh scanning, channel 0 is used and hence RP^1 contains the cycle pulses every 400 n sec. When FF4 is set, the next available cycle pulse will produce a computer read/write cycle. FF5 is set (MEM BUSY=1) on the next Q2.5/ clock pulse, which then resets FF4 through monostable M5. MEM BUSY sends a cycle pulse to RP^1 or RP^2 depending whether CHAN=0 or 1 respectively. FF5 is reset on the next Q2.5/ pulse. If FF5 is reset, RP^1 contains the cycle pulse independent of CHAN.

FF6 generates a strobe pulse with G8 to latch data into port 83, in Fig. 2.9, from the memory data output. P83-I is strobed 300 n sec. after a cycle is initiated from an input instruction so that the data is guaranteed to be valid. Multiplexers X1,2,3 and 4 select the 8 data bits using A_{15} and CHAN which are effectively the 16th and 17th address bits in addition to A_0 - A_{14} required to address 128 kilobytes of memory. All memory reads produce 16 data bits from the memory since BCL1 and BCL2 are low.

In Fig. 2.15, BSEL, the least significant address bit is used to control memory write operations to either half of the 16-bit data of the memory but not both. BCL1 or BCL2 can only be high (write) if the computer is performing a write operation.

In order to achieve 256 element horizontal resolution at the standard television scanning rate, each element is present in the video signal for 200 n sec. In order to satisfy a minimum cycle time of 400 n sec., two 4-bit elements are read from the memory. This data is latched into L1 and L2 in Fig. 2.15. M4 generates a clock pulse just before the next cycle starts. Multiplexer X9 switches between the two picture element data contained in L1 and L2. G24, G25, G26, G28 and I43 generate a symmetrical switching signal as seen in Fig. 2.21 (X9-5).

V0-V3 are the video data bits which are converted to a black and white video signal in Fig. 2.17 and a colour video signal in Fig. 2.16. The resistor network in Fig. 2.17 form a simple 4-bit linear D/A converter. The load impedance of the video lines are 75 ohms. The TV monitor used uses AC coupled video with DC restoration. The DC reference level is produced from the peaks of the video signal. The BLANK/ signal set the video signal to its maximum value during blanking periods, providing a continuous reference signal for the TV monitor.

D2, D3 and D4 in Fig. 2.16 are 4-to-16 line decoders (open collector). The video voltage for each of the 16 binary codes can be chosen independently with a single resistor. Only one of G0 to G15 is low at a time, the others are an open circuit. Each colour is produced from a combination of the three video voltages. In order to produce the reference signal for the DC restoration, the BLANK/ signal goes high during blanking intervals, turning off all outputs from D2, D3 and D4. Hence, the video signals are at maximum value.

Out of the 63.5 μ sec. interval between horizontal sync pulses, the TV display is blanked for 10.7 μsec for retrace. In Fig. 2.14, C8, 9, 10 and 11 are clocked every 200 n sec during the active scan periods. The BLANK signal holds C8 and C9 in a reset state until the start of the next active horizontal scan period (52.8 μ sec). A memory cycle occurs every second time C8 and C9 are clocked. 256 elements are read in 51.2 μsec. When C8 and C9 reach a terminal count of 255, they are held with CO of C9 high. C10 and C11 (line counters) similarly are held at their final values. They are set by the vertical drive (VDR) signal. When either pair of counters reach their maximum count, the screen is blanked by G19 and G21 (in Fig. 2.15) which sets the outputs of X9 to zero (black). Only 241 lines of the full 256 are visible due to standard TV blanking.

2.4 9600 BAUD TAPE INTERFACE

2.4.1 Computer Serial I/O

Due to the large amount of data to be read, processed and stored, a fast, simple and reliable magnetic tape recording system was

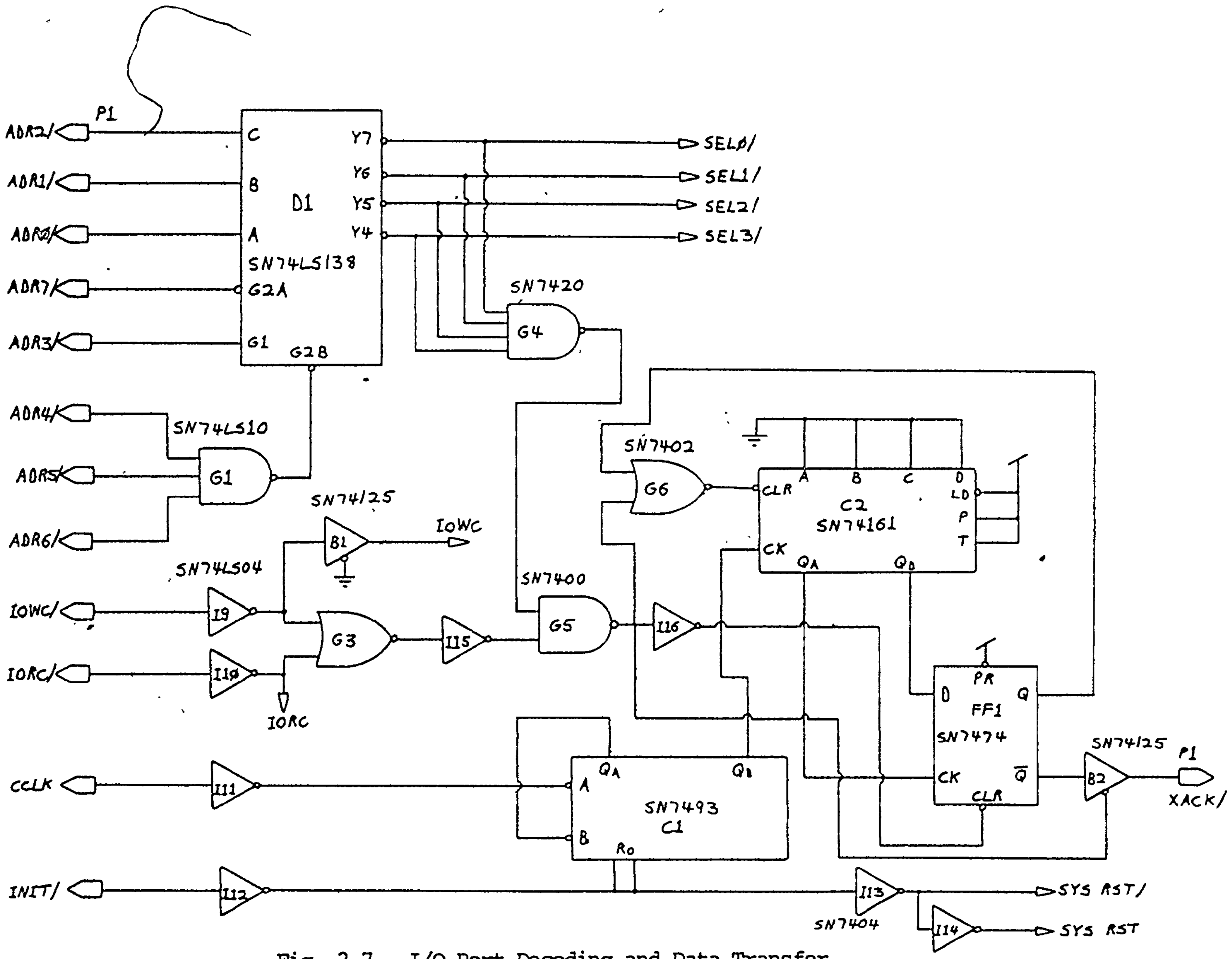


Fig. 2.7. I/O Port Decoding and Data Transfer

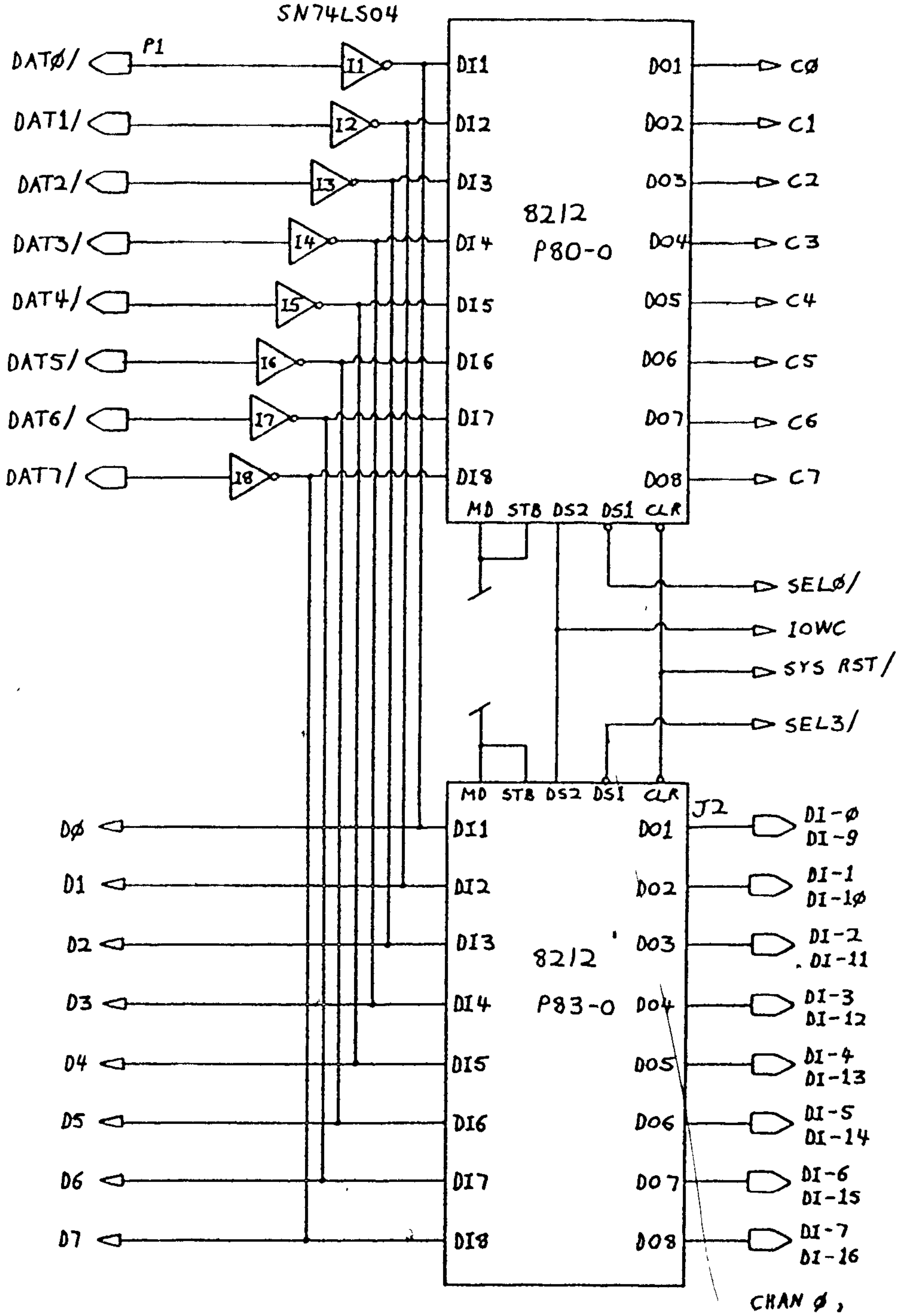


Fig. 2.8. Control and Data Out Ports

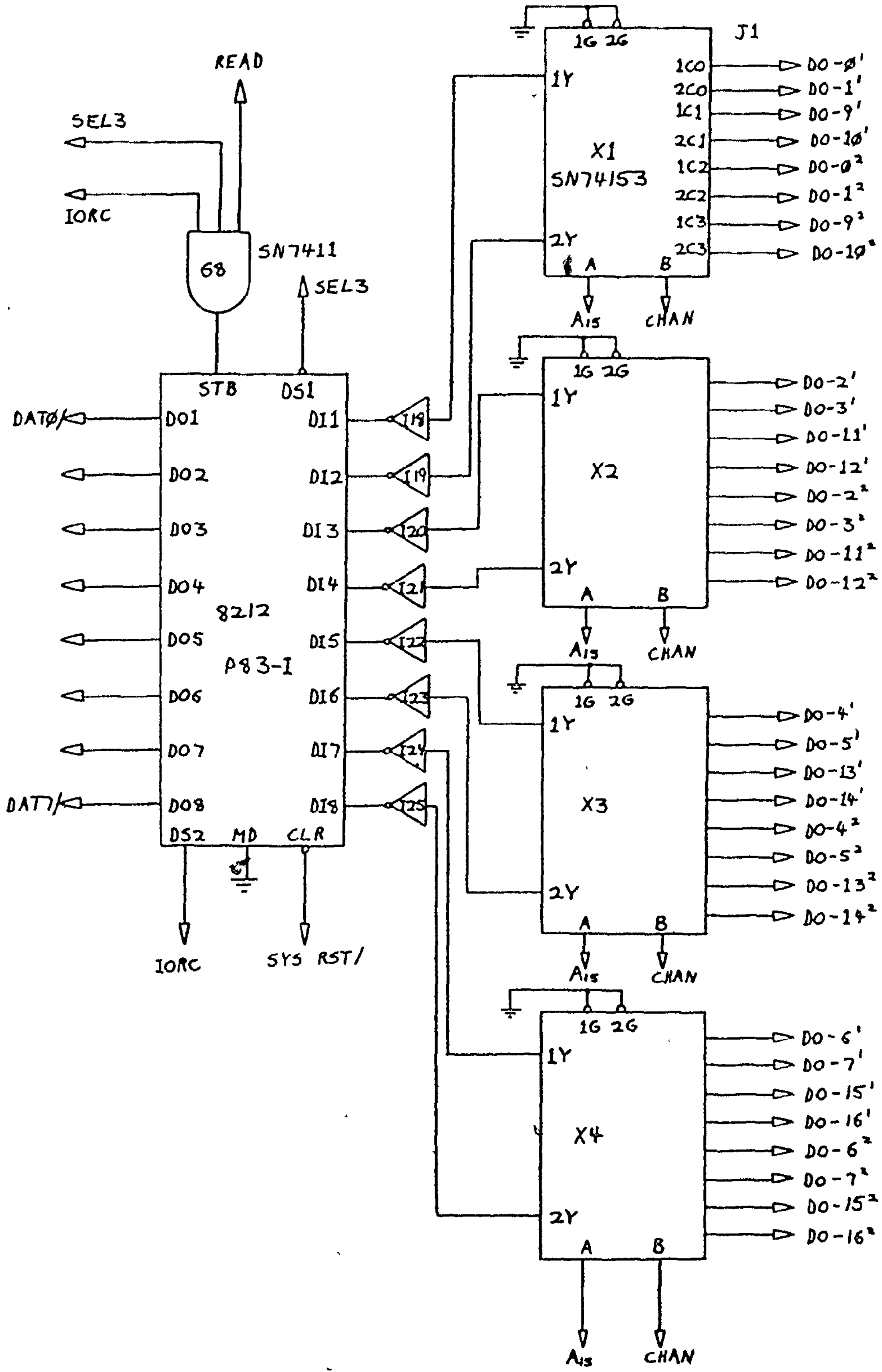


Fig. 2.9. Memory Data Multiplexer and Input Port

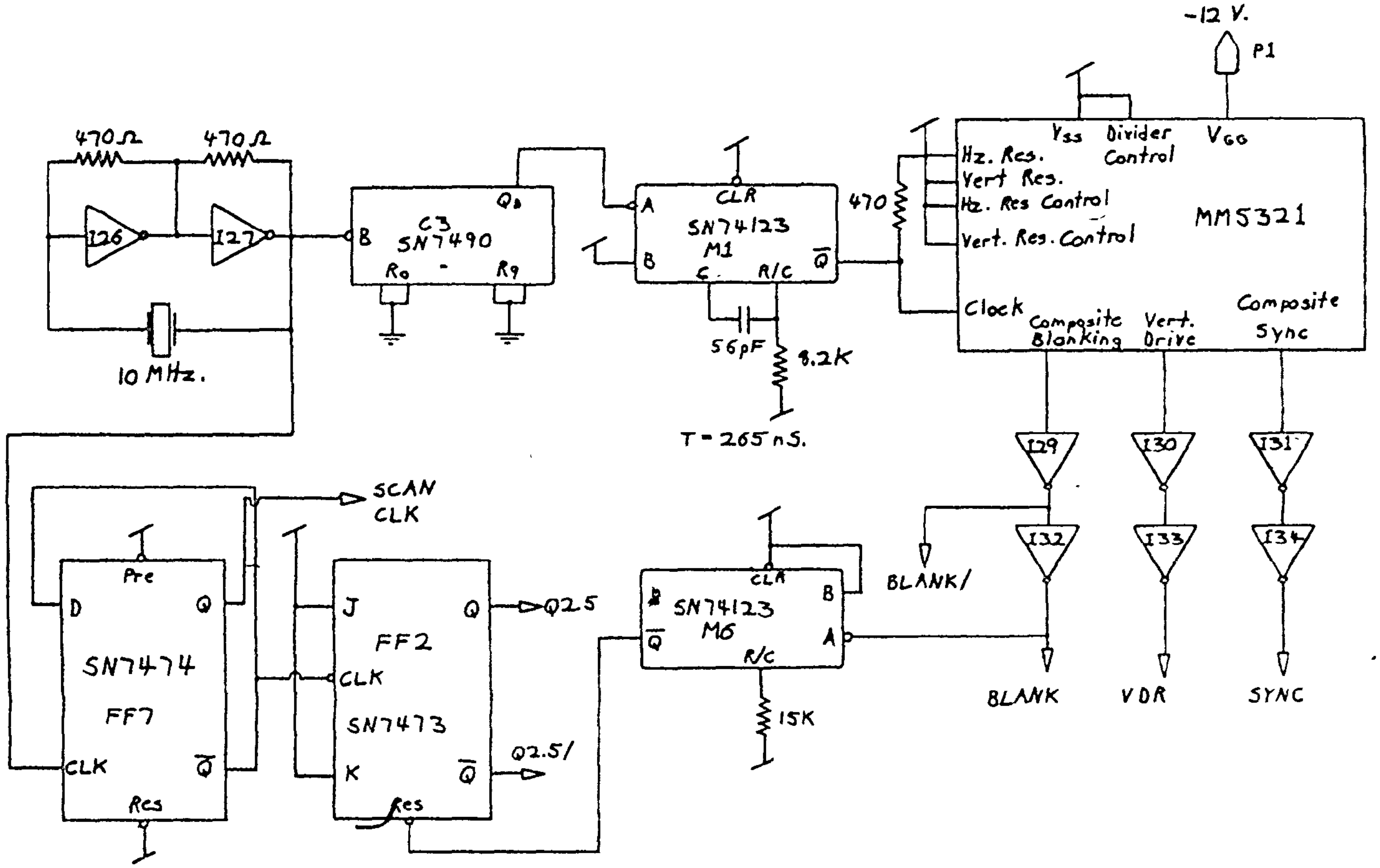


Fig. 2.10. Clock, Synchronization, and Timing Generator

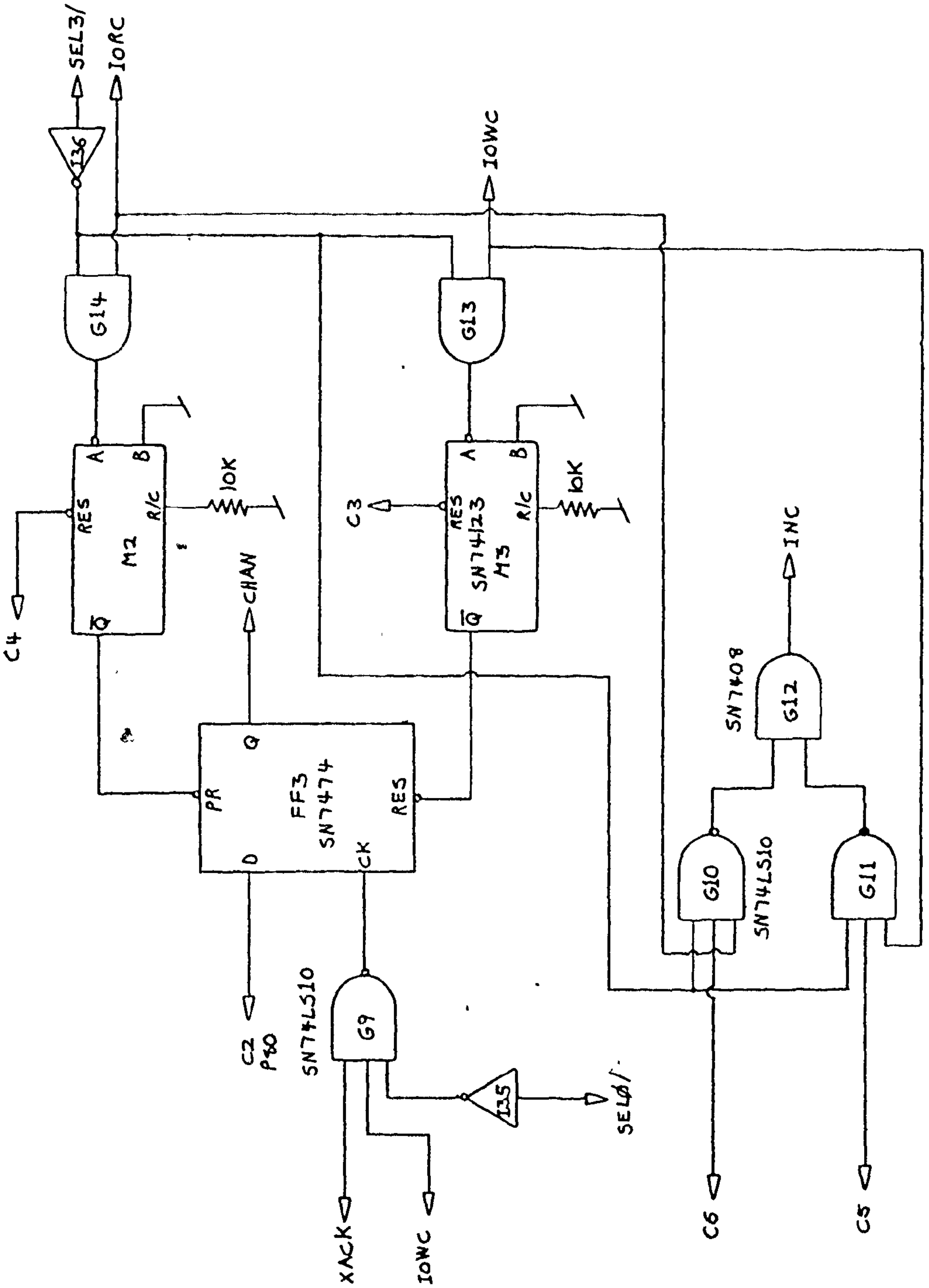


Fig. 2.11. Control Function Logic

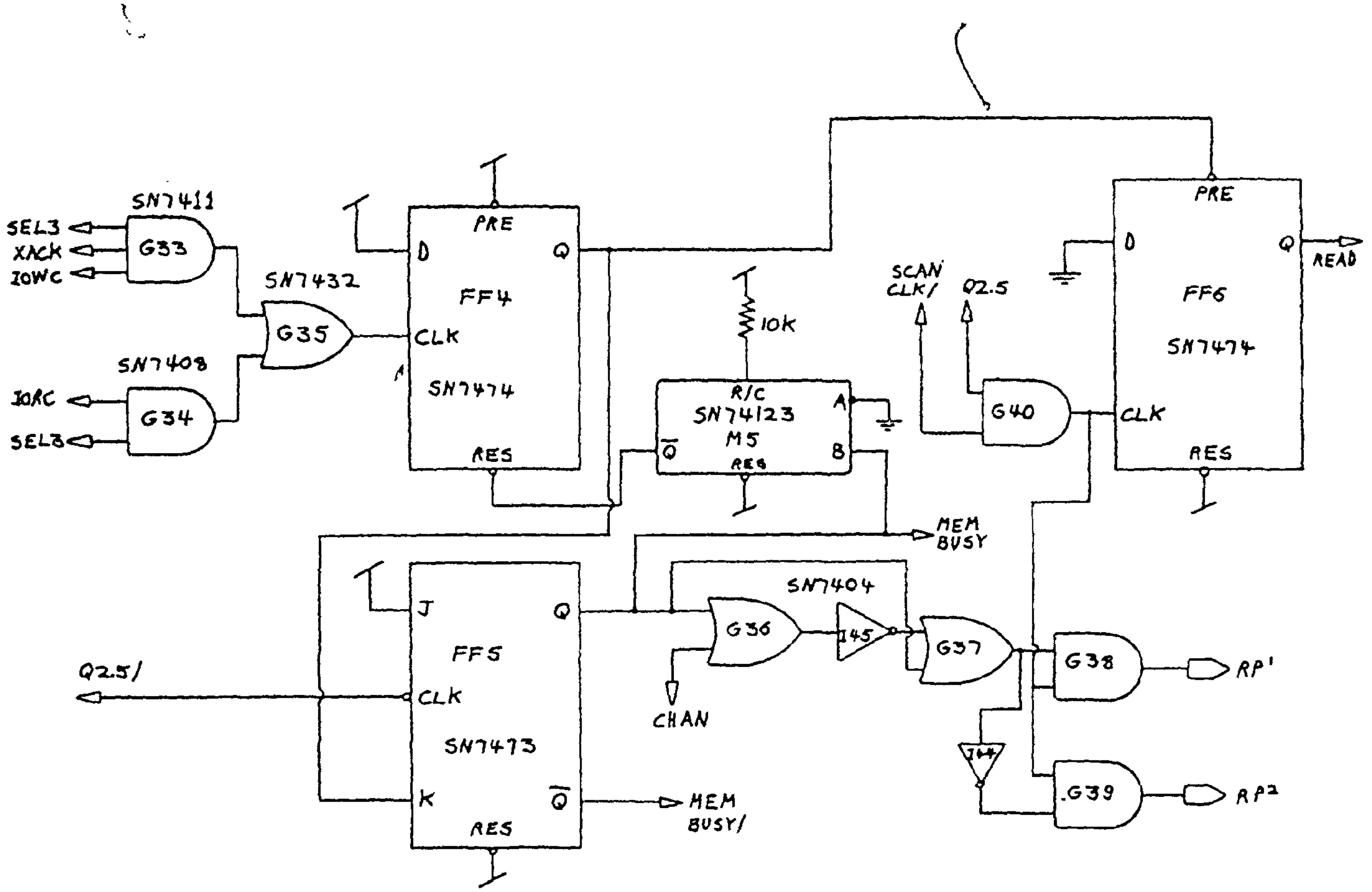


Fig. 2.12. Memory Cycle Sharing Logic

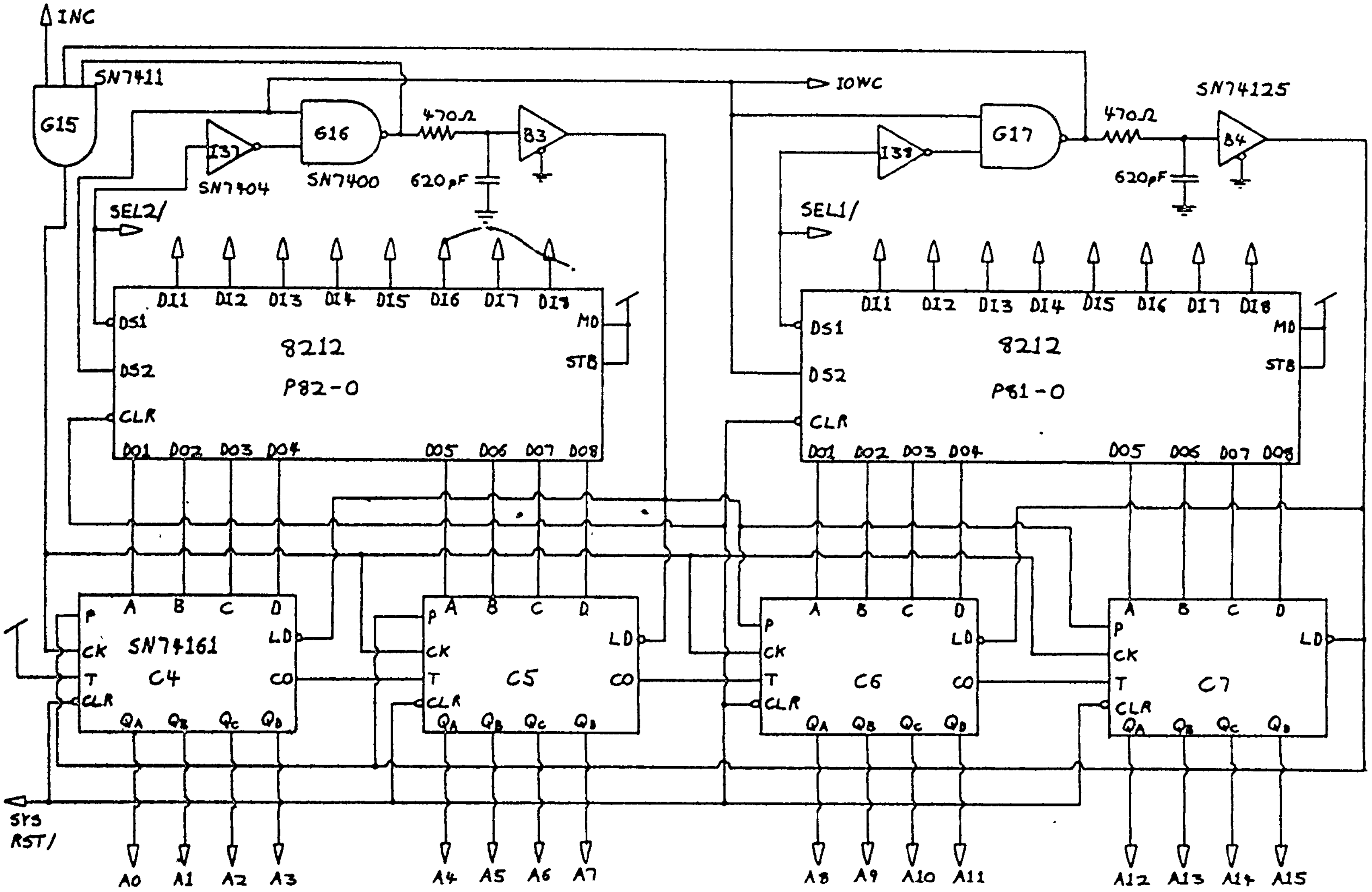


Fig. 2.13. Computer Address Ports and Auto-Increment Registers

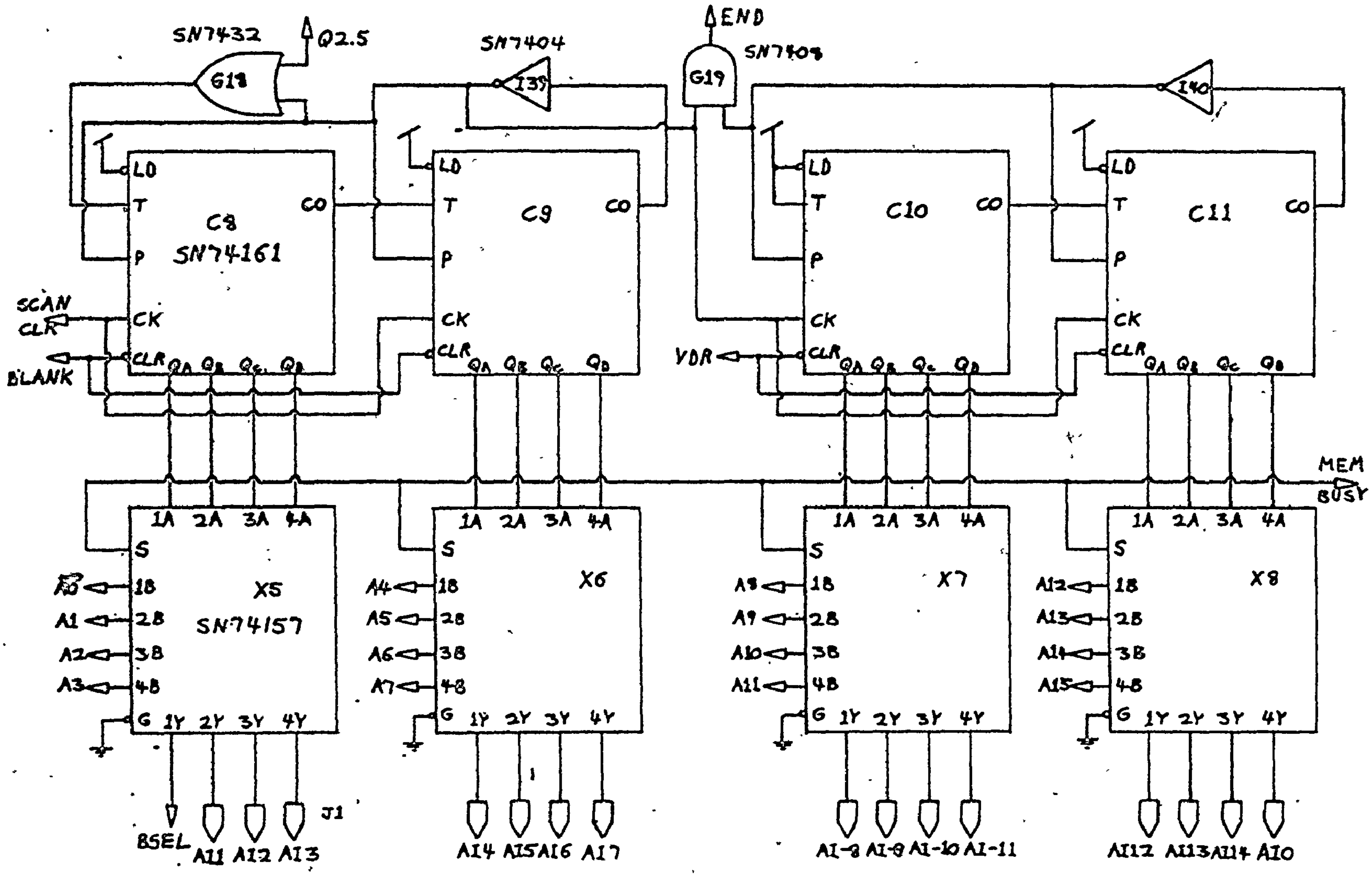


Fig. 2.14. Refresh Address Counters and Address Multiplexers

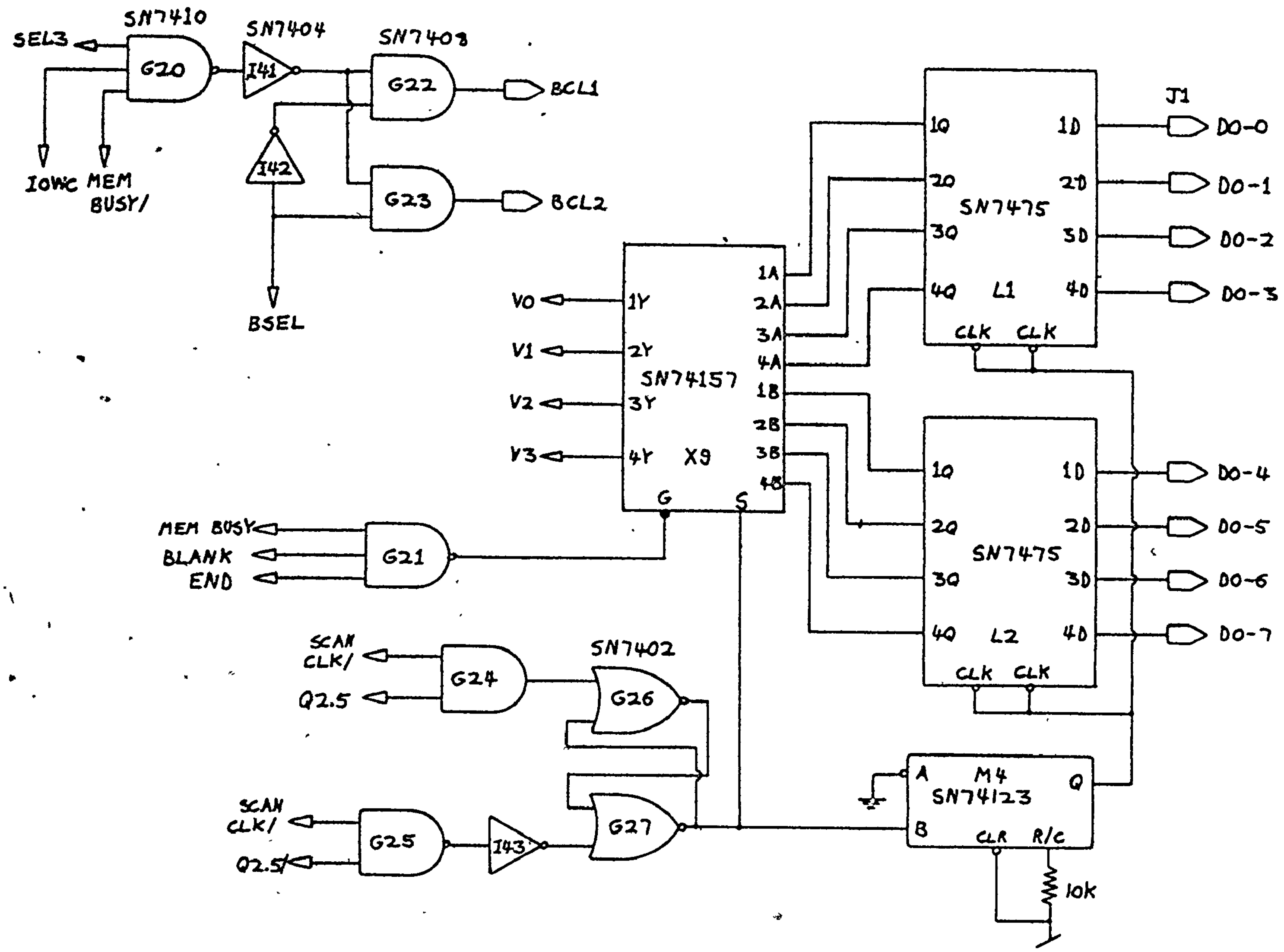


Fig. 2.15. Memory Data Read Latch and Multiplexer

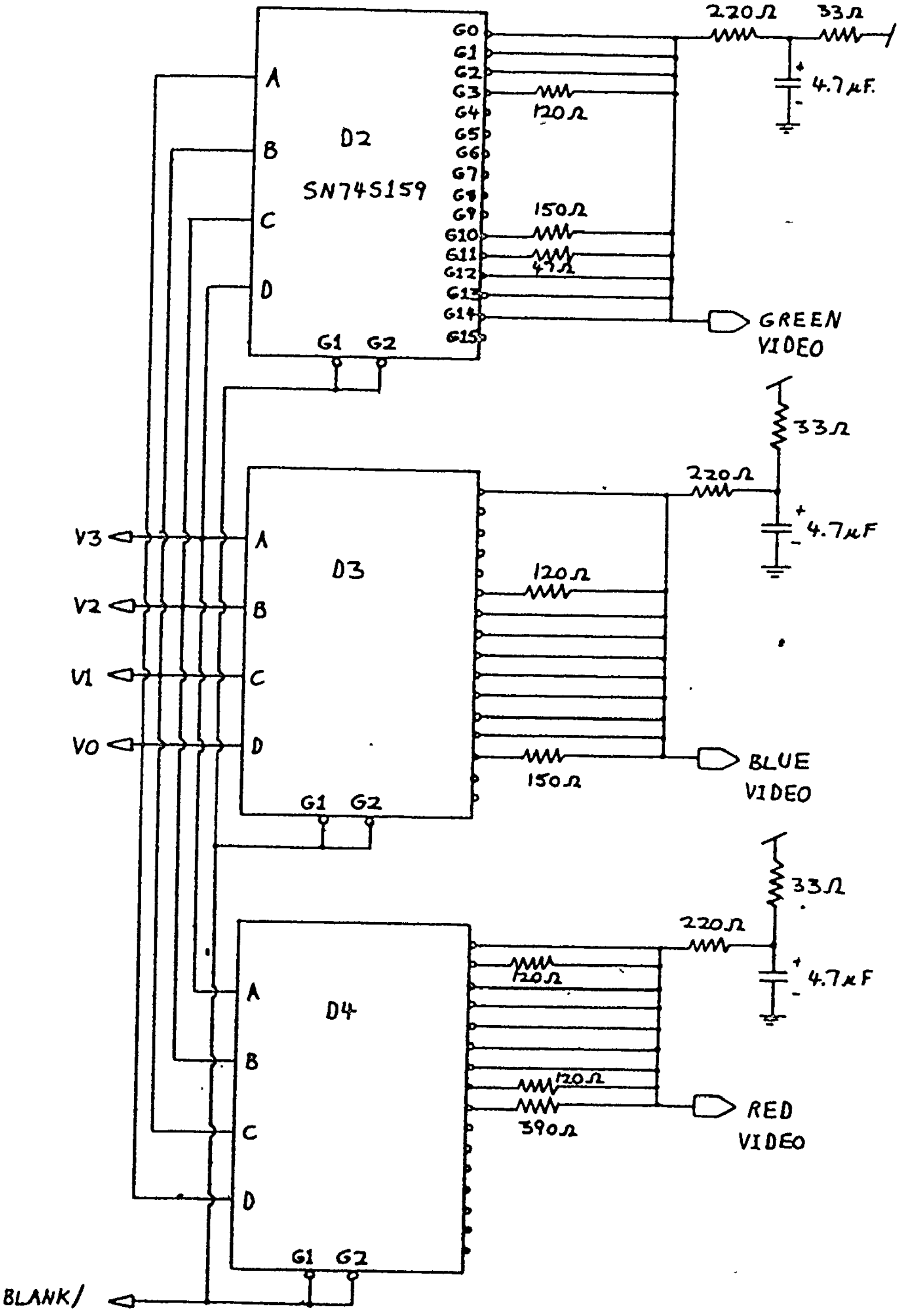


Fig. 2.16. R G B Colour Code D/A Converters

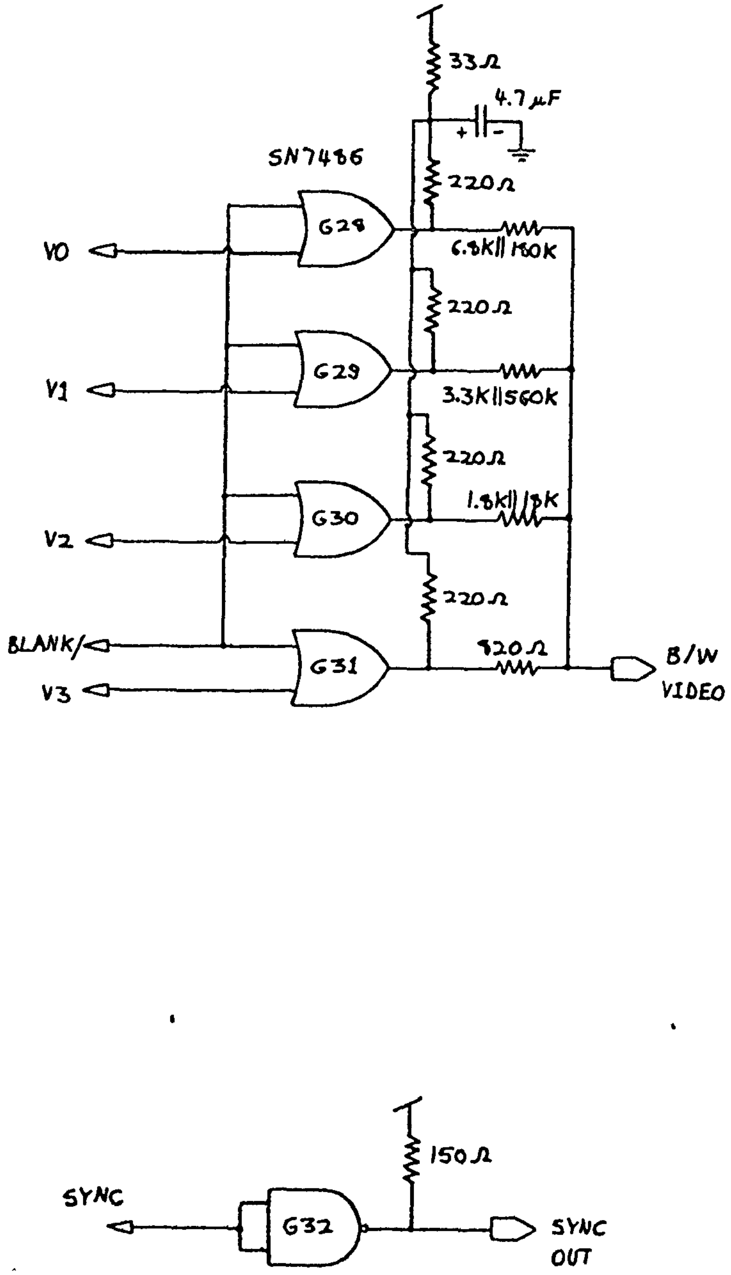


Fig. 2.17. Black and White D/A and Sync Driver

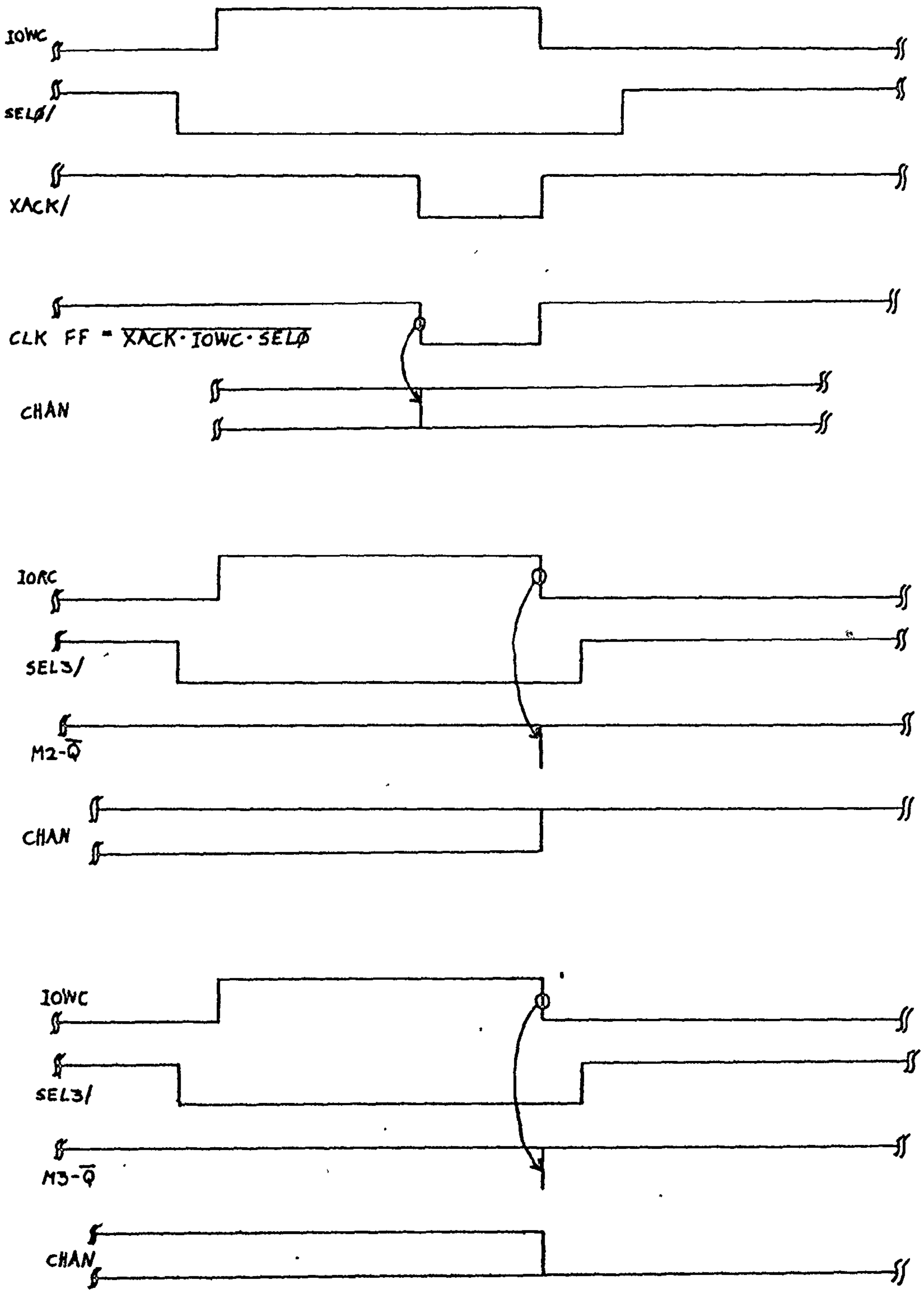


Fig. 2.18 I/O Port Timing and Channel Control

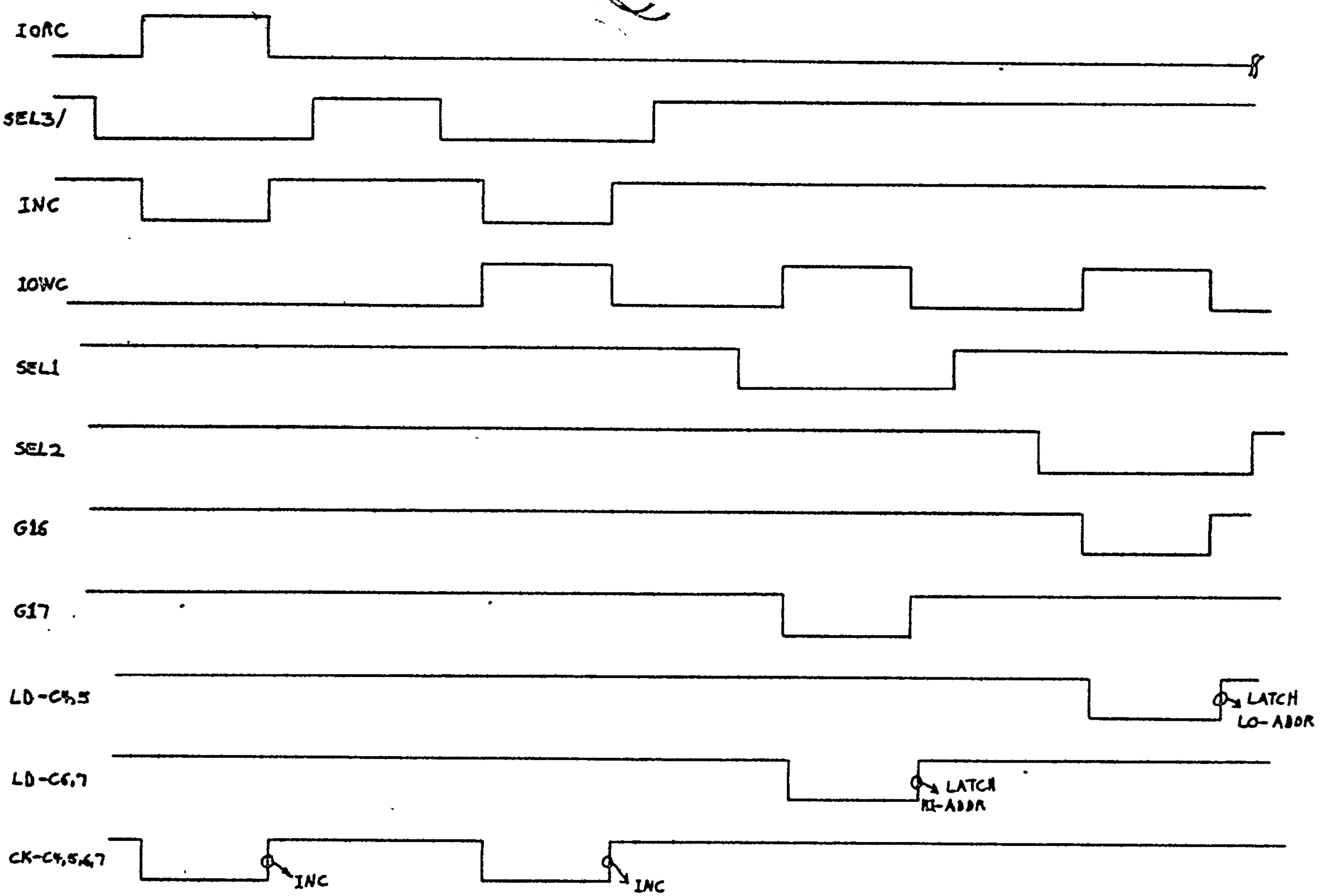


Fig. 2.19 Auto-Increment and Address Control Timing

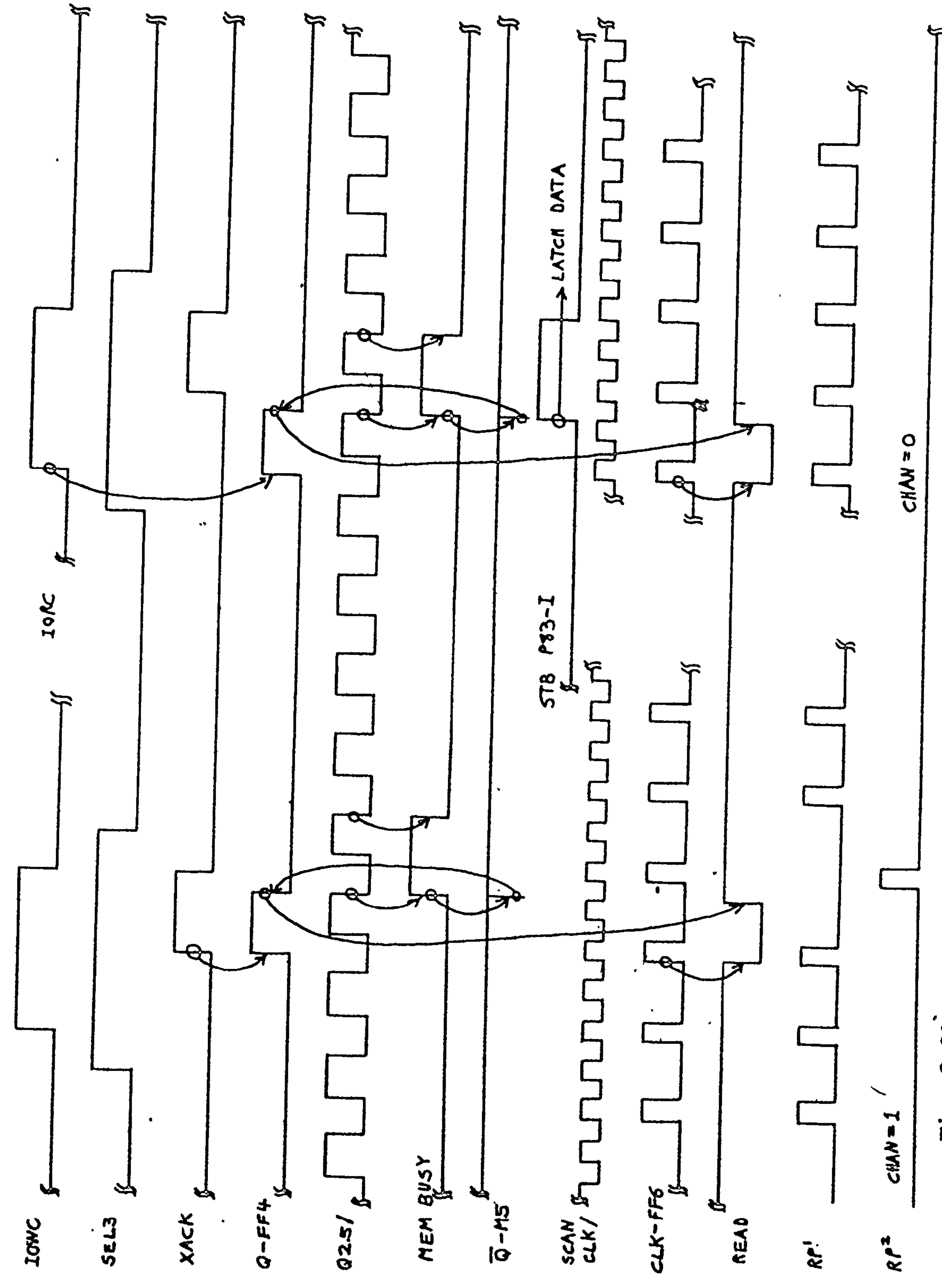


Fig. 2.20 Memory Cycle and Read/Write Timing

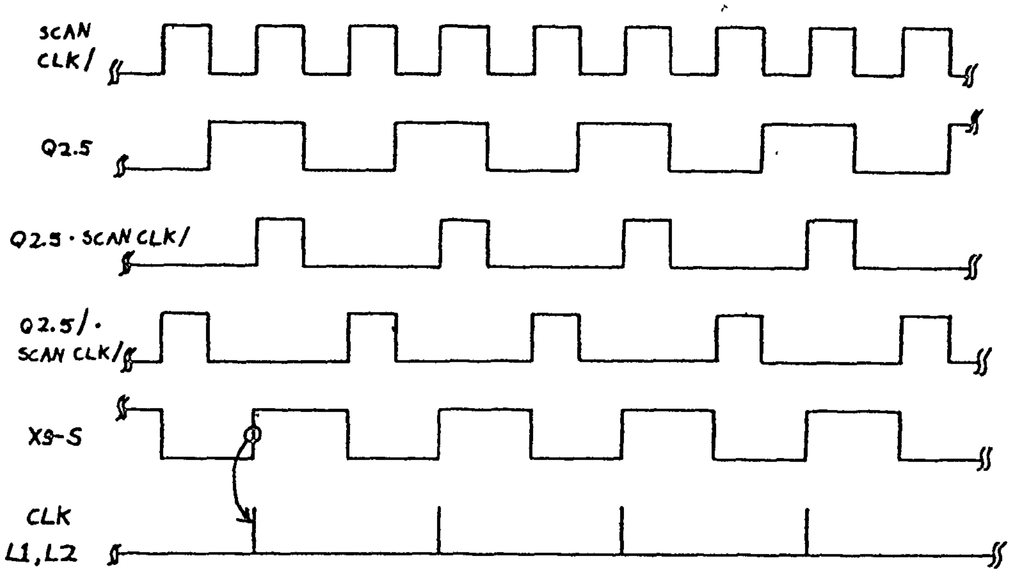


Fig. 2.21 Video Refresh Read Timing

constructed. The Intellec MDS computer has available a serial port for 2400 or 9600 baud, software selectable. Since the teletype was used as the terminal, this CRT port was available. The serial port employs an Intel 8251 UART/USART for serial/parallel conversion. This device is software programmable for various data formats and two baud rates. For the magnetic tape interface, the following format was chosen: 1 start bit, 8 data bits and 2 stop bits with a rate of 9600 baud. The input and output levels are TTL compatible.

The magnetic tape interface was designed to be compatible with any standard computer 9600 baud serial port since a NOVA 3 computer was used as well. The only requirement is that the data rate be crystal-controlled to ensure high accuracy.

2.4.2 Phase Encoding

The recording method chosen was phase encoding, or PE. This is a commonly used commercial technique. It achieves high data density, is self-clocking, reliable and insensitive to tape recorder speed variations. It is a phase-keyed type of carrier modulation, hence the name phase encoding. Fig. 2.24 shows a PE signal generated from the data and a clock. The phase encoded signal always has a transition in the centre of the bit "cell", which is used for the data clock recovery. The transition direction determines whether the data is a one or zero. Note that the PE signal can be generated by exclusive - oring the serial data and its clock, hence the phase modulation.

Experiment showed that the PE signal could be recorded on an audio tape recorder without modifications or equalization. A Sony model TC-640B open reel tape recorder was used, at a tape speed of 7-1/2

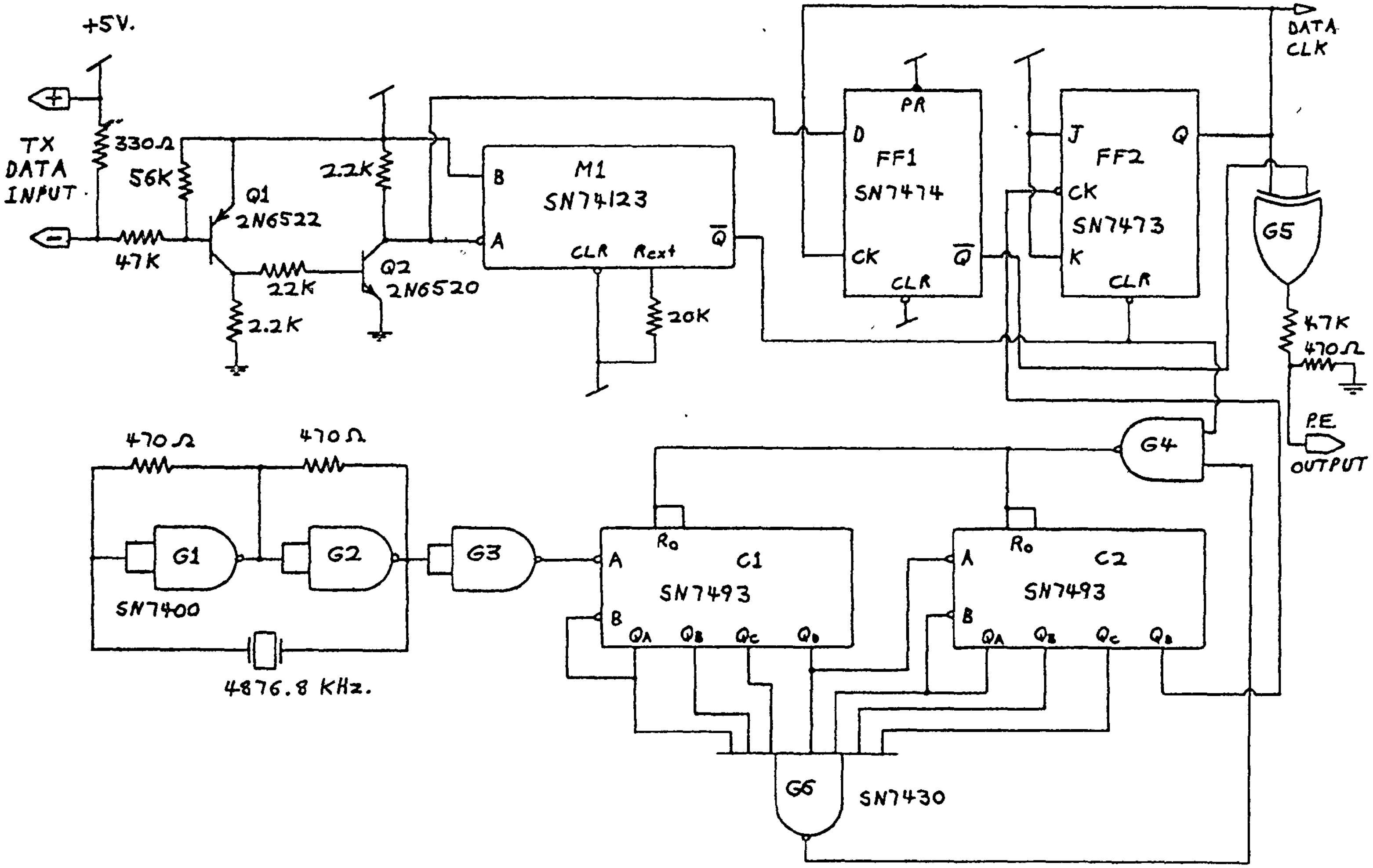


Fig. 2.22 Phase-Encoding Transmitter

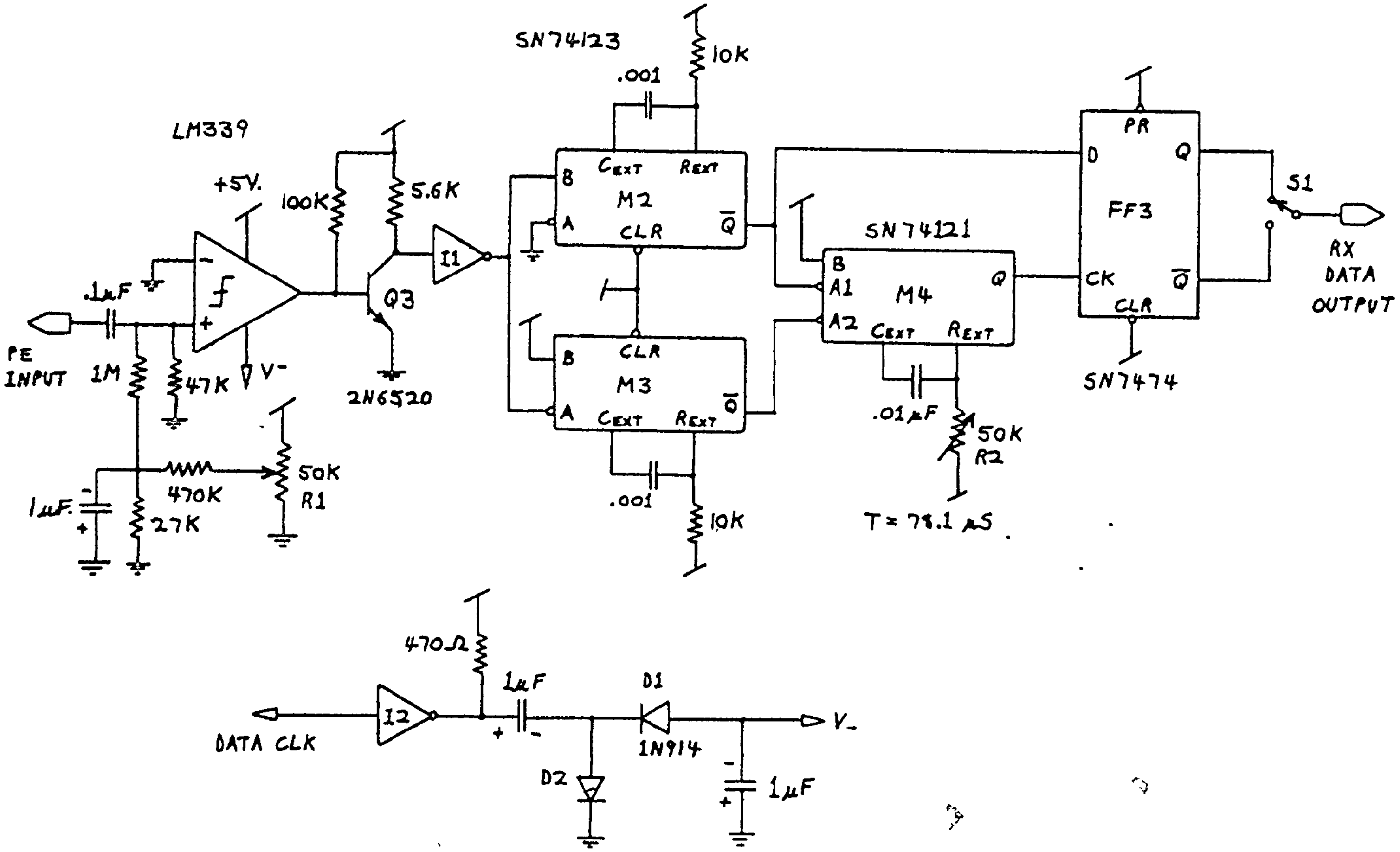


Fig. 2.23 Phase-Encoding Receiver

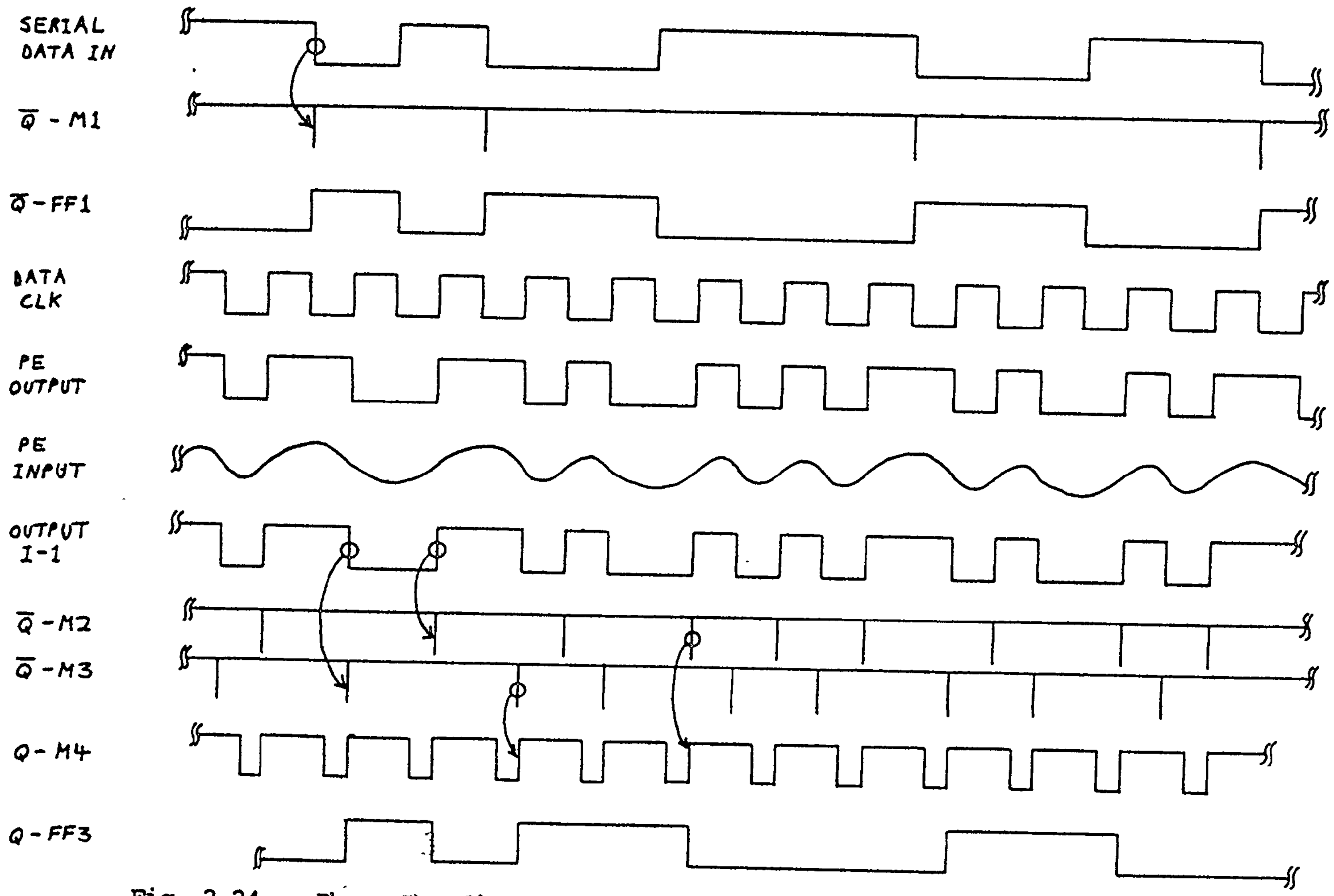


Fig. 2.24 Phase-Encoding Timing

inches per second. Although commercial tape drives use saturation recording, the normal audio channel was used since phase encoding lends itself to a bandlimited channel without DC response very well. The bandwidth requirements for a phase encoded signal is from a frequency of half the baud rate to twice the baud rate. In essence, PE is a wide band PSK signal, with the baud rate equal to the carrier (clock) frequency. At 9600 baud, the required frequency response is from 4,800 to 19,200 Hz. The amplitude and phase characteristics are not critical, as long as timing distortion of the zero crossings is not large. The record level used was at -10dB in order to allow a wide bandwidth, although the level is not critical. Since data is being recorded, high quality tape is necessary to minimize the error-rate.

2.4.3 Transmit Hardware

Fig. 2.22 shows the circuit for the transmit portion of the tape interface. The input data can be either a TTL signal or a 20mA loop. The TTL signal is applied to the - TX DATA INPUT. If a 20mA loop is used, connect the 2 data lines between - and + TX DATA INPUTS. Q1 and Q2 convert the input levels to TTL levels.

In order to guarantee baud rate stability, a crystal-controlled clock is used, generated by G1 and G2. C1 and C2 form a frequency divider with a division ratio of 254. G6 resets the counters just after the 254th count is received. A crystal available on hand with a divide by 254 counter produces, very nearly, a 19.2 KHz signal. It is divided by 2 using FF2 to produce a symmetrical 9600 Hz master data clock (or carrier).

The master clock, however, must be synchronized with the incoming data. Monostable M1 is triggered on each negative-going data transistion at the input. The resulting 50 ns pulse resets C1, C2 and FF2. If the master clock has a frequency very close to the baud rate of the data, the counters will be reset by G6 and M1 at almost the same time, producing little "jitter" in the master clock. The incoming data is clocked into FF1 so that the Q output (buffered data) has transitions at the same time as the clock itself. The clock and data are exclusive-ored to produce the PE signal which is connected to the input of the tape recorder.

2.4.4 Receive Hardware

As previously mentioned, only the zero-crossings in a PE signal are of importance. In Fig. 2.23, the LM339, a comparator, is a zero crossing detector. The output is buffered by Q3 and I1. R1 is adjusted to set the comparator switching point to ground potential. If R1 is properly adjusted, the comparator output will be hard-limited noise, for no input signal. For a zero crossing detector, the LM339 must have a negative power supply, although only a small voltage below ground is required. I2, D1, D2 form a negatitve power supply of 3V, using the data clock as the A.C. source. The LM339 only draws 1 MA of current.

As seen in Fig. 2.24, M1 and M2 act as a bi-directional monostable, which triggers M4 on either transistion of the comparator output. M4 is the clock-recovery monostable, being of non-retriggerable variety. If the output pulse width is adjusted for 3/4 of a baud interval (78.1 μs), M4 will trigger on each input zero-crossing that is in the centre of a bit cell, but will ignore all other zero-crossings since they will occur when M4 is in the triggered state. In order for M4 to successfully recover the clock, the data must not consist of continuous ones or zeros. This "randomness" condition is met since the data always has start (zero) and stop (one) bits in addition to actual data bits. FF3 latches the data, producing the receive data output signal, hopefully a replica of the original transmit data. S1 selects between inverting and non-inverting data. This option is necessary since some recorders are inverting and others are not, and data inputs may not have all the same polarity.

Although phase-encoding is ideally suited to synchronous data formats, an asynchronous format was used for greater simplicity and universality. This tape interface unit will also operate at baud rates which are sub-multiples of 9600 baud, such as 2400 baud, if desired, as long as the baud rate is accurate (less than .05 percent error). At 2400 baud, each bit will be repeated 4 times, although the output will appear as a normal 2400 baud signal.

CHAPTER 3

3.1 THE ITOS METEOROLOGICAL SATELLITE

3.1.1 Introduction

The Improved TIROS Operational Satellite (ITOS) formed the basis of an operational environmental satellite system operated by the National Oceanographic and Atmospheric Administration (NOAA). This series of 4 satellites, named NOAA-2 through NOAA-5 provided successful coverage and monitoring of the earth's meteorological conditions. With a designed lifetime of 2 years for each satellite, these satellites were launched in succession to provide continuous coverage from October 1972 through late 1978. As one satellite reached the end of its useful service life, the next one was launched. When the new satellite was proven to be functional and reliability established, the old deteriorating one was turned off.

The primary environmental instruments on these spacecraft were:

(i) A two channel Scanning Radiometer (SR) transmitting infrared and visible images of low resolution (7.5 Km), in real time, to small local stations.

(ii) A Vertical Temperature Profile Radiometer (VTPR), measuring energy from regions on the earth in 8 infrared channels, allowed vertical temperature profiles in the atmosphere to be computed.

(iii) A Solar Proton Monitor to measure proton and electron fluxes encountered during orbit.

(iv) A Very High Resolution Radiometer (VHRR) with 0.93 Km resolution. It provides continuous, real-time global coverage of the earth, as the SR does.

The ITOS satellites were launched in a 1464 km high orbit. Since it was in a sun-synchronous polar orbit, local coverage was obtained twice daily; in the daytime around 1000 local time, and at night at 2000 hours.

The last ITOS satellite, NOAA-5, provided images for the radar-satellite image comparison study. A new generation of meteorological satellites, TIROS-N, has superceded the ITOS satellites. In addition to improved sensors and instrumentation, for the first time, digital data and image transmission is used, in contrast to previous analogue techniques.

3.1.2 High Resolution Picture Transmission System

The ITOS spacecraft transmits, in real time, images from a scanning radiometer. The Very High Resolution Radiometer (VHRR) is a two channel instrument sensitive to energy in the visible spectrum 0.6 to 0.7 micrometers (green light) and infrared 10.5 to 12.5 μm. This "far infrared" is radiant energy from a body and hence allows measurement of temperature. With an infrared sensor, night-time images are possible.

The energy is gathered by an eliptical scan mirror and is reflected into a telescope optical (Cassegrainian) system. The energy is focused onto a mercury-cadmium-telluride detector sensitive to infrared. The optical system limits the field of view, with field stops, to 0.6 milliradians. In other words, the sensors have a "beam width" of 0.6 mr. This provides a spatial resolution of 0.9 km at the subpoint (directly downwards).

In order to form an image, the "sensor receiving beam" is scanned laterally by the rotating scan mirror at 400 rpm. The satellite motion then advances the scanning pattern, hence sampling the earth's radiation in two dimensions. This opto-mechanical scanning system is very widely used on many remote sensing and imaging satellite and aircraft systems. The image can be reconstructed and recorded on a CRT and film or many other opto-mechanical devices.

The images are transmitted to ground stations on an S-Band frequency of 1697.5 MHz, with a transmitter power of 5 watts and an EIRP of +37 dBm. The sensor baseband video frequency modulates a 99 KHz subcarrier which in turn frequency modulates the main RF carrier with a .1 MHz bandwidth. The FM subcarrier is useful since it can be recorded on a tape recorder at a ground station. The two channels are transmitted in a time multiplex format, with the video alternating between the two sensors for each scan line. By placing the scan mirrors of the two VHRR instruments 180 degress out of phase, the video from one sensor is time delayed with respect to the other.

At an altitude of 1464 km, the earth has an angular width of 108.8 degrees. An important problem in earth imaging systems is the perspective distortion, which increases as the scan deviates from the nadir and approaches the horizon, in this sytem 54.4 degrees from the nadir. The image appears to be compressed in the horizontal axis near either edge. What is most confusing to the human observer is the fact that the image aspect ratio is in considerable error near the edges since there is no perspective distortion in the vertical axis. As the scan deviates from the nadir or sub-point, the effective ground resolution decreases due to the increasing oblique angle with respect to the earth's surface. The ITOS VHRR system limits the maximum sensor viewing angle from the nadir to about ±35 degrees. Video outside this range is cut-off. Clearly, the earth scan video from each sensor occupies a fraction 70/360 or 0.19 of the total scan interval. Hence the two signals can easily time-share a scan with considerable time remaining for calibration, reference and telemetry signals. At sensor angles of ±30 degress, a sharp pulse is added to the video forming vertical white lines, called sub-sync markers, near each side of the images.

There are many ground stations receiving and processing VHRR images. VHRR data for the satellite-radar study was obtained from the Atmospheric Environment Service (AES) ground station in Toronto, Ontario. The ground station receives data for the local area as well as most of North America. The video signals are recorded on a wideband tape recorder and digitized for computer processing. The data is also stored on 9-track computer tape for later use or can be selectively duplicated for external users, as was the case for the satellite-radar study.

The video is quantized to 256 levels or 8-bits. The most important pre-processing task the computer performs is scan linearization. The scans are sampled at a varying rate such that the sample spacing represents a constant distance on the ground. Each earth scan has 2000 samples, with 1951 samples between the sub-sync markers. The data is written on standard 800 BPI (NRZI) 9-track tape according to the format in Table 3.1. Each scan line is stored as one record. The first record of a file is the header record. The remaining records are all data records. The file is terminated by an end-of-file mark. See Table 3.1 for the tape format.

Table 3.1

Full Resolution VHRR Data Tape Format

Contains one header record followed by one or more data records and is terminated with a file mark.

Header Record - 2000 bytes

Byte	Contents	
1,2	Orbit number, 16 bit integer (byte 1 is high-order byte)	
3,4	Equator crossing angle (in degress),	integer part, 16 bits
5,6		fractional part, 16 bits
7,8	Equator crossing time,	day of year, 16 bits
9,10		hours
11,12		minutes
13,14		seconds
15,16	Acquisition time,	day of year
17,18		hours
19,20		minutes
21,22		seconds
23,29	Zero	
30	Data tape type	0 = standard IR/VISUAL tape 1 = backup mode IR/VISUAL 2 - IR data only (2014 byte data records) 3 = VISUAL data only
31-39	Zero	
40	Tape number	0 = first tape (holds 4800 records) 1 = second tape
41-2000	Zero	

Data Record - 2014 or 4014 bytes

Byte	Contents
1,2	Record number since acquisition time, 16 bit unsigned integer (= 150 m Sec. increments from acquisition)
3-9	Voltage calibration wedge, 8 bit integers
10	Visible calibration, 8 bit integer
11	Radiance calibration
12-14	Spacecraft temperature, 3 8-bit integers
15-2014	IR pixel data, 8 bit unsigned integers 00000000 = cold 11111111 = hot
2015-4014	VISUAL pixel data, 8 bit unsigned integers 00000000 = black 11111111 = white

Notes: (1) bytes 2015-4014 not present if one channel tape
(2) Night-time tapes may contain a black visual channel

3.2 ORBIT MECHANICS

3.2.1 Orbits for Meteorological Satellites

There are two most common types of orbits for satellites; geosynchronous and sun-synchronous orbits. A geosynchronous orbit, used for all communications satellites, is also used for some meteorological satellites. A satellite in a geosynchronous orbit is fixed over the equator at some specified longitude with an altitude of 35,000 km. Satellites such as GOES continuously monitors weather activity at all times with a picture interval normally of 20 minutes. Although a geosynchronous satellite can view almost one half of the earth's surface, the viewing angle at high latitudes is oblique, thereby reducing image resolution.

The sun-synchronous orbit is the most commonly used orbit for low altitude satellites, particularly for remote sensing satellites. Such satellites provide coverage of any given area on the earth at the same local sun time each day. Hence the illumination angle is fixed, although in polar regions there is considerable seasonal variation in solar illumination. A morning orbit is generally used, providing coverage at 10:00 local time. A non-direct illumination angle is preferred since the shadows produced on cloud tops and land provide enhanced contrast.

The angle between the plane of the sun-synchronous orbit and the sun is fixed. In order to achieve this effect, the precession or rotation rate of the orbital plane with respect to the earth must be 360 degrees in 365.25 days (a solar year), in order to compensate for the earth's seasonal travel around the sun. The orbital plane is inclined with respect to the equator so that the variation in geopotential, due to equatorial bulge, will perturb the orbit and cause the desired precession.

If the sensor angle range is large enough, the entire earth's surface may be imaged twice daily. Any polar type orbit will pass over a land area approximately 12 hours apart, but travelling in opposite directions (ascending and descending passes). Fig. 3.1 shows the path of the satellite over one day (13 orbits). Fig. 3.2 and 3.3 show ascending and descending, respectively, coverage over North America for a 60 degree wide sensor swath. Notice that below 60 degrees latitude coverage is not obtained for that particular set of orbits. Fig. 3.4 shows the coverage swaths for the next day, "filling in" the uncovered area from the previous day. Higher resolution satellites have even narrower swaths. The low resolution sensor on NOAA has horizon-to-horizon coverage and guarantees complete coverage, even at the equator. The main problem with wide angle sensors is that considerable distortion near the swath edges is present. Ground processing (or

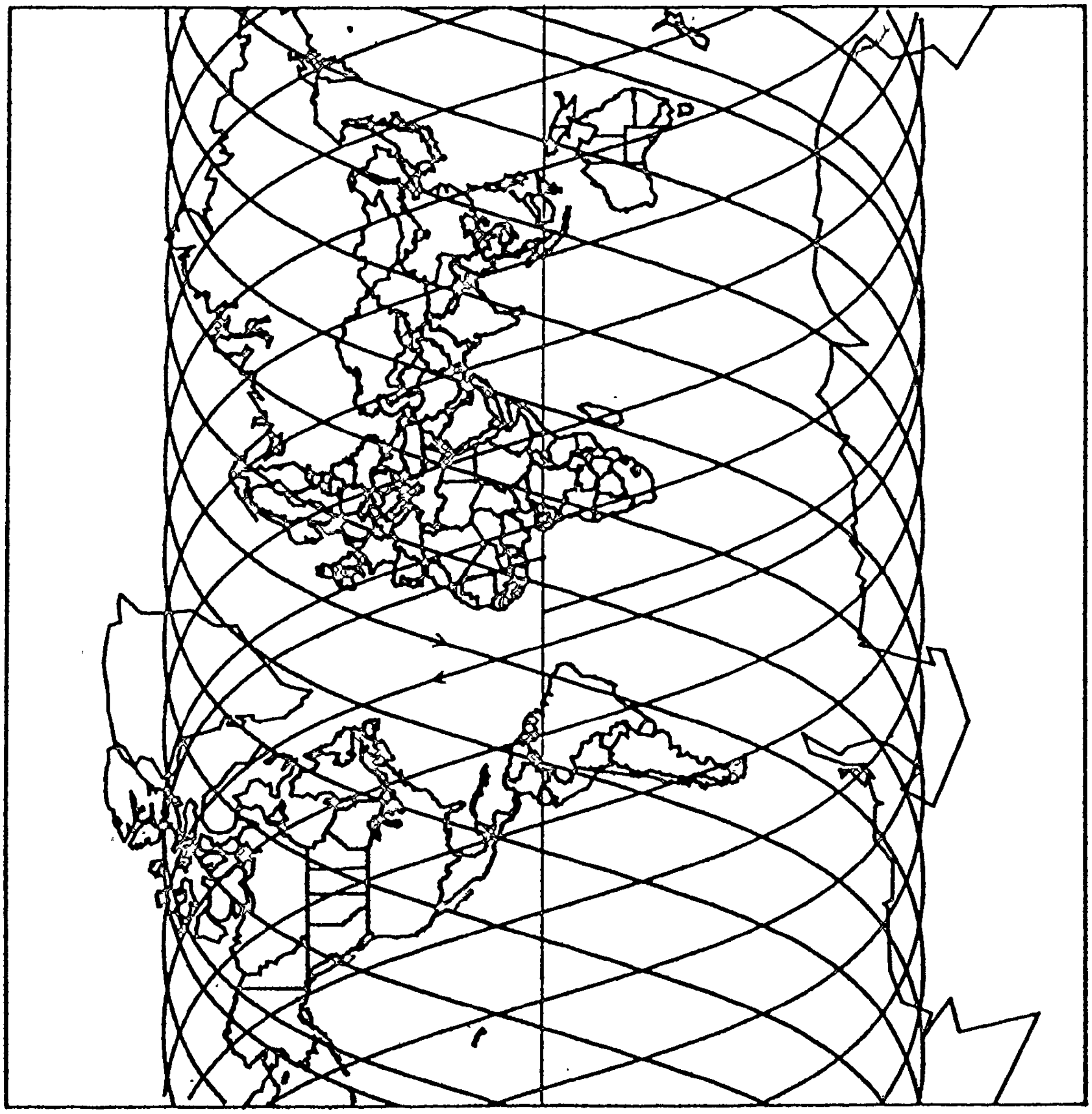


Fig. 3.1 One-Day Coverage (13 Orbits) of World

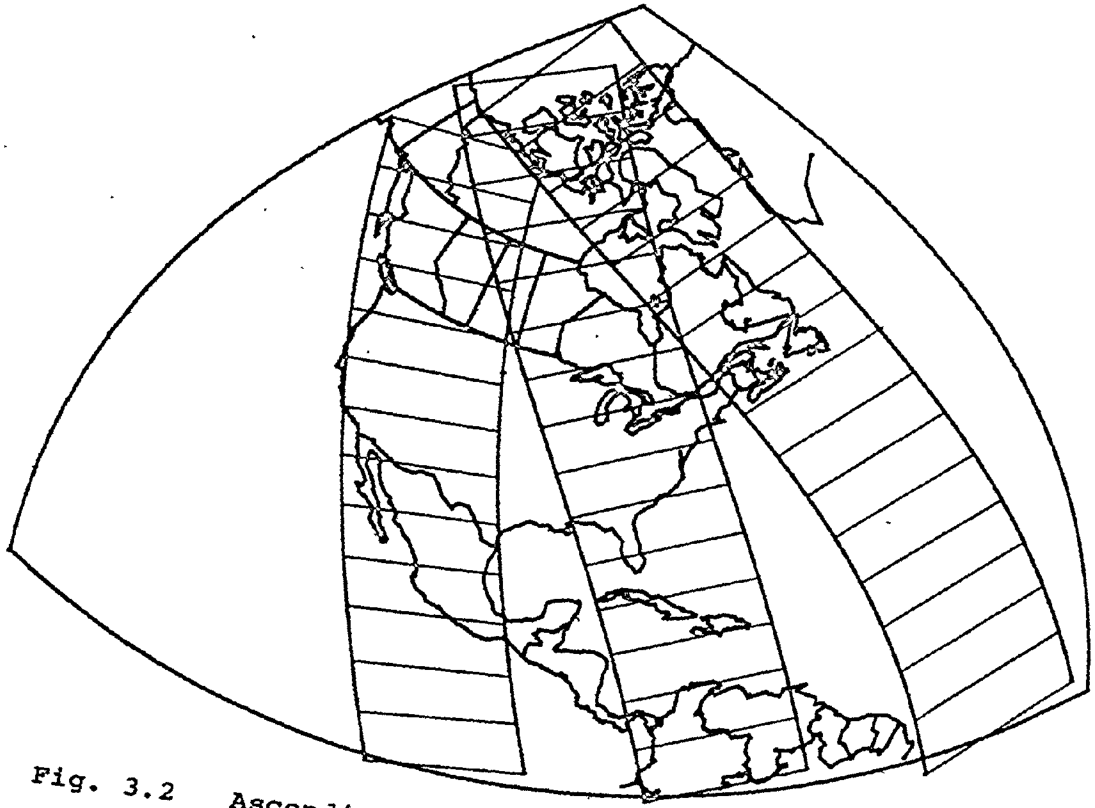


Fig. 3.2 Ascending Orbit Coverage (3 Orbits) of North America

Fig. 3.3 Descending Orbit Coverage (3 Orbits) of North America

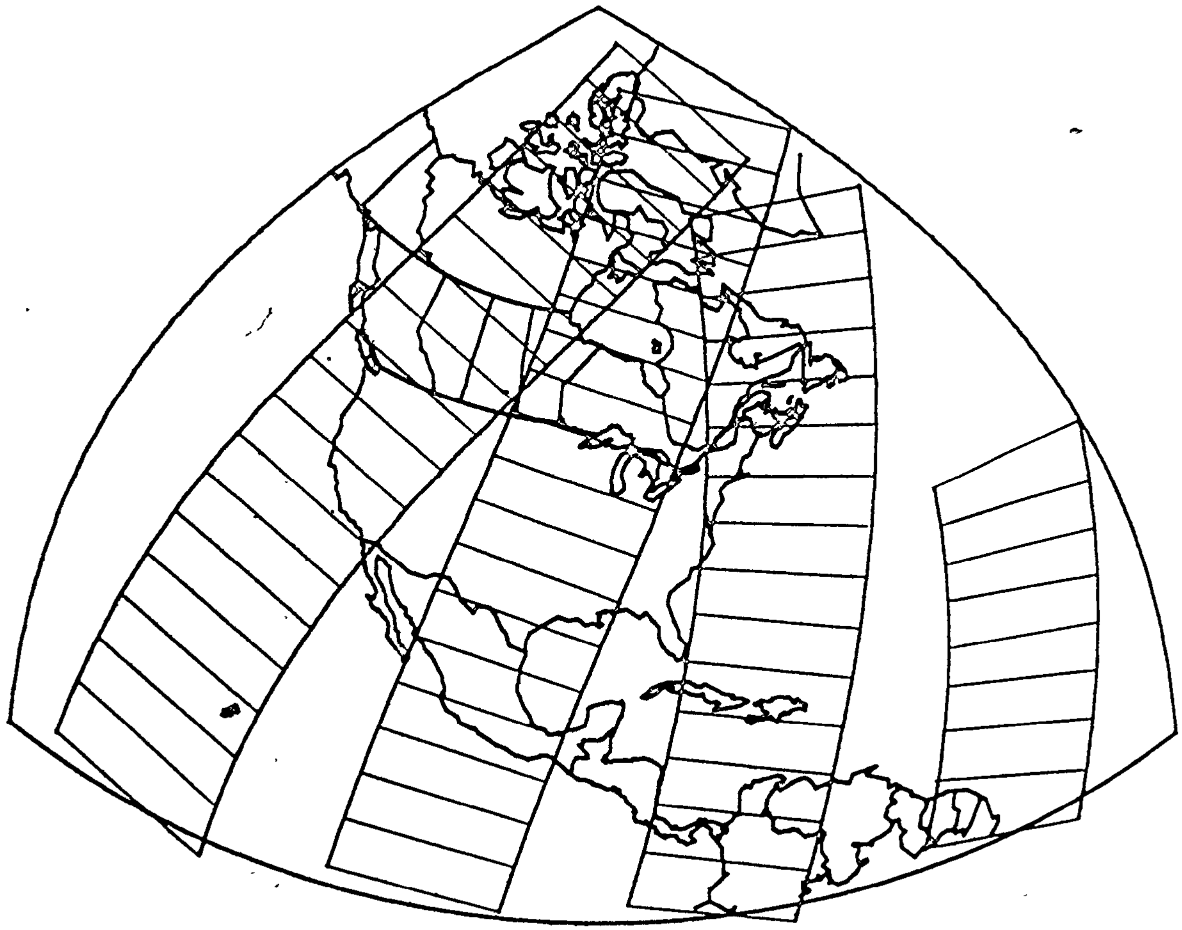


Fig. 3.4 Next-Day Descending Orbit Coverage (4 Orbits) of North America

satellite processing on the new TIROS-N series) compensate for the geometric distortion. However, at small incidence angles (from horizontal), problems still exist due to large variations in cloud height (up to 15 km) and reduced resolution.

In high resolution, narrow swath satellites, orbital period is an important parameter in determining coverage frequency. If a satellite had a period of 110.769 minutes in a sun-synchronous orbit, it would make exactly 13 orbits per day, covering the same ground areas each day. Orbital periods are selected so that the orbital tracks shift in longitude each day. With a nominal period of 115 minutes for NOAA satellites, a two-day earth coverage interval is achieved. Combining ascending and descending orbits reduce the coverage interval although visual images are not available at night.

3.2.2 Earth Co-ordinate Systems

The oblateness of the earth is sufficient that it must be considered in calculations of satellite position. The earth's surface is approximated by a spheroid which is symmetric about the polar axis. The distance from the centre of the earth to a point on the surface is only a function of latitude. The earth's latitude is defined in two ways on a reference spheriod. Geodetic latitude is used for maps and hence a latitude of a tracking or ground reference point read from a map is a geodetic latitude. As shown in Fig. 3.5, geodetic latitude is the angle between the equatorial plane and a plumb-bob or the local gravitational field direction. For satellite position determination, the geocentric latitude is used. Geocentric latitude is referenced from the earth geocenter or centre of gravity which is also the orbit centre

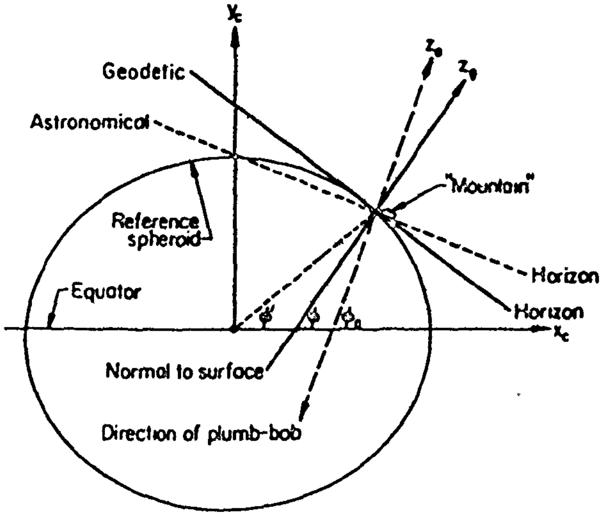


Fig. 3.5 Geocentric, Geodetic, and Astronomical Latitude

(focus of an orbital ellipse).

The earth's spheriod is modelled by an equation. The geodetic radius is [12]

$$R_{TGD} = b_E \left(1 + \frac{C}{2} + \frac{3}{8} C^2 + \frac{5}{16} C^3\right)$$

$$C = (\varepsilon \cos \psi_{TGD})^2$$

where b_E = polar radius (3949.914015 statute miles)

ε = earth eccentricity (.081813334)

ψ_{TGD} = geodetic latitude

The geocentric latitude is

$$\psi_{TGC} = \psi_{TGD} - \frac{X\, R_{TGD}}{R_{TGD} + h_T}$$

$$X = \frac{\varepsilon^2 \sin (2\, \psi_{TGD})}{2(1-\varepsilon^2 \sin^2 \psi_{TGD})}$$

where h_T = altitude above sea level
and ψ_{TGD} = geodetic latitude

3.2.3 Satellite Orbital Parameters [11]

The satellite's orbit describes an ellipse, with one focus at the earth's geocenter. In practice, the orbit achieved is usually very close to being circular, and a circular approximation can be made. Given that the satellite is in a circular retrograde sun-synchronous orbit and that the orbital period is known, all other parameters may be computed. The orbital period is normally specified by the anomalistic period. This is the time between successive passages through the perigee (nearest earth).

The orbital radius (semi-major axis if the orbit is elliptical)

$$R_A = \left(\frac{P_A}{C_1}\right)^{2/3}$$

where P_A = anomalistic period (hr.)

$C_I = 5.643912758 \text{ X}10^{-6}$

R_A in statute miles

Inclination

$$I = \cos^{-1} \dot{\Omega} \left(\frac{R_A}{R_E}\right)^{3.5}$$

where $\dot{\Omega}$ = orbital precession rate in rad./day.

For sun-synchronism, $\dot{\Omega} = -(24\ \omega_E - 2\pi)$

where ω_E = siderial earth rotation rate

$\dot{\Omega}$ is about -.986 degrees per day.

$$C = \frac{C_2}{(1-e^2)^2\ R_A^{7/2}}$$

where e = eccentricity

Orbital rotation rate (rad/hr.)

$$\omega_s = \omega_E + C \cos I$$

Rotation rate of perigee

$$\omega p = \frac{C}{2} (5 \cos^2 I - 1)$$

Nodal period (time between successive ascending nodes)

$$P_N = P_A \left(1 - \frac{\omega_p}{\pi}\right)$$

If the orbit is elliptical

Eccentric anomally (angle between centre of ellipse and perigee)

$$E = \cos^{-1} \left[\frac{e + \cos \omega_p}{I + e \cdot \cos \omega_p}\right]$$

Time from perigee to ascending node

$$T_p = \frac{(E - e \sin E)\, P_N}{2\pi}$$

3.2.4 Satellite Ground Track

The position of a satellite is normally referred to the earth's co-ordinate system of latitude and longitude. The intersection of the line from the earth geocenter to the satellite on the earth's surface is the subsatellite point. It is defined in terms of geocentric latitude and longitude.

Subsatellite latitude $\psi_S = \sin^{-1} (\sin \phi \sin I)$

$\phi = (\tau + P_N) / P_N$

where τ is time after ascending node.

Subsatellite longitude

$\lambda_s = \Omega + \omega_s \tau + \Delta\lambda$

$\Delta\lambda = \sin^{-1} (\tan \psi_s / |\tan I|)$

Ω = ascending node longitude

$\omega_s \tau$ is effect due to the rotating earth.

3.2.5 Satellite Sky Position

The satellite sky position for a local ground station is defined by elevation angle above the horizon and the azimuth or bearing from north. This is computed using basic navigational equations. Angular distance (Great Circle arc) between ground station and satellite

$$SEP = \cos^{-1}(\sin \psi_s \sin \psi_T + \cos \psi_s \cos \psi_T \cos (\lambda_T - \lambda_s))$$

where λ_T and λ_T are ground station (geocentric) latitude and longitude.

Satellite elevation

$$EL = \tan^{-1}\left(\frac{R_s \cos SEP - R_{TGC}}{R_s \sin SEP} \right)$$

Satellite azimuth

$$AZ = \tan^{-1}\left(\frac{\tan \psi_s}{\sin(\lambda_T-\lambda_s)} - \frac{\tan \psi_T}{\tan(\lambda_T-\lambda_s)} \cos \psi\right)$$

3.2.6 Ascending Node Prediction

Given an initial reference ascending node longitude and time, ascending nodes in the future (or past) may be predicted.

Ascending node longitude

$$\Omega_F = \Omega_i + \omega_S \cdot P_N \cdot REVS$$

Ascending node time

$$T_\Omega = T_i + P_N \cdot REVS$$

where Ω_i and T_i are initial (reference) ascending node longitude and time respectively and REVS is number of orbital revolutions after reference ascending node.

3.2.7 Scan Line Orientation

Since the orbit is inclined, (not a direct polar orbit) the image scan lines will be tilted with respect to a line of longitude. The azimuth of a scan line relative to a longitude line.

$$AZ = \tan^{-1}\left[\frac{\sqrt{\cos^2\psi - \cos^2 I}}{C.\cos^2\psi_s - \cos I}\right]$$

where $C = \frac{P_N \cdot \omega_s}{2\pi}$

When the satellite is at maximum latitude ($\psi_s = 2\pi - I$), the scanning is directly along longitude lines. Hence the image orientation changes considerably throughout the orbit.

3.2.8 Sensor Angle

The satellite sensor angle (from nadir)

$$A_s = \tan^{-1}\left[\frac{R_{TGC1} \cdot \sin(SEP)}{R_S - R_{TGC1} \cdot \cos(SEP)}\right]$$

where R_{TGC1} = earth radius at latitude ψ_s.

SEP = earth arc between sub-satellite point and image point along scan.

This function is also used by ground stations to correct image distortion along scan to remove foreshortening.

3.2.9 Computation of Ground Coordinates from Image Point

An image point is usually specified as a column (element on a line) and a line number. Knowing the scanning rate and the sampling rate (number of columns per unit of sensor angle), time after ascending node and sensor angle is directly determined. Then the sub-satellite point, ψ_s, λ_s, and SEP and AZ_s are calculated.

$$SEP = \sin\left[\frac{R_s \sin\theta_s}{R_{TGC1}} - \theta_s\right]$$

By spherical trigonometry, the image point co-ordinates ψ_i, λ_i are calculated.

$$\alpha = (\frac{\pi}{2} - \psi_s - SEP)/2$$

$$\beta = (\frac{\pi}{2} - \Psi_s + SEP)/2$$

$$\gamma = \tan^{-1}\left[\frac{\cos\alpha}{\cos\beta\tan(\frac{AZ_s}{2})}\right]$$

$$\alpha = \tan^{-1}\left[\frac{\sin\alpha}{\sin\beta\tan(\frac{AZ_s}{2})}\right]$$

$$\Psi = \frac{\pi}{2} - 2\tan^{-1}\left[\frac{\tan\alpha \cdot \sin\gamma}{\sin\alpha}\right]$$

$$\lambda_1 = \lambda_2 + \delta - \gamma$$

This procedure may also be used to compute the co-ordinates of a radar target, given its range and azimuth. Figs. 3.6 and 3.7 illustrate the geometry involved.

3.3 COMPUTER PROGRAM FOR ORBIT CALCULATIONS

3.3.1 Introduction

A FORTRAN computer program, listed in Appendix 6, was developed to compute daily time and location predictions for local coverage. For each day, the equator crossing time and ascending node longitude is predicted. The time the satellite scans across a given ground point is calculated, in addition to the image position (line and column number) of that ground point. The program compares the map projections of the radar and satellite image with a common ground centre point. A normal radar Plan Position Indicator (PPI) display has a Lambert's Azimuthal projection. It was found that the satellite image with scan linearization well-approximated a Lambert's projection, for at least a

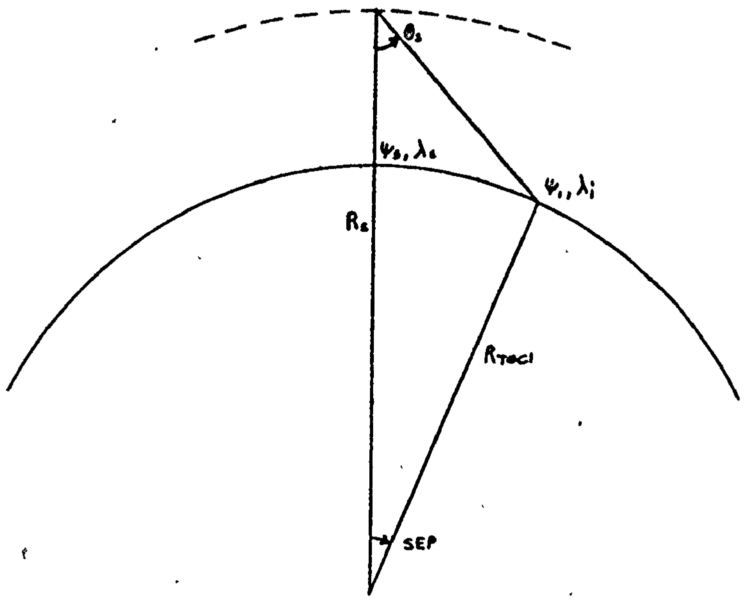


Fig. 3.6 Satellite Sensor Geometry

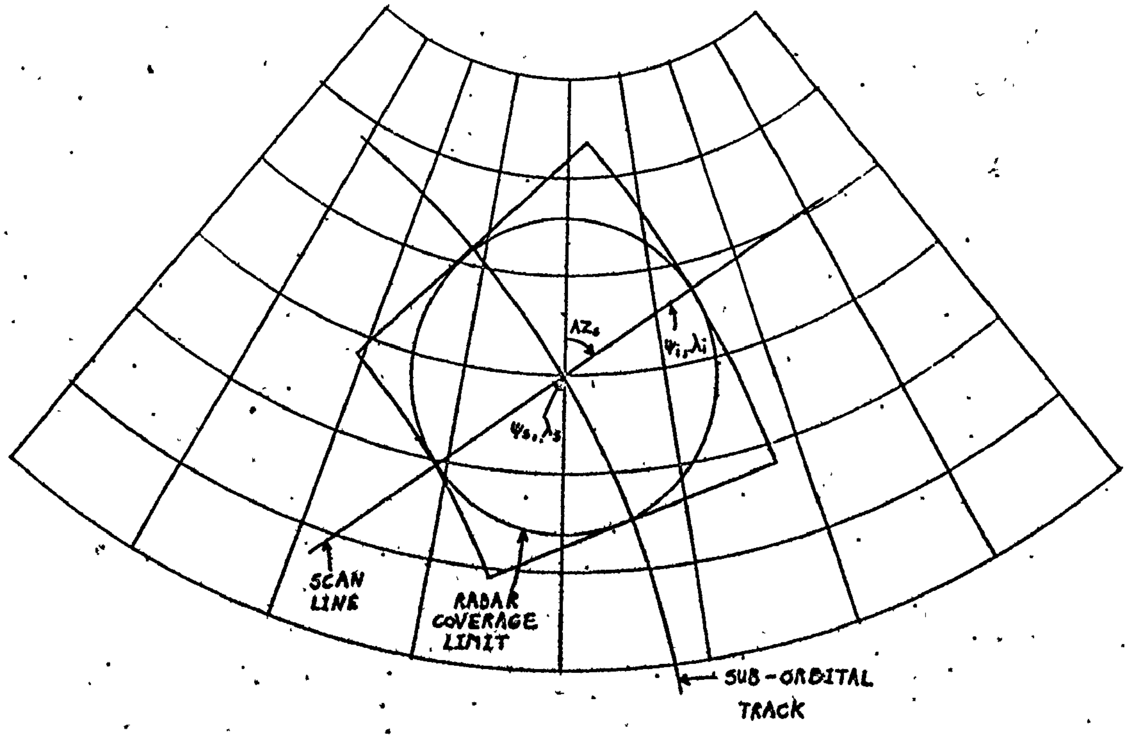


Fig. 3.7 Satellite and Radar Image Geometry

150 nautical mile radius about the centre point. For matching the satellite to the radar image, only horizontal and vertical scaling and translation and rotation is required. Hence, no non-linear image "warping" required.

3.3.2 Input Parameters and Output

The program may be run either on a batch or terminal mode. The end of Appendix 6 is a listing of the data cards used. In interactive terminal mode, the data is entered via the terminal, prompted by a question-mark. The response to a question-mark must satisfy the requirements of the prompting read statement. The input is in free-format, where the numbers are separated by a comma or space. A decimal point is not required for a real number. The parameters are input in the following order:

(i) BE, OBLECC, EROT
(ii) PANOM, ECC, INC, ARGPER
(iii) OMEGA, TOMEGA, D, M, Y, ORNUMI
(iv) PSITRC, LAMTRC, SLHTRC
(v) LAMMAP, PSIMAP
(vi) SHAPE
(vii) OMEGAD1, OMEGAD2
(viii) OMEGAN1, OMEGAN2
(ix) JOB, MODE
(x) D, M, Y
(xi) D, M, Y
(xii) KKK

See Table 3.2 for definitions of variable names. Record (x) and (xi) are the prediction starting and ending dates respectively. Records (x), (xi) and (xii) are repeated JOB times. After record (x) is read, the heading is printed, containing the month and year.

A sample run is shown in Tables 3.3 to 3.8. The following prediction information is printed out:

DATE - date of month and day of week
ASCENDING TIME- hours, minutes, seconds of ascending node for DATE
NODE LONG - ascending node longitude in degrees west
MAP LINE TIME - hours, minutes, seconds satellite scans map point (radar centre)
MAX ANG - sensor angle in degrees from nadir of west radar coverage limit
CENTRE - sensor angle of radar centre
MIN ANG - sensor angle of east radar coverage limit
ORBIT - orbit number
LINE NO - line number relative to ascending node of map line time
COLUMN - number of picture elements from line start to CENTRE

Five tables are printed out, containing image co-ordinate information. The satellite and radar images are assigned LP1 horizontal grid points and HP1 vertical grid points. The radar centre co-ordinates are LAMMAP, PSIMAP and a maximum range of RANGE. The satellite image grid is centered over LAMMAP, PSIMAP. The time TAUO is computed where a scan line crosses the centre ground point. This must be done iteratively, using Newton's method. An initial time TAU after ascending node is chosen, GUESS. The sub-satellite point ψ_s, λ_s and then scan azimuth AZ_s is computed. Given AZ_s and the ground point longitude λ_i, the ground point latitude is computed as follows:

$$\Delta\lambda = \lambda_s - \lambda_i$$

$$z = \tan\left[\frac{\frac{\pi}{2} - \psi_s}{2}\right]$$

$$\sigma = (\Delta\lambda - AZ_s)/2$$

$$\zeta_T = \frac{\pi}{2} - \tan^{-1}\left(z \cdot \frac{\cos\sigma}{\cos\zeta}\right) + \tan^{-1}\left(z \cdot \frac{\sin\sigma}{\sin\zeta}\right)$$

If the time TAU = TAUO, then $\psi_T = \psi_i$. Newton's method is used to converge ψ_T to ψ_i such that $|\psi_T - \psi_i| < \epsilon_{MAX}$, where ϵ_{MAX} is the maximum allowable error.

The sub-satellite track is divided into HP1 points, equally spaced in time (and line numbers) so that the total length is 2*RANGE. This sets the vertical scaling such that the vertical satellite image scale is the same as the radar image scale. The scan line at time TAUO is similarly divided into LP1 points, equally spaced in ground distance (and column numbers) so that the total length is again 2*RANGE. Horizontal centering is achieved by computing the distance along the scan line (at time TAUO) from the subpoint to the ground centre point ψ_i, λ_i. Over a small region of the earth, the radar and satellite images have the same geometry except for a rotation factor due to the satellite inclination.

Table 3.4 is a grid of satellite image latitudes and longitudes. The longitudes along the centre column are not constant due to image rotation from true north. Table 3.5 is a grid of radar image latitudes and longitudes. Table 3.6 is the difference between the two previous grid co-ordinates, which have a common centre point. Table 3.7 is the satellite grid points expressed as azimuth and range relative to the centre point. Table 3.8 is an array of azimuth and range correction factors such that when applied to the radar image, it will exactly match the satellite image. The azimuth correction is a rotation and the range correction is a multiplier to the range. All the azimuth and range correction factors are constant within at least two significant digits. For an image of 256 square points, the two images will be aligned with a maximum error .5 of a picture element distance if the rotation corrections are less than .16 degree about the mean and if the range correction factors are within .0028 of unity. Although the maximum error in table 3.8 is greater, the image alignment will be satisfactory if only a constant mean rotation is applied.

Table 3.2

Main Program Variable Definitions

A	Vertical distance from grid centre to grid point in grid units
AANOM	Semi-major axis of anomalous orbit
ARGPER	Argument of perigee
AZ	Azimuth of satellite from ground image centre point
AZAR	Array of satellite grid point azimuths
AZSCAN	Satellite scan azimuth at a row of grid points
AZSTR	Satellite scan azimuth at TAUO (centre of image)
AZ1	Azimuth of satellite at time TAUO + 1 sec.
B	Horizontal distance from grid centre to grid point in grid units
BE	Polar radius of earth (statute mi.) (3949.914015)
C	Fraction of earth's circumference travelled in one orbit
COLUMN	Element number to image centre from scan start
CORA	Azimuth correction array
CORR	Range correction array
COSINC	Cosine of inclination
COSMAP	Cosine of PSIMAP
COSPSI	Cosine of PSISAT
D	Day of reference orbit
DAY	Hollerith (character) array of day names
DELTA	Earth surface arc (rad.) of RANGE
DELTAT	Time (hr.) of satellite travel between grid points
DIFLAT	Array of latitude differences between satellite and radar grids
DIFLON	Array of longitude differences between satellite and radar grids
DLONG	Longitude difference between LAMSAT and grid point
DOMEGA	Longitude increment between ascending nodes
DOW	Day of Week (Sunday = 1)
ECC	Orbital eccentricity (.00101 for NOAA 5)
EL	Elevation of satellite from ground image centre point
EL1	Elevation of satellite at time TAUO
EROT	Earth rotation rate (rad./hr.) (.2625161708)
GUESS	Initial guess of TAUML (hr.)
H	Number of vertical points - 1 in co-ordinate grid

HP1	H+1
INC	Orbital inclination (1.781547709 for NOAA5)
JDNE	Julian day number of epoch orbit
JDNREF	Julian day number of reference orbit
JDN1	Julian day number of first predicted orbit
JDN2	Julian day number of last predicted orbit
JOB	Number of program runs for different sets of predictions
JOBI	Loop index for JOB
K	Current Julian day number
KKK	1 for day, 0 for night pass in each prediction set
L	Number of horizontal points - 1 in co-ordinate grid
LAMMAP	Geodetic longitude of image centre
LAMSAT	Longitude of satellite subpoint
LAMTRC	Geodetic longitude of tracking station (rad.)
LAT	Satellite grid point latitude
LINE	line number at TAU0
LNRATE	Image line per minute
LONG	Satellite grid point longitude
LOOP	Loop index for rows
LP1	L+1
LSCAN	Loop index for columns
M	Month of reference orbit
MH	Hours of TML
MM	Minutes of TML
MODE	0 for prediction times only
MONTH	Hollerith (character) array of month names
MS	Seconds of TML
NH	Ascending node hours
NM	Ascending node minutes
NS	Ascending node seconds
OBLECC	Eccentricity of earth (.081813334)
OMEGA	Longitude of ascending node
OMEGAD1	East daytime ascending node limit for radar area coverage (rad.)
OMEGAD2	West daytime ascending node limit
OMEGAN1	East nightime ascending node limit
OMEGAN2	West nightime ascending node limit

OMEGA1	East ascending node limit in each prediction set
OMEGA2	West ascending node limit in each prediction set
ORNUM	Orbit number of reference orbit
PANOM	Anomalistic period (1.938111545 for NOAA 50
PI	π
PNODAL	Orbital nodal period (hr.)
PROT	Rotation rate of perigee (rad./hr.)
PSIMAP	Geodetic latitude of image centre
PSISAT	Latitude of satellite subpoint
PSITRC	Geodetic latitude of tracking station (rad.)
PSI1	Latitude of satellite subpoint at time TAUO + 1 sec.
RADAZ	Azimuth of a radar grid point
RADLAT	Array of latitudes of radar grid points
RADLON	Array of longitudes of radar grid points
RADSEP	Range (earth arc) to a radar grid point from centre
RANGE	Radius (nmi.) to radar image edge
RAR	Array of RSEP valaues of radar image grid points
REVS	Integer part no. of orbits since epoch orbit
RSAT	Satellite orbital radius
RSEP	Earth arc from centre
RTGC	Geocentric radius of tracking station (statue mi.)
RTGC1	Geocentric radius of image centre (statue mi.)
RTOD	Radians to degrees conversion factor
SATLAT	Array of latitudes of satellite, image grid points
SATLON	Array of longitudes of satellite, image grid points
SENSA	Sensor angle from nadir to image centre
SENSB	Sensor angle from nadir to right edge of image (RANGE)
SENSC	Sensor angle from nadir to left edge of image
SENSP	Sensor angle from nadir to grid point in grid units
SEP	Earth angular arc from ground image centre to satellite subpoint
SEPGRD	Sensor angle between grid points (horizontal)
SEPMAX	Earth arc from nadir to sub-sync marker
SEPPIX	Sensor angle between each image element
SEPSCN	Earth arc from nadir to grid point
SEP1	SET at time TAUO + 1 sec.
SHAPE	1 for elliptical orbit, 0 for circular approximation

SHIFT	Picture elements from line start to 1st sub-sync marker
SINMAP	sin of PSIMAP
SIZEH	Number of satellite image horizontal elements for distance 2*RANGE
SIZEV	Number of satellite image lines for distance 2*RANGE
SLHTRC	Altitude of tracking station (statute mi.)
SROT	Orbital rotation rate (rad./hr.)
TAUPER	Time from perigee to ascending node (hr.)
TAUO	TAUML
TCOV	Time (hr.) required to image distance RANGE
TML	Time of day (hr.) of TAUML
TMLI	
TOMEGA	Time of ascending node (equator crossing) (hr.) for each predicted orbit
TOSAV	Time of ascending node
WIDTH	Picture elements between sub-sync markers
X	Current ascending node in predictions
Y	Year of reference orbit
Y1	Satellite longitude at time TAUO + T sec.

3.4. DATA FORMATTING FOR IMAGE DISPLAY

3.4.1 9-Track to Phase-Encoded Format

The NOAA satellite raw image data as supplied on tape is not in a suitable format for direct display since only a section of the image data is to be processed and displayed. At most, a 300 nmi. square area would be required, corresponding to radar with a range of 150 nmi. Since the satellite data is supplied on 9-track 800-BPI magnetic tape, the data had to be copied to a tape compatible with the microcomputer system which is described in section 2.4. A NOVA-3 minicomputer with a 9-track tape drive was used to transfer the data to the secondary serial port wired for 9600 baud. Appendix 7 is the assembly language program.

A large word count of 32767 and the buffer start address is sent to the tape drive. The record is read (terminated by an end of record

```
            ? 3949.914015 .081813334 .2625161708
            ? 1.938111545 .00101 1.781547709 0.
            ? -2.2019073 7.2333333 20 4 1978 7796
            ? .8 1.407171709 .0615530303
            ? 1.6970417 0.8709192
            ? 0
            ? -1.6256327 -1.1180248
            ? 1.1162795 1.6238873
            ? 1 1
            ? 6 5 1978


                          MAY 1978

              DATE  ASCENDING NODE   MAP LINE
                     TIME     LONG     TIME

            ? 6 5 1978
            ? 2
             6SAT  20 42 58   76.02  20 59 43




MAX ANG   CENTRE    MIN ANG       ORBIT LINE NO   COLUMN


15.9422    6.0806   -4.3071       8001  6703.75  1177.60
```

Table 3.3 Sample Run of Program

50.91	51.05	51.19	51.33	51.47	51.60	51.74	51.87	52.00
102.46	102.01	101.57	101.12	100.67	100.22	99.77	99.31	98.85
50.63	50.77	50.91	51.05	51.19	51.32	51.45	51.59	51.72
102.23	101.79	101.35	100.90	100.45	100.00	99.55	99.10	98.64
50.35	50.49	50.63	50.77	50.90	51.04	51.17	51.30	51.43
102.00	101.57	101.13	100.68	100.24	99.79	99.34	98.89	98.43
50.07	50.21	50.35	50.49	50.62	50.75	50.89	51.02	51.14
101.78	101.35	100.91	100.47	100.02	99.58	99.13	98.68	98.22
49.79	49.93	50.07	50.20	50.34	50.47	50.60	50.73	50.86
101.56	101.13	100.69	100.25	99.81	99.37	98.92	98.47	98.02
49.51	49.65	49.78	49.92	50.05	50.18	50.32	50.44	50.57
101.34	100.91	100.48	100.04	99.60	99.16	98.72	98.27	97.82
49.23	49.36	49.50	49.64	49.77	49.90	50.03	50.16	50.28
101.13	100.70	100.27	99.83	99.40	98.96	98.51	98.07	97.62
48.94	49.08	49.22	49.35	49.48	49.61	49.74	49.87	50.00
100.92	100.49	100.06	99.63	99.19	98.75	98.31	97.87	97.43
48.66	48.80	48.93	49.07	49.20	49.33	49.46	49.58	49.71
100.71	100.28	99.85	99.42	98.99	98.55	98.12	97.68	97.23
48.38	48.52	48.65	48.78	48.91	49.04	49.17	49.30	49.42
100.50	100.07	99.65	99.22	98.79	98.35	97.92	97.48	97.04
48.10	48.23	48.37	48.50	48.63	48.76	48.88	49.01	49.13
100.29	99.87	99.45	99.02	98.59	98.16	97.72	97.29	96.85
47.81	47.95	48.08	48.21	48.34	48.47	48.60	48.72	48.85
100.09	99.67	99.25	98.82	98.39	97.96	97.53	97.10	96.66
47.53	47.66	47.80	47.93	48.06	48.18	48.31	48.44	48.56
99.89	99.47	99.05	98.62	98.20	97.77	97.34	96.91	96.48
47.25	47.38	47.51	47.64	47.77	47.90	48.02	48.15	48.27
99.69	99.27	98.85	98.43	98.01	97.58	97.15	96.72	96.29
46.96	47.10	47.23	47.36	47.48	47.61	47.74	47.86	47.98
99.49	99.07	98.66	98.24	97.82	97.39	96.97	96.54	96.11
46.68	46.81	46.94	47.07	47.20	47.32	47.45	47.57	47.69
99.29	98.88	98.46	98.05	97.63	97.21	96.78	96.36	95.93
46.39	46.52	46.65	46.78	46.91	47.04	47.16	47.28	47.40
99.10	98.69	98.27	97.86	97.44	97.02	96.60	96.18	95.75

Table 3.4 Satellite Image Co-ordinate Grid

52.13	52.26	52.38	52.50	52.63	52.74	52.86	52.98
98.38	97.92	97.45	96.98	96.50	96.03	95.55	95.07
51.84	51.97	52.09	52.22	52.34	52.46	52.57	52.69
98.18	97.71	97.25	96.78	96.31	95.83	95.36	94.88
51.56	51.68	51.81	51.93	52.05	52.17	52.28	52.40
97.97	97.51	97.05	96.58	96.11	95.64	95.17	94.69
51.27	51.40	51.52	51.64	51.76	51.88	51.99	52.11
97.77	97.31	96.85	96.39	95.92	95.45	94.98	94.51
50.98	51.11	51.23	51.35	51.47	51.59	51.70	51.82
97.57	97.11	96.65	96.19	95.73	95.26	94.80	94.32
50.70	50.82	50.94	51.06	51.18	51.30	51.41	51.52
97.37	96.92	96.46	96.00	95.54	95.08	94.61	94.14
50.41	50.53	50.65	50.77	50.89	51.01	51.12	51.23
97.17	96.72	96.27	95.81	95.35	94.89	94.43	93.97
50.12	50.24	50.36	50.48	50.60	50.72	50.83	50.94
96.98	96.53	96.08	95.63	95.17	94.71	94.25	93.79
49.83	49.96	50.08	50.19	50.31	50.43	50.54	50.65
96.79	96.34	95.89	95.44	94.99	94.53	94.08	93.62
49.55	49.67	49.79	49.90	50.02	50.13	50.25	50.36
96.60	96.15	95.71	95.26	94.81	94.36	93.90	93.44
49.26	49.38	49.50	49.61	49.73	49.84	49.96	50.07
96.41	95.97	95.52	95.08	94.63	94.18	93.73	93.27
48.97	49.09	49.21	49.32	49.44	49.55	49.66	49.77
96.23	95.79	95.34	94.90	94.45	94.00	93.55	93.10
48.68	48.80	48.92	49.03	49.15	49.26	49.37	49.48
96.04	95.60	95.16	94.72	94.28	93.83	93.38	92.93
48.39	48.51	48.63	48.74	48.86	48.97	49.08	49.19
95.86	95.42	94.99	94.55	94.11	93.66	93.22	92.77
48.10	48.22	48.34	48.45	48.57	48.68	48.79	48.90
95.68	95.25	94.81	94.37	93.93	93.49	93.05	92.60
47.81	47.93	48.05	48.16	48.27	48.39	48.50	48.60
95.50	95.07	94.64	94.20	93.76	93.33	92.88	92.44
47.52	47.64	47.76	47.87	47.98	48.09	48.20	48.31
95.32	94.90	94.46	94.03	93.60	93.16	92.72	92.28

Table 3.4 Continued

52.14	52.16	52.17	52.18	52.19	52.20	52.21	52.21	52.21
101.31	100.80	100.29	99.78	99.27	98.76	98.25	97.74	97.23
51.83	51.85	51.86	51.87	51.88	51.89	51.89	51.90	51.90
101.28	100.77	100.27	99.76	99.26	98.75	98.25	97.74	97.23
51.52	51.53	51.55	51.56	51.57	51.58	51.58	51.58	51.59
101.25	100.75	100.25	99.75	99.24	98.74	98.24	97.74	97.23
51.21	51.22	51.24	51.25	51.26	51.26	51.27	51.27	51.27
101.23	100.73	100.23	99.73	99.23	98.73	98.23	97.73	97.23
50.89	50.91	50.92	50.93	50.94	50.95	50.96	50.96	50.96
101.20	100.70	100.21	99.71	99.22	98.72	98.23	97.73	97.23
50.58	50.60	50.61	50.62	50.63	50.64	50.64	50.65	50.65
101.17	100.68	100.19	99.70	99.20	98.71	98.22	97.73	97.23
50.27	50.29	50.30	50.31	50.32	50.33	50.33	50.33	50.34
101.15	100.66	100.17	99.68	99.19	98.70	98.21	97.72	97.23
49.96	49.97	49.99	50.00	50.01	50.01	50.02	50.02	50.02
101.12	100.64	100.15	99.66	99.18	98.69	98.21	97.72	97.23
49.65	49.66	49.67	49.69	49.69	49.70	49.71	49.71	49.71
101.10	100.61	100.13	99.65	99.17	98.68	98.20	97.72	97.23
49.33	49.35	49.36	49.37	49.38	49.39	49.39	49.40	49.40
101.07	100.59	100.11	99.63	99.15	98.67	98.19	97.71	97.23
49.02	49.04	49.05	49.06	49.07	49.08	49.08	49.08	49.09
101.05	100.57	100.09	99.62	99.14	98.66	98.19	97.71	97.23
48.71	48.72	48.74	48.75	48.76	48.76	48.77	48.77	48.77
101.02	100.55	100.08	99.60	99.13	98.66	98.18	97.71	97.23
48.40	48.41	48.43	48.44	48.44	48.45	48.46	48.46	48.46
101.00	100.53	100.06	99.59	99.12	98.65	98.18	97.70	97.23
48.09	48.10	48.11	48.12	48.13	48.14	48.14	48.15	48.15
100.98	100.51	100.04	99.57	99.11	98.64	98.17	97.70	97.23
47.77	47.79	47.80	47.81	47.82	47.83	47.83	47.83	47.84
100.95	100.49	100.02	99.56	99.09	98.63	98.16	97.70	97.23
47.46	47.48	47.49	47.50	47.51	47.51	47.52	47.52	47.52
100.93	100.47	100.01	99.55	99.08	98.62	98.16	97.70	97.23
47.15	47.16	47.18	47.19	47.20	47.20	47.21	47.21	47.21
100.91	100.45	99.99	99.53	99.07	98.61	98.15	97.69	97.23

Table 3.5 Radar Image Co-ordinate Grid

52.21	52.21	52.20	52.19	52.18	52.17	52.16	52.14
96.72	96.21	95.70	95.19	94.69	94.18	93.67	93.16
51.90	51.89	51.89	51.88	51.87	51.86	51.85	51.83
96.73	96.22	95.71	95.21	94.70	94.20	93.69	93.19
51.58	51.58	51.58	51.57	51.56	51.55	51.53	51.52
96.73	96.23	95.72	95.22	94.72	94.22	93.72	93.21
51.27	51.27	51.26	51.26	51.25	51.24	51.22	51.21
96.73	96.23	95.74	95.24	94.74	94.24	93.74	93.24
50.96	50.96	50.95	50.94	50.93	50.92	50.91	50.89
96.74	96.24	95.75	95.25	94.75	94.26	93.76	93.27
50.65	50.64	50.64	50.63	50.62	50.61	50.60	50.58
96.74	96.25	95.76	95.26	94.77	94.28	93.79	93.29
50.33	50.33	50.33	50.32	50.31	50.30	50.29	50.27
96.74	96.25	95.76	95.28	94.79	94.30	93.81	93.32
50.02	50.02	50.01	50.01	50.00	49.99	49.97	49.96
96.75	96.26	95.77	95.29	94.80	94.32	93.83	93.35
49.71	49.71	49.70	49.69	49.69	49.67	49.66	49.65
96.75	96.27	95.78	95.30	94.82	94.34	93.85	93.37
49.40	49.39	49.39	49.38	49.37	49.36	49.35	49.33
96.75	96.27	95.79	95.31	94.83	94.35	93.87	93.40
49.08	49.08	49.08	49.07	49.06	49.05	49.04	49.02
96.76	96.28	95.80	95.33	94.85	94.37	93.90	93.42
48.77	48.77	48.76	48.76	48.75	48.74	48.72	48.71
96.76	96.29	95.81	95.34	94.86	94.39	93.92	93.44
48.46	48.46	48.45	48.44	48.44	48.43	48.41	48.40
96.76	96.29	95.82	95.35	94.88	94.41	93.94	93.47
48.15	48.14	48.14	48.13	48.12	48.11	48.10	48.09
96.77	96.30	95.83	95.36	94.89	94.42	93.96	93.49
47.83	47.83	47.83	47.82	47.81	47.80	47.79	47.77
96.77	96.30	95.84	95.37	94.91	94.44	93.98	93.51
47.52	47.52	47.51	47.51	47.50	47.49	47.48	47.46
96.77	96.31	95.85	95.38	94.92	94.46	94.00	93.54
47.21	47.21	47.20	47.20	47.19	47.18	47.16	47.15
96.77	96.31	95.85	95.39	94.93	94.48	94.02	93.56

Table 3.5 Continued

1.23 -1.15	1.11 -1.22	.98 -1.28	.85 -1.34	.72 -1.40	.60 -1.46	.47 -1.51	.34 -1.56	.21 -1.61
1.20 -.95	1.07 -1.01	.95 -1.08	.82 -1.14	.69 -1.20	.57 -1.25	.44 -1.30	.31 -1.36	.18 -1.40
1.17 -.75	1.04 -.82	.92 -.88	.79 -.94	.66 -.99	.54 -1.05	.41 -1.10	.28 -1.15	.16 -1.20
1.14 -.56	1.01 -.62	.89 -.68	.76 -.74	.64 -.79	.51 -.85	.38 -.90	.26 -.95	.13 -.99
1.11 -.36	.98 -.42	.86 -.48	.73 -.54	.61 -.59	.48 -.65	.36 -.70	.23 -.74	.10 -.79
1.07 -.17	.95 -.23	.83 -.29	.70 -.35	.58 -.40	.45 -.45	.33 -.50	.20 -.54	.08 -.59
1.04 .02	.92 -.04	.80 -.10	.67 -.15	.55 -.20	.43 -.25	.30 -.30	.18 -.35	.05 -.39
1.01 .20	.89 .15	.77 .09	.65 .04	.52 -.01	.40 -.06	.27 -.11	.15 -.15	.03 -.19
.98 .39	.86 .33	.74 .28	.62 .23	.50 .18	.37 .13	.25 .08	.12 .04	.00 .00
.95 .57	.83 .52	.71 .47	.59 .41	.47 .37	.35 .32	.22 .27	.10 .23	-.02 .19
.93 .76	.80 .70	.68 .65	.56 .60	.44 .55	.32 .51	.20 .46	.07 .42	-.05 .38
.90 .94	.78 .88	.66 .83	.54 .78	.41 .74	.29 .69	.17 .65	.05 .61	-.07 .57
.87 1.11	.75 1.06	.63 1.01	.51 .96	.39 .92	.27 .88	.15 .83	.02 .79	-.10 .76
.84 1.29	.72 1.24	.60 1.19	.48 1.14	.36 1.10	.24 1.06	.12 1.02	-.00 .98	-.12 .94
.81 1.47	.69 1.42	.57 1.37	.46 1.32	.34 1.28	.22 1.24	.10 1.20	-.02 1.16	-.15 1.12
.78 1.64	.67 1.59	.55 1.54	.43 1.50	.31 1.46	.19 1.41	.07 1.38	-.05 1.34	-.17 1.30
.76 1.81	.64 1.76	.52 1.72	.40 1.67	.28 1.63	.17 1.59	.05 1.55	-.07 1.52	-.19 1.48

Table 3.6 Differential Co-ordinate Grid

.08 -1.66	-.05 -1.70	-.18 -1.74	-.31 -1.78	-.44 -1.82	-.57 -1.85	-.70 -1.88	-.84 -1.91
.05 -1.45	-.08 -1.49	-.21 -1.53	-.34 -1.57	-.47 -1.60	-.60 -1.64	-.73 -1.66	-.86 -1.69
.03 -1.24	-.10 -1.28	-.23 -1.32	-.36 -1.36	-.49 -1.39	-.62 -1.42	-.75 -1.45	-.88 -1.48
.00 -1.03	-.13 -1.08	-.26 -1.11	-.38 -1.15	-.51 -1.18	-.64 -1.21	-.77 -1.24	-.90 -1.27
-.02 -.83	-.15 -.87	-.28 -.91	-.41 -.94	-.54 -.98	-.66 -1.01	-.79 -1.03	-.92 -1.06
-.05 -.63	-.18 -.67	-.30 -.71	-.43 -.74	-.56 -.77	-.69 -.80	-.81 -.83	-.94 -.85
-.08 -.43	-.20 -.47	-.33 -.50	-.45 -.54	-.58 -.57	-.71 -.60	-.84 -.62	-.96 -.65
-.10 -.23	-.23 -.27	-.35 -.31	-.48 -.34	-.60 -.37	-.73 -.40	-.86 -.42	-.98 -.44
-.12 -.04	-.25 -.08	-.37 -.11	-.50 -.14	-.63 -.17	-.75 -.20	-.88 -.22	-1.00 -.24
-.15 .15	-.27 .12	-.40 .08	-.52 .05	-.65 .02	-.77 -.00	-.90 -.03	-1.02 -.05
-.17 .34	-.30 .31	-.42 .28	-.54 .25	-.67 .22	-.79 .19	-.92 .17	-1.04 .15
-.20 .53	-.32 .50	-.44 .47	-.57 .44	-.69 .41	-.82 .38	-.94 .36	-1.06 .34
-.22 .72	-.34 .69	-.47 .66	-.59 .63	-.71 .60	-.84 .57	-.96 .55	-1.08 .53
-.24 .91	-.37 .87	-.49 .84	-.61 .81	-.73 .79	-.86 .76	-.98 .74	-1.10 .72
-.27 1.09	-.39 1.06	-.51 1.03	-.63 1.00	-.75 .97	-.88 .95	-1.00 .93	-1.12 .91
-.29 1.27	-.41 1.24	-.53 1.21	-.65 1.18	-.78 1.16	-.90 1.13	-1.02 1.11	-1.14 1.09
-.31 1.45	-.43 1.42	-.55 1.39	-.67 1.36	-.80 1.34	-.92 1.32	-1.04 1.30	-1.16 1.28

Table 3.6 Continued

291.77	295.58	299.89	304.74	310.17	316.16	322.66	329.55	336.66
212.61	199.81	188.00	177.39	168.20	160.69	155.09	151.61	150.41
287.94	291.75	296.15	301.20	306.98	313.51	320.74	328.53	336.64
199.71	186.02	173.28	161.71	151.59	143.21	136.91	132.97	131.61
283.61	287.34	291.73	296.92	303.03	310.14	318.24	327.19	336.63
187.81	173.18	159.43	146.78	135.55	126.13	118.93	114.38	112.80
278.72	282.26	286.52	291.72	298.05	305.73	314.87	325.33	336.61
177.11	161.53	146.69	132.84	120.33	109.61	101.25	95.88	94.00
273.25	276.44	280.39	285.36	291.70	299.82	310.10	322.60	336.60
167.86	151.33	135.38	120.24	106.26	93.96	84.07	77.52	75.20
267.21	269.86	273.23	277.64	283.55	291.68	302.97	318.20	336.58
160.29	142.89	125.88	109.44	93.87	79.69	67.75	59.45	56.40
260.66	262.57	265.07	268.44	273.21	280.34	291.65	310.07	336.57
154.66	136.55	118.63	101.02	83.91	67.67	53.12	42.02	37.60
253.70	254.71	256.05	257.90	260.63	265.04	273.18	291.63	336.55
151.18	132.59	114.06	95.62	77.32	59.31	41.95	26.55	18.80
246.53	246.53	246.53	246.53	246.53	246.53	246.53	246.52	156.56
150.00	131.25	112.50	93.75	75.00	56.25	37.50	18.75	.01
239.36	238.36	237.02	235.17	232.43	228.02	219.88	201.43	156.52
151.18	132.59	114.07	95.62	77.33	59.32	41.96	26.56	18.81
232.41	230.50	228.00	224.63	219.86	212.72	201.41	183.00	156.51
154.66	136.55	118.64	101.03	83.92	67.68	53.12	42.03	37.61
225.86	223.21	219.83	215.43	209.52	201.39	190.09	174.87	156.49
160.29	142.89	125.89	109.45	93.88	79.69	67.76	59.46	56.41
219.82	216.63	212.68	207.71	201.37	193.25	182.97	170.47	156.48
167.86	151.33	135.38	120.25	106.27	93.96	84.08	77.53	75.21
214.35	210.82	206.55	201.36	195.02	187.34	178.20	167.74	156.46
177.11	161.53	146.69	132.85	120.33	109.62	101.26	95.89	94.02
209.47	205.74	201.34	196.15	190.05	182.94	174.83	165.88	156.45
187.81	173.19	159.43	146.79	135.56	126.13	118.94	114.39	112.82
205.13	201.32	196.93	191.87	186.09	179.56	172.33	164.54	156.43
199.71	186.02	173.28	161.72	151.59	143.22	136.92	132.98	131.62
201.31	197.50	193.19	188.33	182.91	176.92	170.41	163.52	156.42
212.61	199.80	188.00	177.39	168.21	160.69	155.10	151.62	150.42

Table 3.7 Satellite Image Co-ordinate Grid in Polar Co-ordinates

343.77	350.67	357.18	3.19	8.64	13.51	17.84	21.67
151.53	154.93	160.46	167.92	177.05	187.61	199.38	212.16
344.75	352.56	359.81	6.36	12.15	17.23	21.64	25.46
132.90	136.77	143.02	151.35	161.43	172.96	185.67	199.34
346.07	355.03	3.15	10.28	16.40	21.61	26.02	29.76
114.32	118.82	125.97	135.35	146.55	159.17	172.91	187.52
347.90	358.37	7.53	15.23	21.58	26.79	31.07	34.61
95.83	101.16	109.48	120.17	132.67	146.50	161.33	176.90
350.60	3.11	13.42	21.56	27.91	32.89	36.84	40.03
77.48	84.00	93.86	106.15	120.12	135.25	151.19	167.71
354.98	10.22	21.53	29.67	35.59	40.00	43.38	46.03
59.42	67.70	79.62	93.80	109.37	125.80	142.81	160.20
3.08	21.51	32.83	39.97	44.74	48.12	50.62	52.53
42.01	53.09	67.64	83.88	100.99	118.60	136.51	154.62
21.49	39.94	48.09	52.50	55.23	57.08	58.42	59.43
26.55	41.94	59.30	77.31	95.61	114.05	132.58	151.17
66.55	66.54	66.54	66.54	66.54	66.54	66.54	66.54
18.75	37.50	56.25	75.00	93.75	112.50	131.25	150.00
111.60	93.14	84.99	80.58	77.85	75.99	74.66	73.65
26.55	41.95	59.31	77.32	95.61	114.06	132.59	151.17
130.00	111.57	100.25	93.11	88.33	84.96	82.46	80.55
42.02	53.10	67.65	83.88	100.99	118.60	136.51	154.62
138.10	122.86	111.55	103.41	97.49	93.08	89.70	87.05
59.43	67.71	79.63	93.81	109.37	125.81	142.81	160.21
142.47	129.96	119.66	111.52	105.17	100.19	96.23	93.04
77.50	84.01	93.87	106.16	120.13	135.25	151.20	167.72
145.18	134.70	125.55	117.84	111.50	106.29	102.01	98.47
95.84	101.17	109.49	120.19	132.68	146.51	161.34	176.91
147.01	138.05	129.92	122.80	116.67	111.47	107.06	103.32
114.34	118.83	125.98	135.37	146.57	159.19	172.92	187.53
148.32	140.52	133.27	126.72	120.92	115.85	111.44	107.62
132.91	136.79	143.04	151.36	161.45	172.98	185.69	199.36
149.31	142.41	135.89	129.88	124.44	119.57	115.24	111.41
151.55	154.95	160.48	167.94	177.07	187.63	199.41	212.18

Table 3.7 Continued

```
 23.23  23.24  23.24  23.25  23.27  23.28  23.30  23.32  23.34
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.25  23.25  23.25  23.26  23.28  23.29  23.31  23.34  23.36
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.26  23.26  23.27  23.27  23.28  23.30  23.32  23.35  23.37
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.28  23.28  23.28  23.28  23.29  23.31  23.33  23.36  23.39
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.31  23.31  23.30  23.30  23.30  23.31  23.33  23.37  23.40
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.34  23.34  23.33  23.33  23.32  23.32  23.34  23.37  23.42
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.38  23.37  23.37  23.36  23.35  23.35  23.35  23.37  23.43
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.42  23.42  23.41  23.41  23.40  23.39  23.38  23.37  23.45
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.47  23.47  23.47  23.47  23.47  23.47  23.47  23.48   0.00
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.51  23.51  23.52  23.52  23.53  23.54  23.56  23.57  23.48
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.55  23.56  23.56  23.57  23.58  23.59  23.59  23.57  23.49
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.59  23.59  23.60  23.61  23.61  23.61  23.60  23.56  23.51
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.62  23.62  23.63  23.63  23.63  23.62  23.60  23.56  23.52
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.64  23.65  23.65  23.64  23.64  23.62  23.60  23.57  23.54
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.66  23.66  23.66  23.66  23.64  23.63  23.61  23.58  23.55
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.68  23.68  23.67  23.66  23.65  23.63  23.61  23.59  23.57
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00

 23.69  23.69  23.68  23.67  23.66  23.64  23.62  23.60  23.58
  1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00   1.00
634.8425119754
587.3576953916
```

Table 3.8   Azimuth and Range Correction Factors

23.36	23.37	23.37	23.37	23.36	23.36	23.34	23.33
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.38	23.39	23.39	23.39	23.38	23.37	23.36	23.35
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.39	23.41	23.41	23.41	23.40	23.39	23.38	23.37
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.41	23.43	23.43	23.43	23.42	23.41	23.40	23.38
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.44	23.45	23.45	23.44	23.43	23.42	23.41	23.40
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.46	23.47	23.47	23.46	23.45	23.43	23.43	23.42
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.49	23.49	23.48	23.47	23.45	23.45	23.44	23.43
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.51	23.49	23.48	23.47	23.46	23.45	23.45	23.45
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.45	23.46	23.46	23.46	23.46	23.46	23.46	23.46
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.40	23.42	23.44	23.45	23.46	23.47	23.47	23.48
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.43	23.43	23.44	23.45	23.47	23.48	23.48	23.49
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.46	23.45	23.45	23.46	23.48	23.49	23.50	23.51
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.49	23.47	23.47	23.48	23.49	23.50	23.51	23.52
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.51	23.50	23.49	23.50	23.50	23.51	23.53	23.54
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.53	23.52	23.51	23.51	23.52	23.53	23.54	23.55
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.55	23.54	23.53	23.53	23.54	23.55	23.56	23.57
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
23.57	23.56	23.55	23.55	23.56	23.56	23.57	23.59
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 3.8 Continued

mark) and transferred to the computer memory using DMA. When the transfer is complete, the tape drive interrupts the computer. The computer then reads the tape word counter and status. The word counter contains the actual record length. Since the record length is not fixed, the tape drive is commanded to read until the next end of file mark or 65534 bytes, whichever is less. The data in the buffer is sent to the serial port. When the transfer is complete, the computer commands the tape drive to read another record. 250 nulls are sent to the serial port (as an inter-record gap), which gives the tape drive enough time to read the next record (.29 sec.). Hence continuous data is sent to the serial port. This is required by the phase-encoding tape interface so that there will be no synchronization problem on the first data byte. To indicate the start of a new record (and the end of the null sequence), a sync byte (16 hexadecimal) is sent. The nulls are FF hexadecimal rather than 00 since the UART in the microcomputer would not synchronize to a 00 null sequence, with only the stop-bit being high.

3.4.2 Identifying Tapes

Each file (set of data records) is identified with a header record as seen in Table 3.9. Appendix 8 is a microcomputer program to read the header record and print it. Table 3.9 is a listing of all the header records. The orbit number (number of orbital revolutions since launch) will be used in referencing particular images under study. Fig. 3.8 is a flowchart for the program section which reads the tape into memory. This is a routine which is used in many other programs. The coding is critical due to the nature of the 8251 UART which makes it difficult to program.

```
ØRBITAL DATA FØR NØAA5
ØRBIT NUMBER: 7989
ASCENDING NØDE: -87.65
EQUATØR CRØSSING TIME: 126D 2H 26M 56S Z
ACQUISITIØN TIME: 126D 2H 41M 56S Z
DATA TAPE TYPE: 00
TAPE #: 00


ØRBITAL DATA FØR NØAA5
ØRBIT NUMBER: 8063
ASCENDING NØDE: -79.89
EQUATØR CRØSSING TIME: 131D 1H 55M 49S Z
ACQUISITIØN TIME: 131D 1H 59M 19S Z
DATA TAPE TYPE: 02
TAPE #: 00


ØRBITAL DATA FØR NØAA5
ØRBIT NUMBER: 8001
ASCENDING NØDE: -76.66
EQUATØR CRØSSING TIME: 127D 1H 42M 57S Z
ACQUISITIØN TIME: 127D 1H 46M 30S Z
DATA TAPE TYPE: 02
TAPE #: 00


ØRBITAL DATA FØR NØAA5
ØRBIT NUMBER: 7971
ASCENDING NØDE: 76.40
EQUATØR CRØSSING TIME: 124D 15H 32M 53S Z
ACQUISITIØN TIME: 124D 16H 11M 53S Z
DATA TAPE TYPE: 00
TAPE #: 00
```

Table 3.9 Header Record Listing

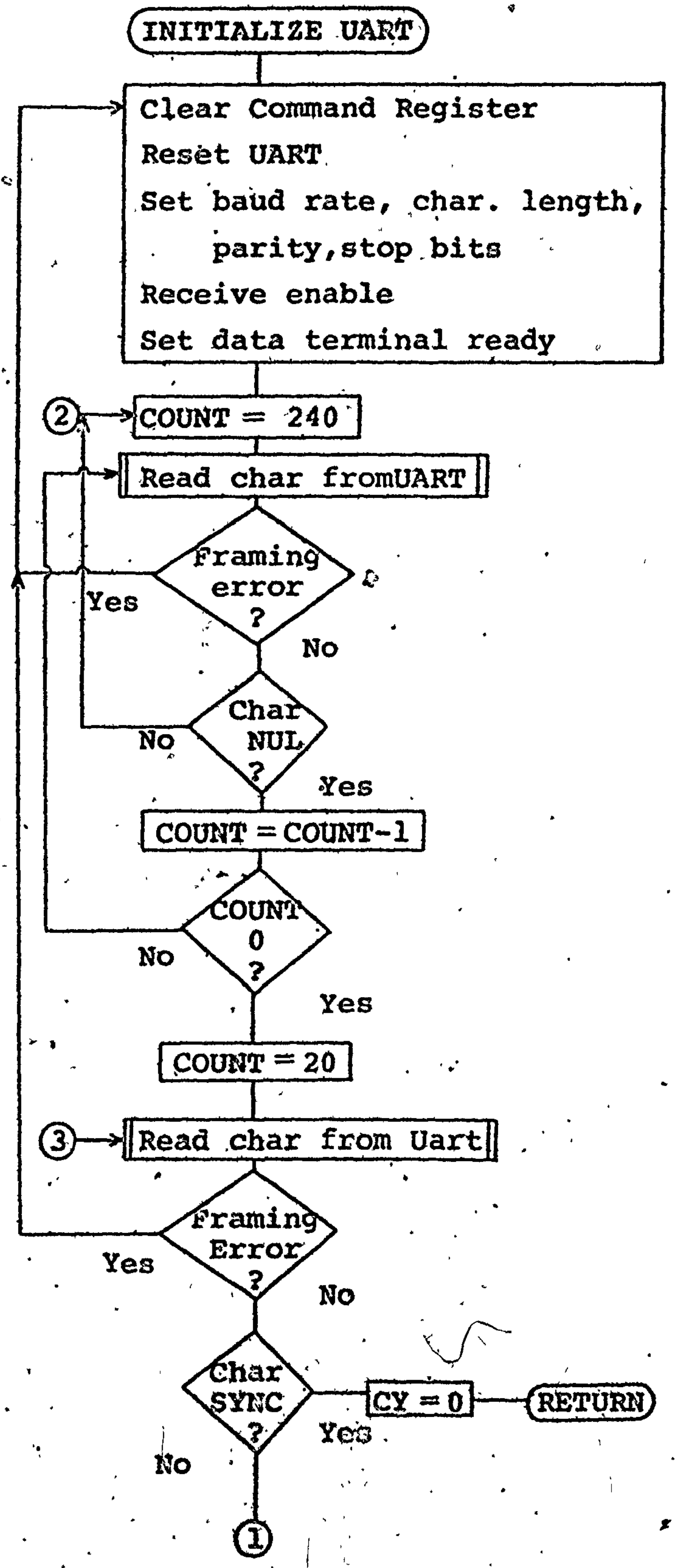


Fig. 3.8 Flowchart for Magnetic Tape Read

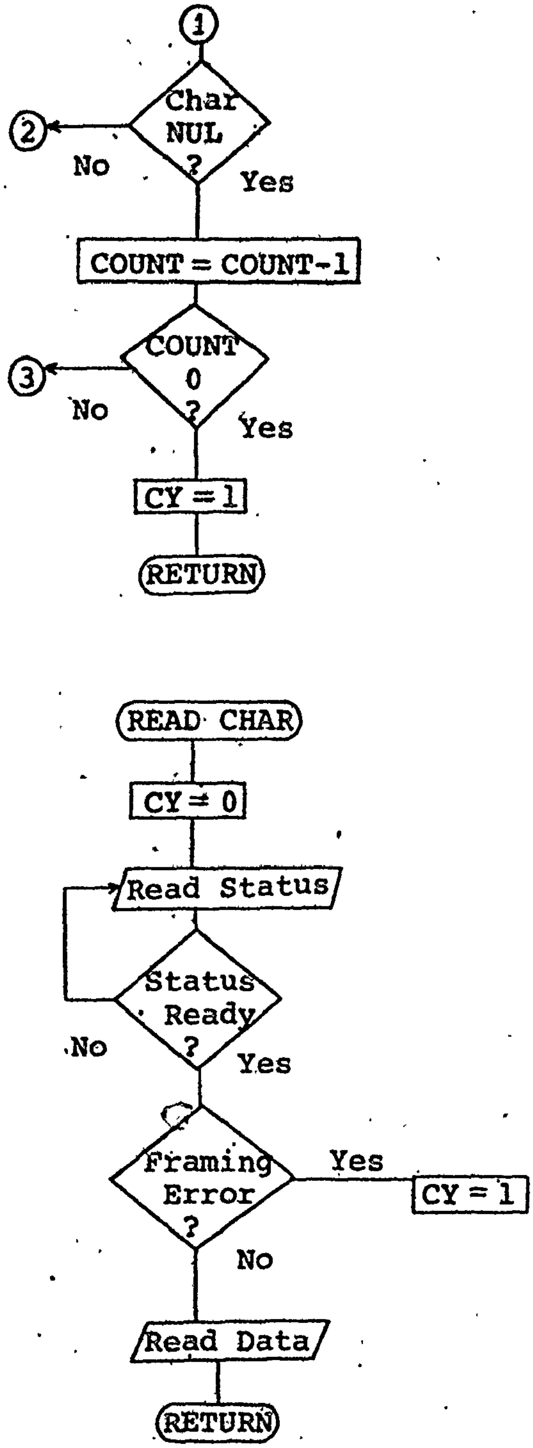


Fig. 3.8 Continued

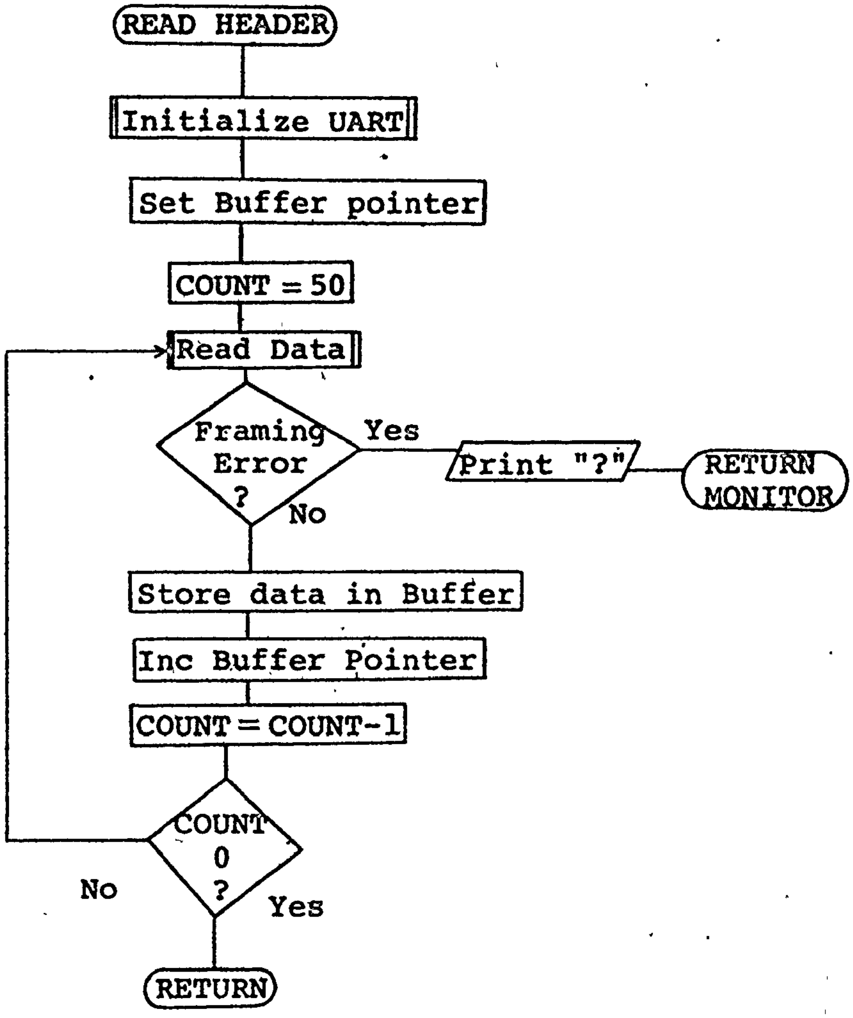


Fig. 3.8 Continued

At the end of Appendix 8 is a Basic language implementation of the header record data printout program. The assembly language program was written before the Basic language interpreter was obtained and configured for the microcomputer system. The demonstrates the considerable programming simplification possible when using a high level language, particularly for non-real time applications where execution speed is not important.

3.4.3 Data Acquisition For Display

In order to display a small geographical area, data is selectively read from the raw data. If the satellite has an orbital period of 116.29 min. and a scan rate of 360 lines/min., the scan lines will be spaced by 0.473 nmi. If 256 lines are read into the display, the vertical coverage will be 121.0 nmi. With 60 degrees of sensor scan digitized to 1951 points, each pixel is separate by 0.511 nmi. 256 elements cover 130.8 nmi. If a 300 x 300 nmi. image is desired, 635 lines with 587 elements will be required. In order to display the image on a 256 line by 256 element display, each display line should be produced from 2.48 scan lines and each display element should be produced from 2.29 elements of the raw data. Hence the image size must be scaled by horizontal and vertical factors. Since, in general, scale factors are not integer numbers, an estimate of the value of the desired element must be made based on the values of the given elements. Bernstein [7] discusses practical methods of estimation, namely nearest neighbour, bi-linear interpolation and cubic spline interpolation. The nearest neighbour algorithm was implemented mainly due to its simplicity. The results from the three algorithms do not differ greatly for imagery containing natural features. A significant improvement is achieved using the cubic spline algorithm where man-made features are encountered, particularly diagonal line structures.

If a scale factor of 2.29 is used, the nearest neighbour algorithm would read elements 1, 3, 6, 8, 10, 12, 15, 17, ..., 587 from the raw data tape into a display line of 256 elements. A similar process is used for reading the lines. In addition to scaling, the starting line and element (column) must be known. A program computes the line number (since equator crossing) and element column from the scan line start for the centre of the image desired. For orbit 7971, the image centre (Winnipeg airport) should be located at line 16576 and column 1150. Since acquisition time (start of data tape) occured 2340 seconds after equator crossing and the start of the image is at line 16259 (1657.35 - 635/2), the first line is contained in record 2219. The tape is searched for record 2219 and the image is read from the tape.

Appendix 9 is a program for reading the raw data tapes and Fig. 3.9 is the corresponding flowchart. The program clears the image memory and requests the following input response:

"DAY OR NIGHT PASS?"	- input either "D" or "N"
"ORBIT NUMBER="	- input orbit number to verify correct tape
"START COLUMN="	- element number from start of data record (i.e. an X-shift factor)
"LINE SINCE AOS="	- record number of first line (i.e. y-shift factor)
"VISUAL OR IR?"	- input "V" or "I" to select visual or IR image

	(assuming data is present as indicated by tape type)
"RECORD LENGTH="	- normally 2016 or 4016 depending on tape type)
"START TAPE G,"	- position tape just before header start, start tape, type "G" and when tone starts type carraige return.

If the header is correctly read with the correct orbit number as requested, an "R" will be typed. If not, "WRONG ORBIT" will be printed and control will return to the monitor. Reposition the tape after checking tape counter, controls, etc. Start the tape and repeat the "G" procedure. The tape will be read and searched for the desired record number. The image will be read according to the scale factors set by locations INCL and INCP. INCL is the increment in records per required line, and consists of three bytes; an integer part INCL+2, INCL+1 and the fractional part INCL. INCP is the inverse of the column increment per horizontal element. Location INCP+2 is the integer part and INCL+1, INCL is the fractional part.

If a required record is not found, an error message will be printed: "LINE NOT FOUND=" PREC, where PREC is the record number being searched for. If a framing error occurs, usually due to tape problems, a message will be printed: "READ ERROR IN LINE=" PREC "BYTE=" BNUM, where BNUM is the byte number in the record. If an end of tape is encountered, "END OF TAPE" will be printed. Mount the continuation tape and continue the program. When all the required data is read, "READ DONE" will be printed.

As the tape is read, the process can be monitored on the display. If the pass is a daytime one, the display will be filled in

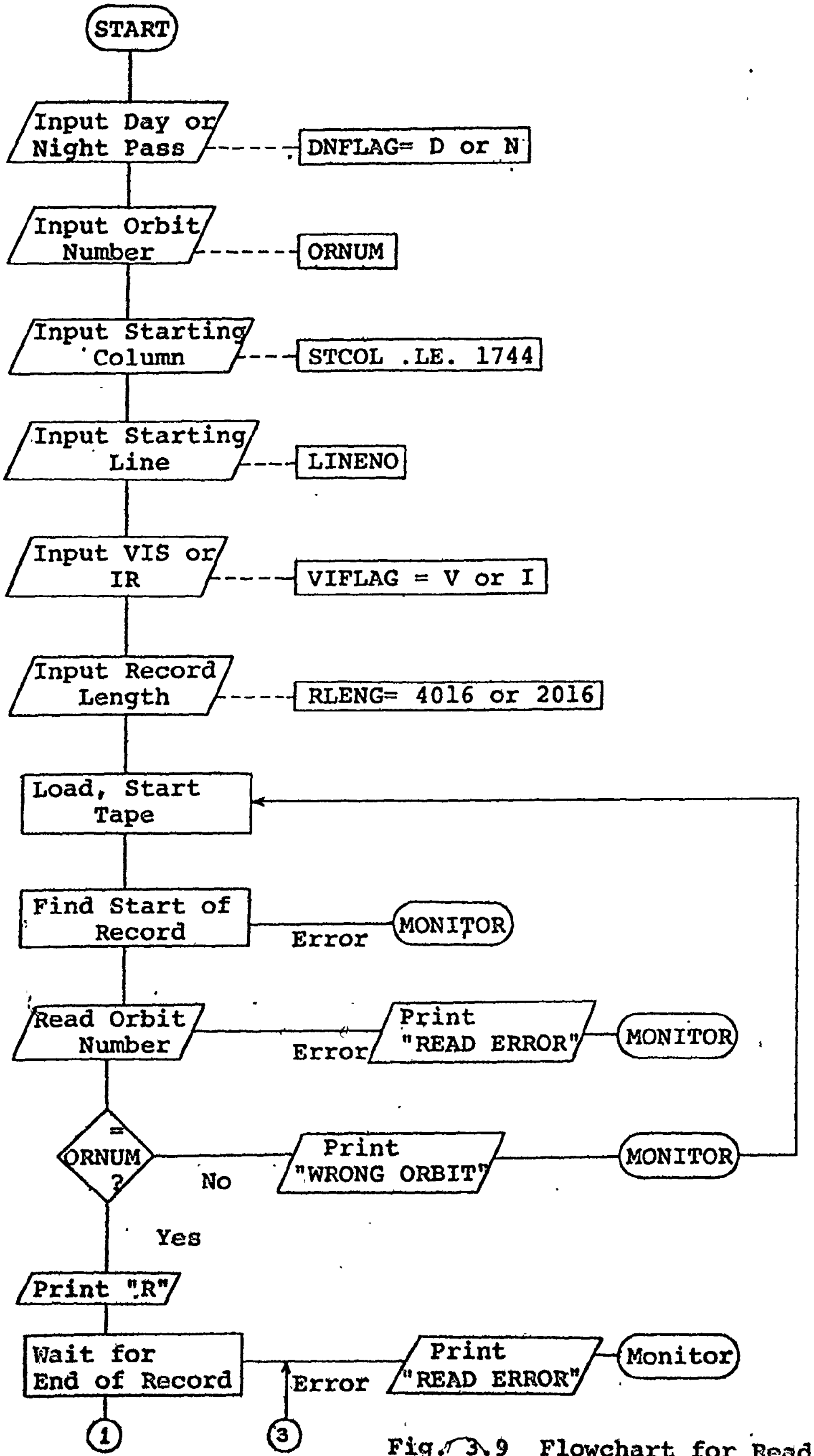


Fig. 3.9 Flowchart for Reading Raw Data Tapes

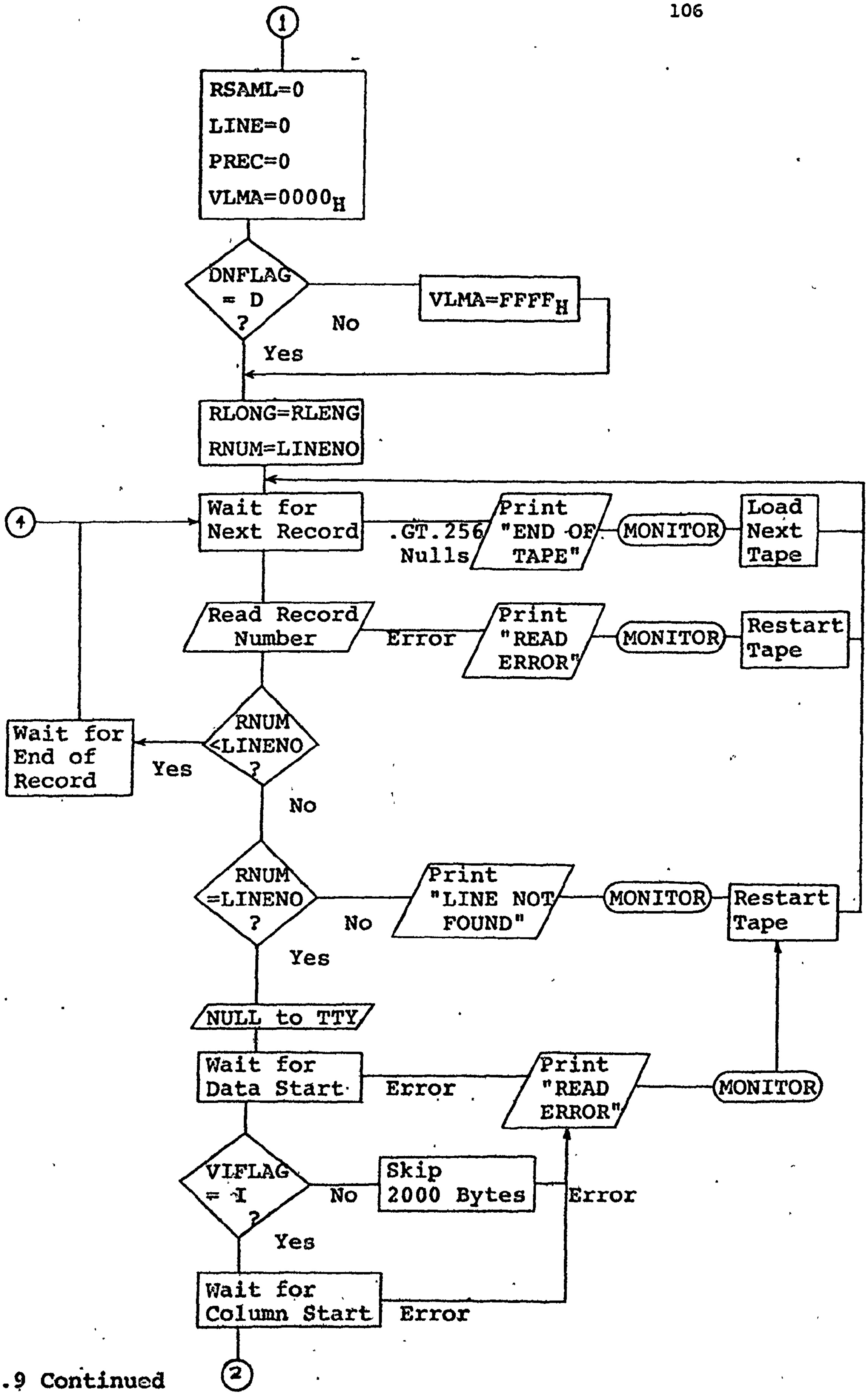


Fig. 3.9 Continued

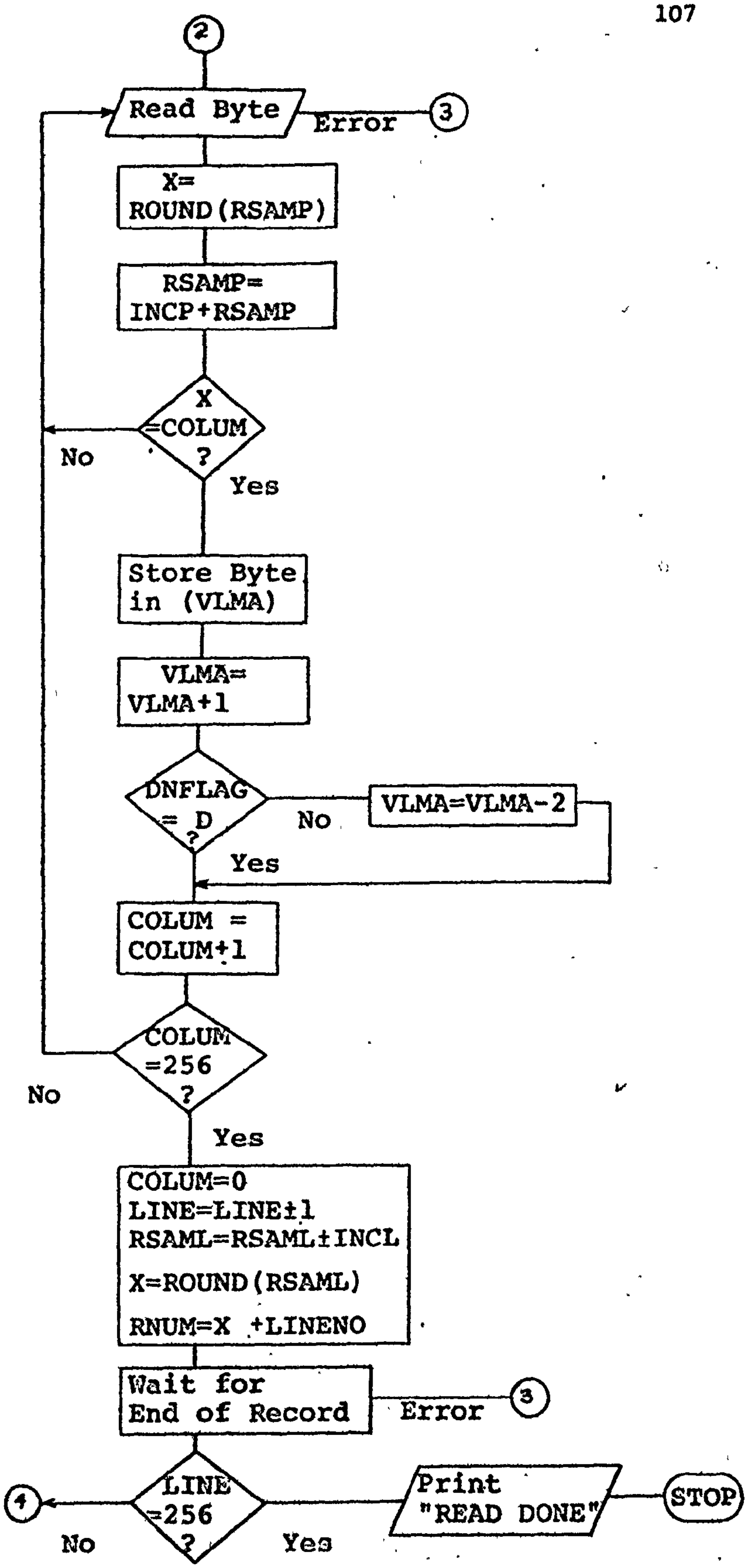


Fig. 3.9 Continued

the normal scan direction. For a nighttime pass, the display will be filled backwards, reading from the bottom and right to left. In the nighttime, the satellite moves from south to north and effectively the satellite views the earth upside-down.

3.4.4 Position Corrections

Due to several sources of error, including satellite roll and prediction errors, the image will be a few elements offset from the predicted position. Visible features, particularly land-water interfaces, are often used to verify and correct positioning errors. Once geographical features are identified, they can be traced onto a clear plastic overlay on the monitor face. Assuming horizontal and vertical linearity, distances can be compared to a map and corrections to horizontal and vertical position determined. Corrections can be made to the position by either re-reading the raw data tape with corrected starting column and record or using the image translation program. If the translation program is used, data near two edges (depending on the shift direction) will be missing. It was found that vertical and horizontal scale factors and rotation were as predicted. The cursor program can be used to determine the vertical and horizontal position of various points on the screen, in order to compare positions on a map.

3.4.5 Transfer of Image Memory to Tape

After various stages of processing, the image memory may be saved on magnetic tape and later read back from the tape. Appendix 10 is a program to read and write to tape. The entire 128 kilobytes is transferred. 1000 null characters are first written and then the data is written continuously. No error checking is employed since an error will normally affect the rest of the data read in. An error will usually result in a framing error detected by the read program or an undetected error will normally be visible on the display. To ensure data integrity, the image memory should be written twice. The normal causes of errors are bad sections of tape (non-recoverable) or dirt in the tape heads.

CHAPTER 4

RADAR DATA ACQUISITION AND PROCESSING

4.1 RADAR DATA ACQUISITION

4.1.1 Introduction

The radar used in the satellite-radar study was a long range (150 nmi) L-band Airport and Airways Surveillance Radar (AASR). It has a pulse repitition frequency (PRF) of 380 Hz with a pulse width of 2 μsec. The antenna rotation rate is 6 rpm, causing a complete 380 degree sweep in 10 seconds. The Winnipeg AASR provided both normal and Moving Target Indicator (MTI) video processing. The radar video signals were recorded in real time on a two-channel wideband RCA Advisor video recorder. The video recorder allowed 30 minutes of continuous data to be recorded on one tape. In order to compare the satellite and radar data, the radar signals were recorded during the time of the satellite pass.

The radar forms an image of radio reflecting objects using a polar scan, where scan lines radiate from the centre of the image which are swept or rotated about the centre. This is a different scanning technique from the rectangular scanning system of the satellite. If the radar has a PRF of 380 Hz (resulting in 380 scans per second) one antenna rotation or sweep in 10 seconds will contain 3600 scans. The antenna azimuth beamwidth of the radar at the -3 dB points is 1.35 degrees. Point targets 1.35 degrees apart in azimuth at the same range can be barely resolved. The actual ground distance between the two targets is proportional to range. Hence the radar distance resolution varies with range whereas satellite image resolution is fairly constant over the image. At a beamwidth of 1.35 degrees, at least 13 echos will be received from a target using a PRF of 380 Hz. An azimuth of 1.35 degrees corresponds to a ground distance of 2.8 nmi at a range of 60 nmi.

The AASR pulse width of 2 μsec allows of ground resolution in the range direction of about .2 nmi, independent of range.

4.1.2 Recording Format

The radar video signal can be directly recorded on tape using a wideband recorder, with a bandwidth on the order of 2 MHz. Two channels were used, one for normal video and the other for MTI processed video. The video signal was a unipolar positive signal, with a maximum playback output voltage of .5 volt. Recorded along with the normal video signal were system trigger pulses. These pulses are negative, with a peak amplitude -.6 volts and a pulse width of 2μsec. The system trigger pulses correspond to the transmitter pulses.

In order to establish an azimuth reference, North-mark or azimuth reference pulses (ARP) were recorded on one of the audio channels. These are negative pulses, with a peak amplitude of -.6 volt and a pulsewidth of 20 μsec. The second audio channel contains azimuth clock pulses (ACP), which are positive pulses, with a peak amplitude of .8 volt and a pulse width of 70 μsec. The ACP's are derived from an antenna position sensor, producing 4096 pulses per revolution. By counting ACP's since the ARP gives a direct indication of the antenna azimuth, independent of the PRF, antenna rotation speed or variations in playback tape speed.

4.1.3 Sampling the Radar Data

In order to display the radar images on the image display system (rectangular), the radar scanning must be converted from polar to rectangular form. The radar signal is first sampled in polar form (as a function of time), converted to binary form and stored in the 128 kilo-byte RAM. As a compromise between the number of samples and number of quantization levels, 262144 (2^{18}) 4-bit samples were taken, with 128 samples in range and 2048 samples in azimuth. This polar data is converted to a 256 x 256 element rectangular form. If the radar maximum range circle inscribes a square of dimension 256 elements, then a range scan requires 128 elements. Sampling 2048 range scans with 128 samples per scan ensures a sufficient density of samples for the polar to rectangular scan conversion process. In the simplest form of the scan conversion, each point in the rectangular output image is selected from the input polar image using the nearest neighbour technique. The density of points in the input image must be sufficient so that the maximum distance difference (error) between the desired point and the closest available point is less than 1/2 element spacing in the output image.

Hardware was constructed to interface with the video recorder, digitize the video and store it in the memory unit. Figure 4-1 is a block diagram of the hardware system.

The sweep sampling process is initiated by the first ARP after reset and start. The wait period can be up to 10 seconds. Every second ACP initiates the range sweep wait period. The next system trigger pulse initiates the sampling of a range sweep. 128 samples are taken at a rate of 1/16th the clock frequency. The required clock frequency is given by:

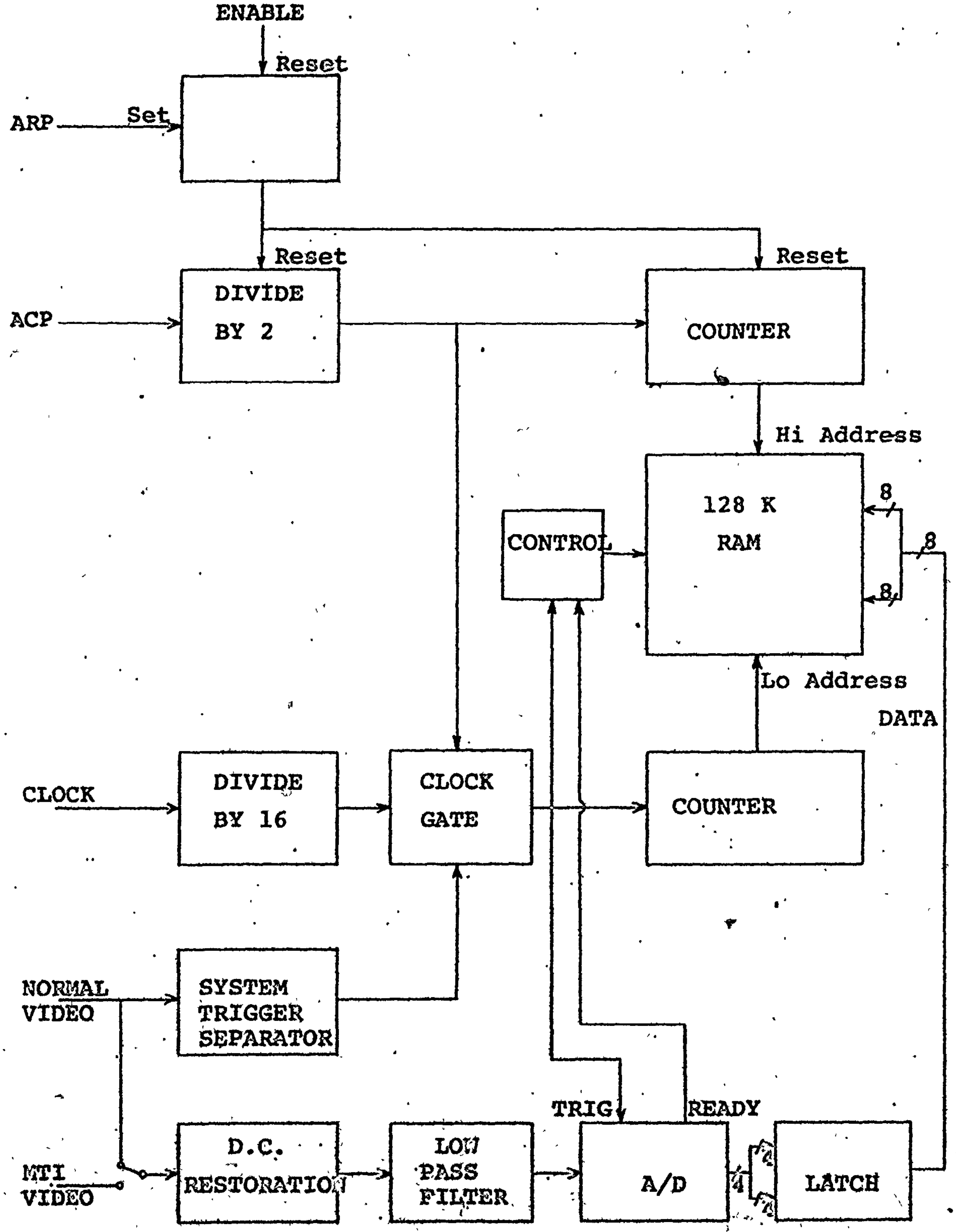


Fig. 4.1 Radar Digitizing Hardware Block Diagram

$$f_c = 16 \times \frac{128\ c}{2R}$$

where

R = maximum range (edge)

c = velocity of light

The low-pass filter cut-off frequency (maximum) is $f_0 = f_c/2$ in order to satisfy the Nyquist criterion.

4.1.4 Hardware

Figure 4.2 is the interface and video processing hardware. Transistor Q1 and Schmitt trigger G_1 convert the ARP signal from the tape to a TTL pulse signal. The base bias potentiometer sets the threshold level to discriminate a low from a high logic level. An optimum setting is where all ARP pulses are recognized and no false pulses are produced from noise. Q2 and G6 form a similar circuit for the ACP pulses. Q3 and G2 remove the system trigger pulses from the normal video output of the recorder. The base bias potentiometer is adjusted so that the video signal itself does not cause false system trigger pulses.

Q5 and Q4 process the video signal. The switch S1 selects either normal or MTI video. The function of Q5 is to clip-off the negative system trigger pulses from the normal video and buffer the signal. Q5 is an emitter follower amplifier. The base bias potentiometer sets the clipping threshold. The MTI video is not affected since it does not contain system trigger pulses.

Since the video from the radar system is A.C. coupled (i.e. the D.C. component is lost), the D.C. component must be restored. This is the function of D1, as a negative peak detector. In the case of the radar video, the minimum video level (negative peaks) is zero. The zero level of the video signal from the radar system fluctuates due to the A.C. coupled nature of the signal. D1 almost completely restores the signal to its correct form. D2 and D3 set the bias (minimum level) for Q4. The video amplitude is adjusted by the 100 K potentiometer.

The A/D converter produces an offset binary output, where a zero binary count (minimum) requires a -.5 volt input. The maximum binary count requires a +.5 volt input. The level control and V^- is adjusted so that these requirements are met, with the video signal filling the entire range of the A/D converter.

Figure 4.4 initiates and counts the ACP's. Figure 4.6 is the corresponding timing diagram. Switch S2 resets and starts the sampling process. When S2 is switched to the start position, FF3 is set by the next ARP to arrive. This could take as long as 10 seconds. C4, C5, and C6 count the ACP pulses after being enabled by FF3. When the maximum count of 4095 is reached (one antenna rotation), the output of G9 goes low, disabling the counters from being clocked and hence the count is held at 4095, until reset by S2.

In Figure 4.3, FF1 is clocked by the even ACP's (ACP/2 output from C4, QA). When FF1 is set, FF2 will be set by the next system trigger to arrive after the even ACP. C4, C5, and C6 provide the high order address to the memory (azimuth bin count).

FF2 enables C1, C2, and C3 to count and cause the A/D to sample 128 points in range, after receiving the system trigger. The timing diagram is shown in Figure 4.7. After the counters are enabled, the A/D converter will be triggered 8 clock pulses later, from the sampling clock, and at intervals of 16 clock pulses thereafter. When the A/D converter is triggered, it converts the video input to a 4-bit binary equivalent and issues a data ready pulse after the conversion is complete.

Counters C2 and C3 count 128 A/D samples and also supply the memory with the lower address bits. After 128 samples in range are taken, QD of C3 goes high, resetting FF1 and FF2, which in turn reset C1, C2 and C3.

Since the A/D data samples are 4 bits in length, two A/D samples must be catenated together, forming an 8 bit byte which then can be written to memory. This is performed by the logic in Figure 4.5. The multiplexer select line, MUX SEL, is effectively the least significant address bit. The first A/D sample is held in L1 and the second in L2. The A/D converter indicates the conversion is complete by the data ready output, DATA RDY1. This clocks the data into L1 or L2. BCL1 and BCL2 controls which half of the 16-bit word is written to memory. The byte select signal, BYTE SEL, selects which half is written.

The first 131072 samples are written into channel 0 of the memory and the remainder are written into channel 1. In Figure 4.3, RP1 and RP2 are memory cycle signals for channel 0 and 1 respectively. The cycle pulses are generated by M1, when the A/D data is ready on the odd samples. The most significant address bit, CHAN, causes the cycle pulses to cycle either channel 0 or 1.

When the sampling process is complete, (after 10 seconds) the computer may process the radar data. In the present configuration, the memory must be physically disconnected from the radar sampling hardware and connected to the computer interface, with memory power on. A switch on the memory interface disables memory cycling during the process of

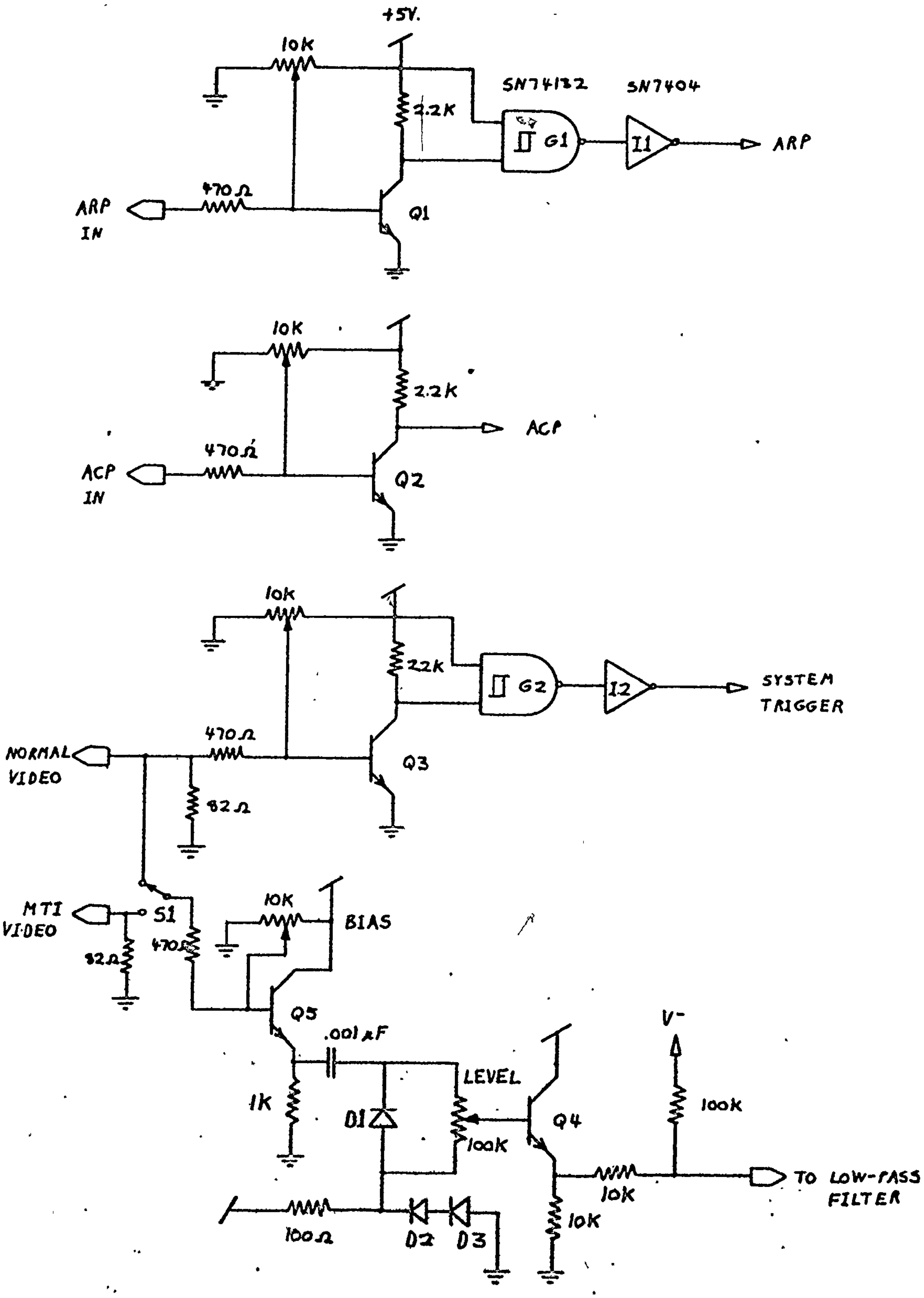


Fig. 4.2 Radar Video Processing and Recorder Interface

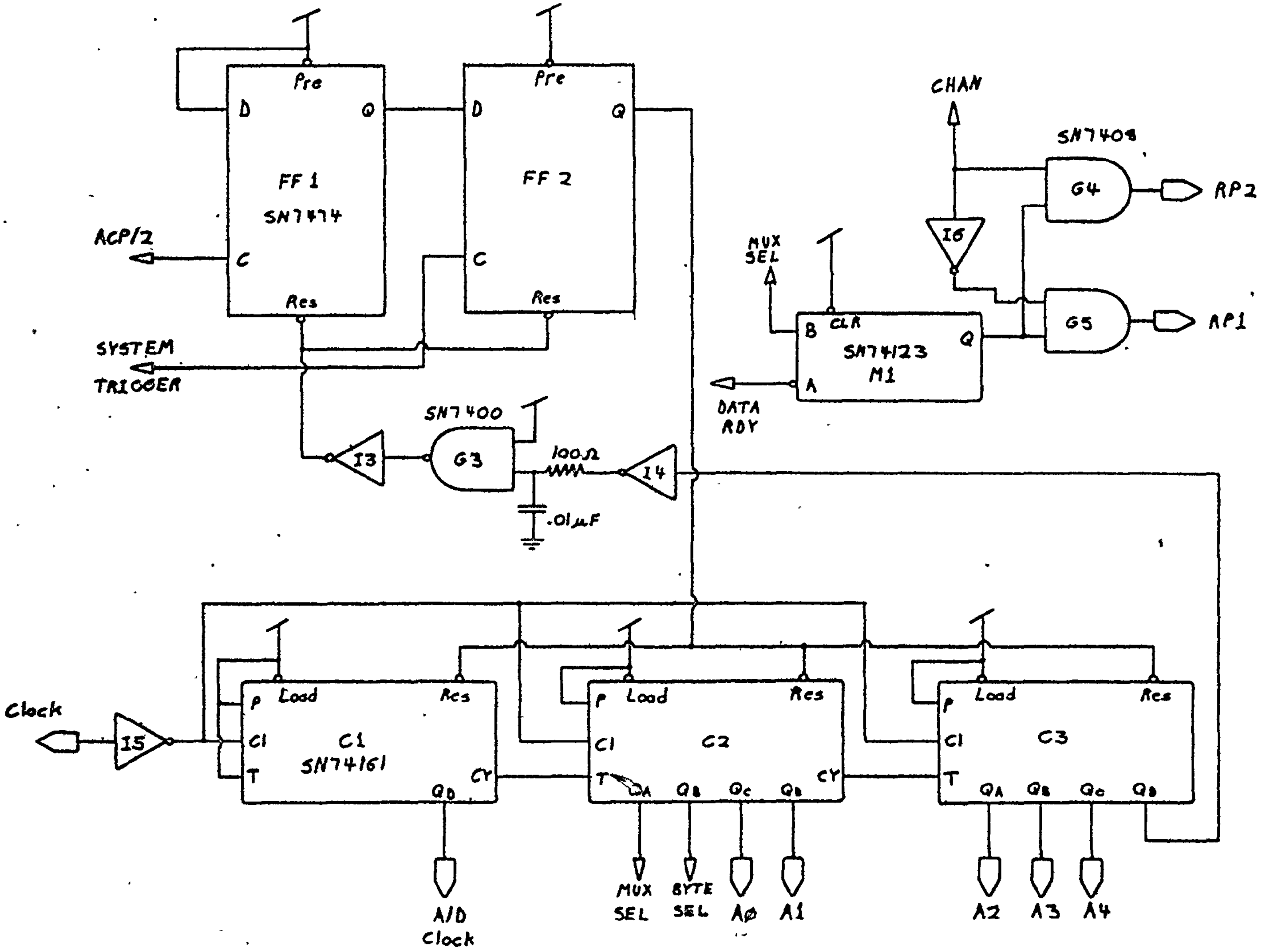


Fig. 4.3 Memory Controller for Range Sampling

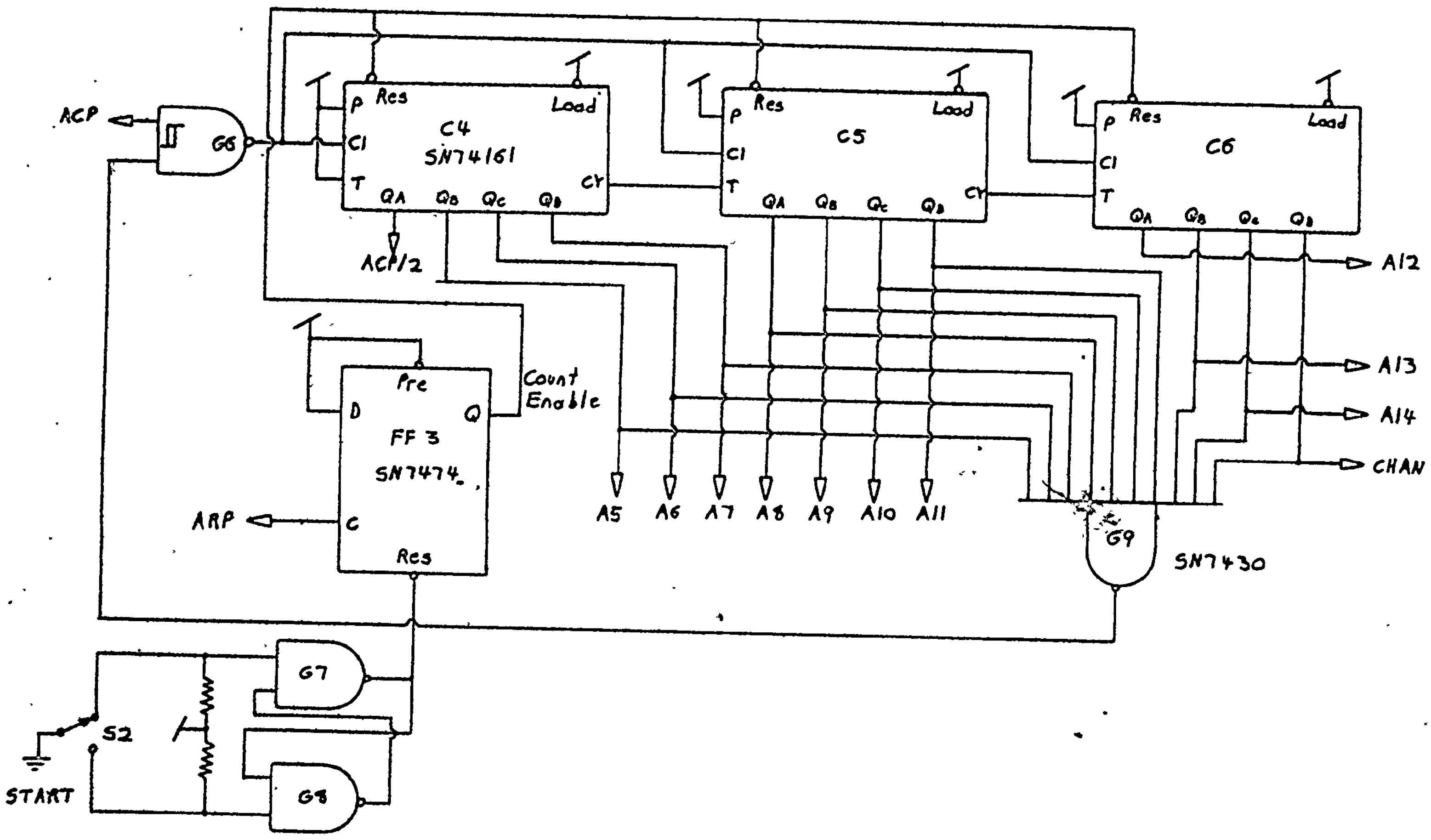


Fig. 4.4 ARP Initiation and ACP Counter

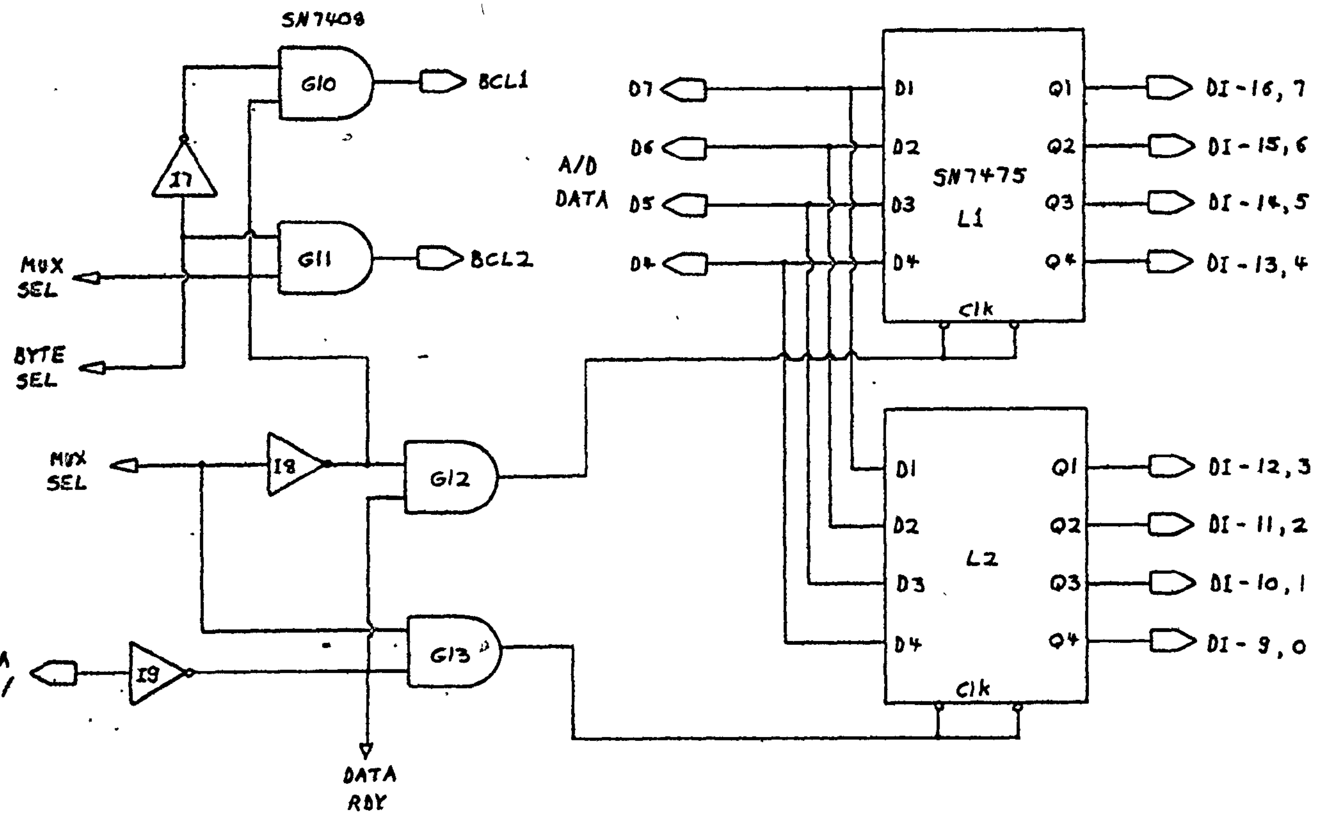


Fig. 4.5 A/D Data Interface to Memory

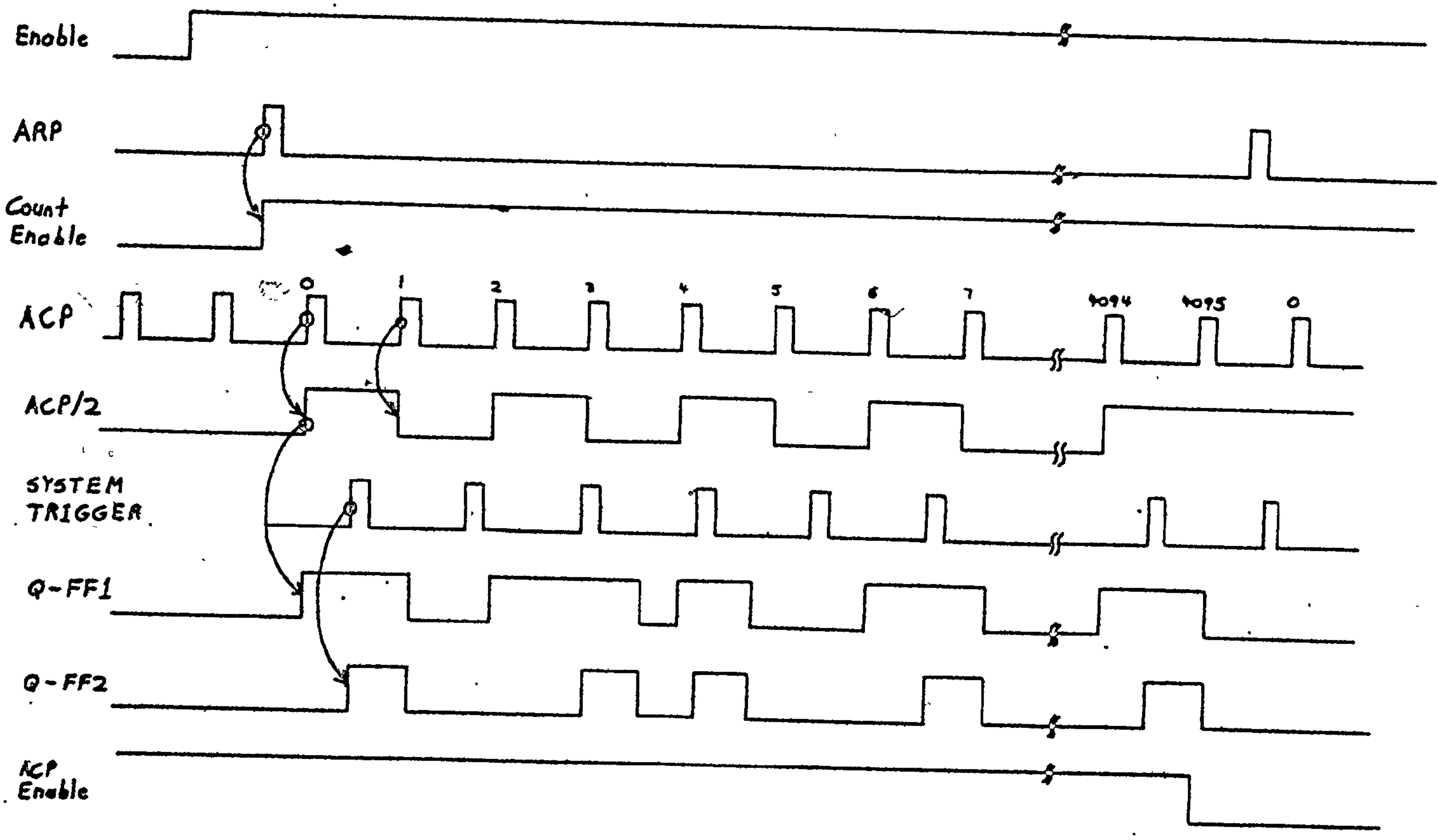


Fig. 4.6 Timing Diagram for ACP Counter

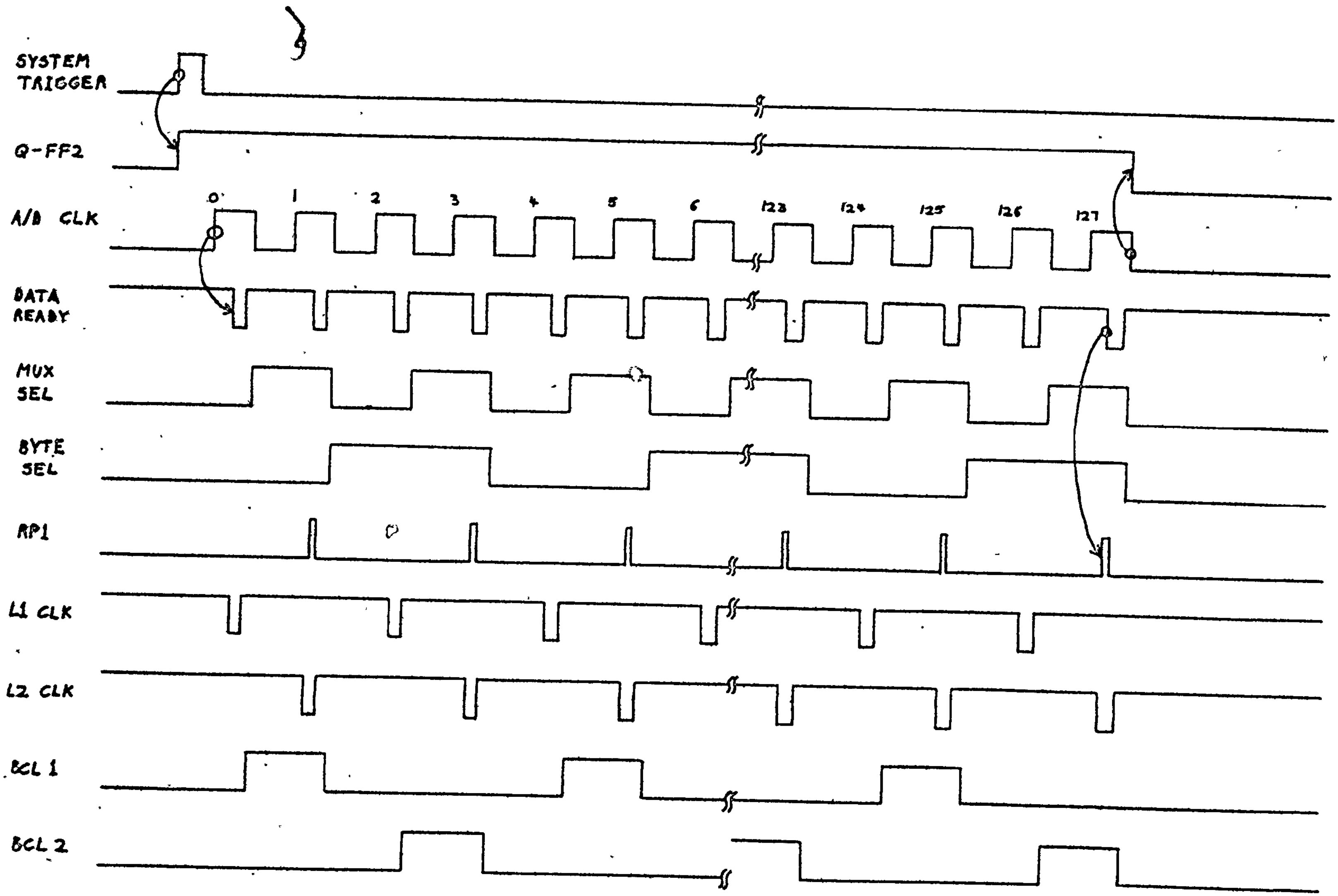


Fig. 4.7 Timing Diagram for Range Sampling

changing cables.

4.2 SCAN CONVERSION TO PPI FORMAT

4.2.1 Introduction

The radar data stored initially in memory is in a polar format. If the memory contents is displayed, a "B-scan" image will result, where the horizontal direction is range and the vertical direction is azimuth. This is shown in Figure A5.47. The data was processed so that the 2048 range sweeps were averaged into 256 line display, each line containing 128 samples in range. It is required to convert the B-scan display into a PPI format, in order to be compatable with the satellite images; a more meaningful format. This result of the scan conversion is Figure A5.48.

The scan conversion is performed in software. The computer performs the scan conversion at a rate of approximately 8 times slower than real time, requiring 80 seconds to complete. The raw radar data in memory is first stored on magnetic tape since the display portion of the memory is required for the production of the PPI display. Then the tape is read back and the data is written to the display in the same manner as a conventional PPI display. At a rate of 9600 baud, the actual conversion process takes 150 seconds. Hence the computer program is waiting for data almost half of the time.

4.2.2 Algorithm

The basic process of scan conversion is to compute the required x,y display address for each input point and then write the data to that address. Since there are more input points then display points, data reduction is required. A considerable simplification is made if the input points are written over if two or more points are close enough to require the same memory location on the display. This occurance is more frequent near the image centre. Otherwise, a complex spatial filtering operation would be required. Near the edges, some memory locations may not be written into at all.

The image centre on the display, (x_o, y_o), is defined as $x_o = 128$, $y_o = 128$. The memory address x,y is computed as

$$x = r \sin A + x_o$$

$$y = -r \cos A + y_o$$

where r is the range of a point; $0 \leq r \leq 127$ and A is the azimuth; $A = 2\pi n/2048$, n is the number of a range scan (sector); $0 \leq n \leq 2047$. On a PPI display, zero azimuth is towards the top. The range of x and y is from 0 to 255.

For any given sector n, a straight line will be drawn from the centre to the edge as r ranges from 0 to 127. The equations for computing x,y can be re-written as

$$x_{r+1} = x_r + \Delta x$$

$$y_{r+1} = y_r + \Delta y\ .$$

where

$$\Delta x = \sin A,\ \Delta y = -\cos A$$

Hence each new address along a sector line is simply generated by addition. Since Δx and Δy are fractional numbers, the addition must be carried out in fixed point format and the numbers rounded to the nearest integer only for setting-up an address. For computation speed and simplicity, sin and cos are generated from a lookup table. Only 2048 discrete values are required.

When the address is computed and set-up, the data is read from the tape and written into that location.

4.2.3 Program Implementation

Figure 4-8 is a flowchart of the scan conversion program. The lookup table consists of 1536 entries, one byte for each entry. This represents 3/4 of a cycle, allowing simple generation of sin and cos values. For negative arguments, the identities $\sin(-A) = -\sin A$ and $\cos(-A) = \cos A$ are employed. A cosine is generated as $\cos A = \sin(A + \pi/2)$. All values are the absolute values in the form of unsigned integers. Program logic generates the sign. The values of x and y are computed using 16-bit addition, with the upper byte as the integer part and the lower byte as the fractional part. The 16-bit integer is unsigned since x and y are always positive.

If the argument for the sin function is A (in radians), then the lookup table address HL is computed as

$$HL = \frac{2048\ A}{2\pi} + TABLE$$

where TABLE is the starting address of the table. For the scan conversion program, the term $2048\ A/2\pi$ is the sector number since the angular increment between adjacent sectors is $2\pi/2048$.

Each byte read from the data tape contains two radar points, each consisting of 4 bits. Hence the program must perform two co-ordinate conversions for each byte read. Appendix 14 is the listing for the scan

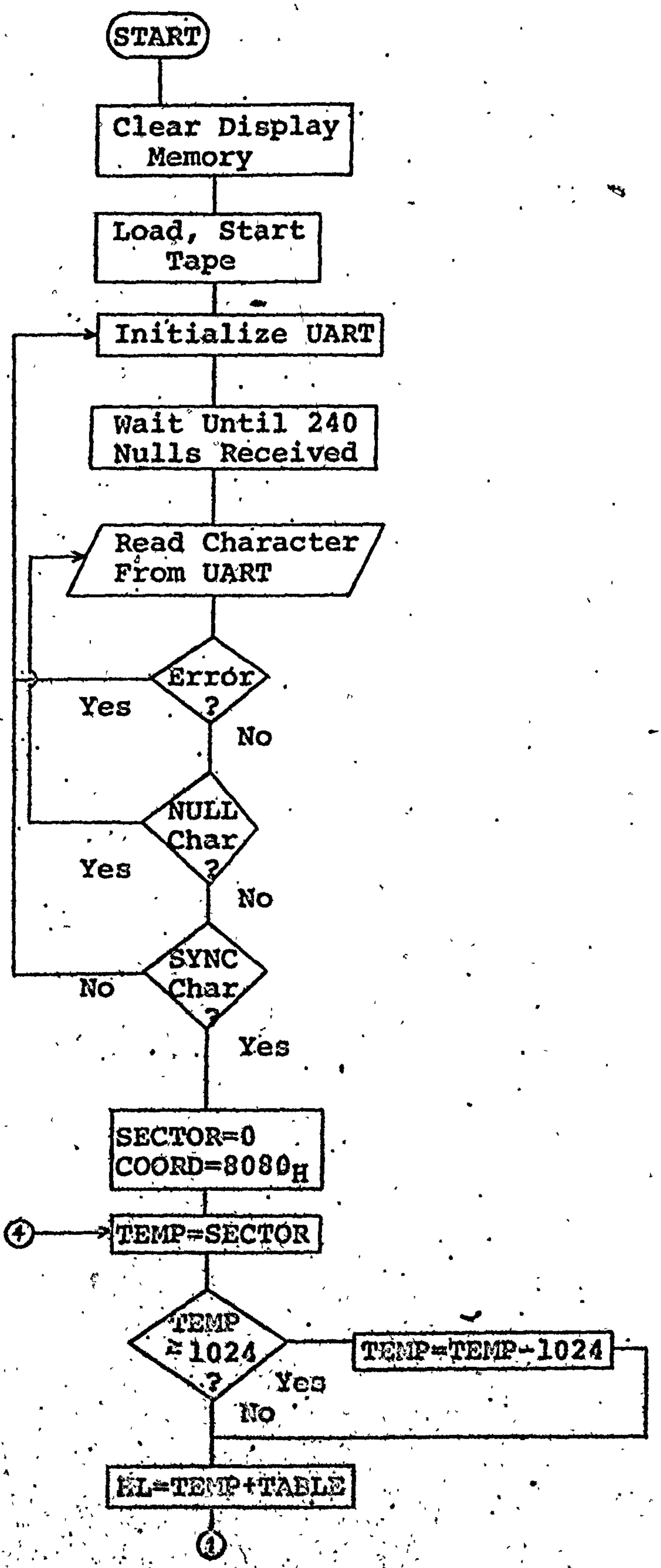


Fig. 4.8 Scan Conversion Program Flowchart

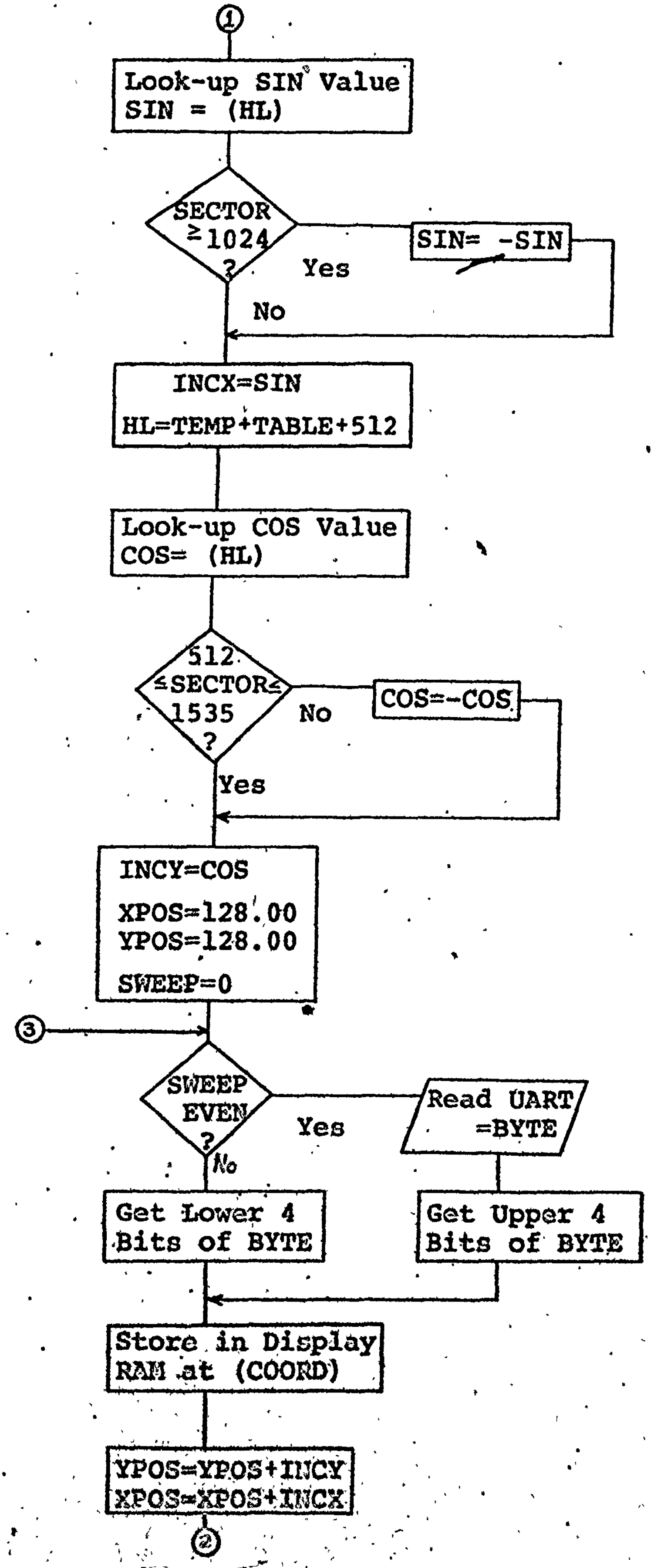


Fig. 4.8 Continued

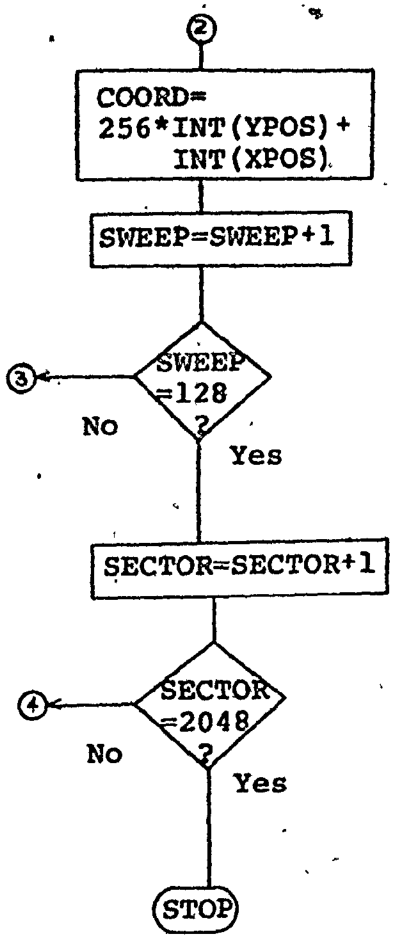


Fig. 4.8 Continued

conversion program. The lookup table was generated by a BASIC program and the table was stored along with the object code for the scan conversion program.

4.3 MECHANICAL-OPTICAL SCANNING OF PHOTOGRAPHS

4.3.1 Scanning Device

A photograph containing data may be digitized by a mechanical-optical scanner. A scanner consists of several components:

(i) a constant speed rotating drum driven by a motor (synchronous)
(ii) high intensity spot lamp
(iii) photodetector, optics and high gain amplifier
(iv) lateral drum feed mechanism and motor
(v) drum phasing contacts

The photograph is mounted on the drum. The drum is rotated by a motor through a worm gear drive and is slowly fed laterally by another motor through a rack and pinion gear drive. A photodetector and optics receive light from a small area on the potograph at any given time, due to its very narrow field of view. This determines the resolution of the system. The light is reflected from the photograph originating from a high intensity light source. The amount of light reflected and hence the signal from the photodetector is proportional to the reflectivity of the point on the photograph. As the drum rotates and is fed laterally, a raster scan signal results. The signal bandwidth and scan line density must match the detector aperture size.

4.3.2 Digitizing PPI Photographs

Due to capstan motor problems on the video recorder, the photographs were used instead. Photographs of the PPI CRT were made during the satellite passes. The photographs corresponding to the satellite passes were digitized via the mechanical-optical scanner and stored in the image memory. The image did not have to be scan-converted since it was already in PPI format.

Figure 4-9 is a block diagram of the photograph digitizing system. The power line frequency of 60 Hz produces a stable drum motor speed of 180 RPM. The phasing contacts are opened momentarily each drum revolution by a cam. This flags the computer when a new line is about to begin. The lateral feed motor is driven by a low frequency power amplifier (10 watts) rather than from the power line. A variable frequency oscillator sets the motor speed (a synchronous motor) and hence the lateral feed rate. The size of the area of interest on the photograph required an oscillator frequency of 50 Hz to produce a square image. A 60 nautical mile radius on the coverage area photograph is contained in square of 2.4 inch sides. This occupies a fraction 0.38 of the drum circumference (2 inch diameter drum). If the digitized image contains 256 by 256 elements and the scanning rate is 3 lines per second, a horizontal element is digitized every .5 m sec. In order to prevent aliasing, the low pass filter cut-off frequency must be less than 1 KHz. A high gain current-to-voltage converter amplifies the phototube video signal to the .5 volt bipolar level required by the A/D converter. The photograph elements are quantized to 16 levels.

4.3.3 Computer Data Acquisition

Unlike the radar signals, the digitized photograph data may be stored in memory under program control, since the sampling rate is 2 KHz. Figure 4-10 is the program flow-chart and Appendix 15 is the program listing. The A/D converter output is connected to the memory

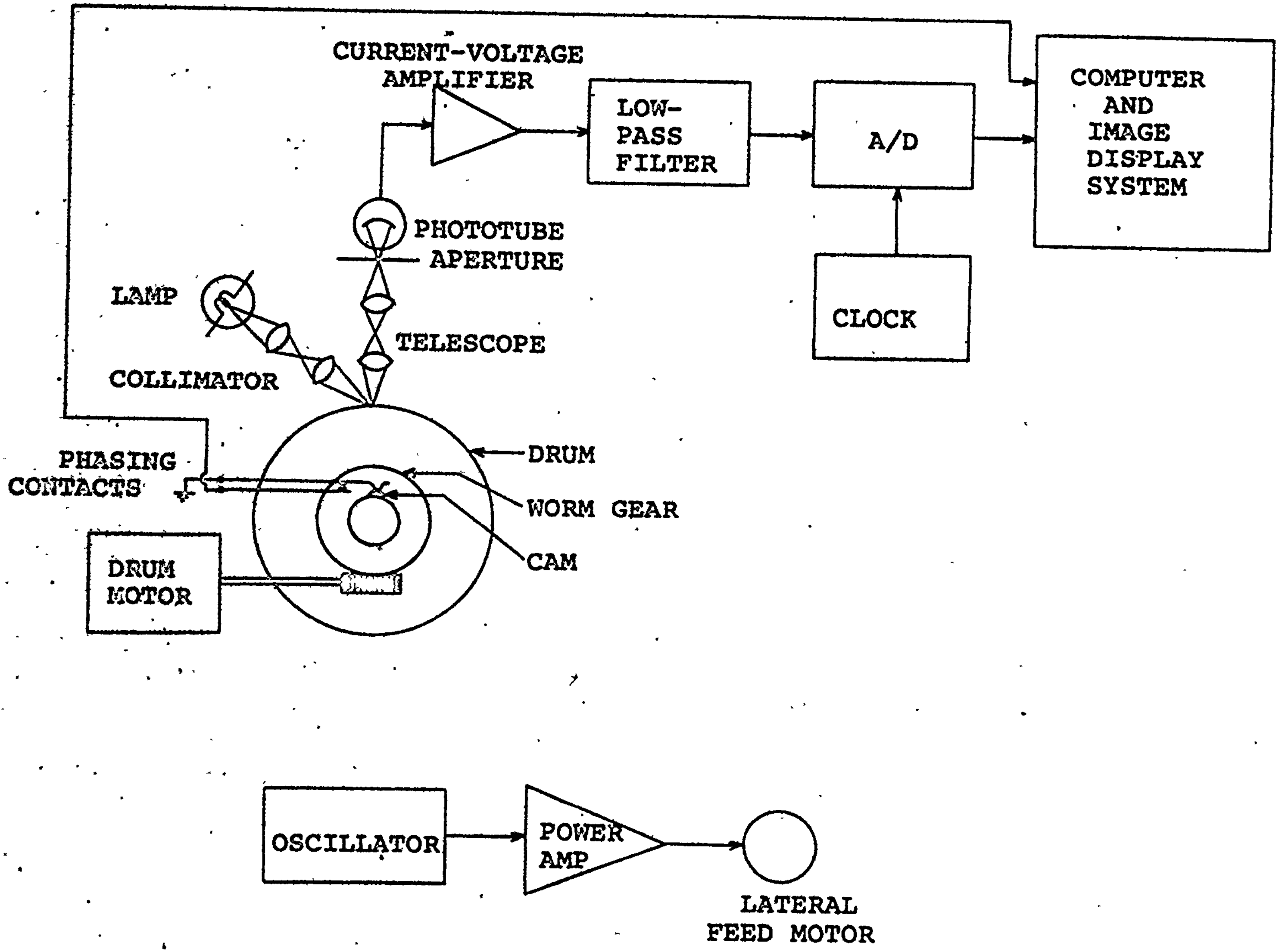


Fig. 4.9 Photograph Digitizing System Block Diagram

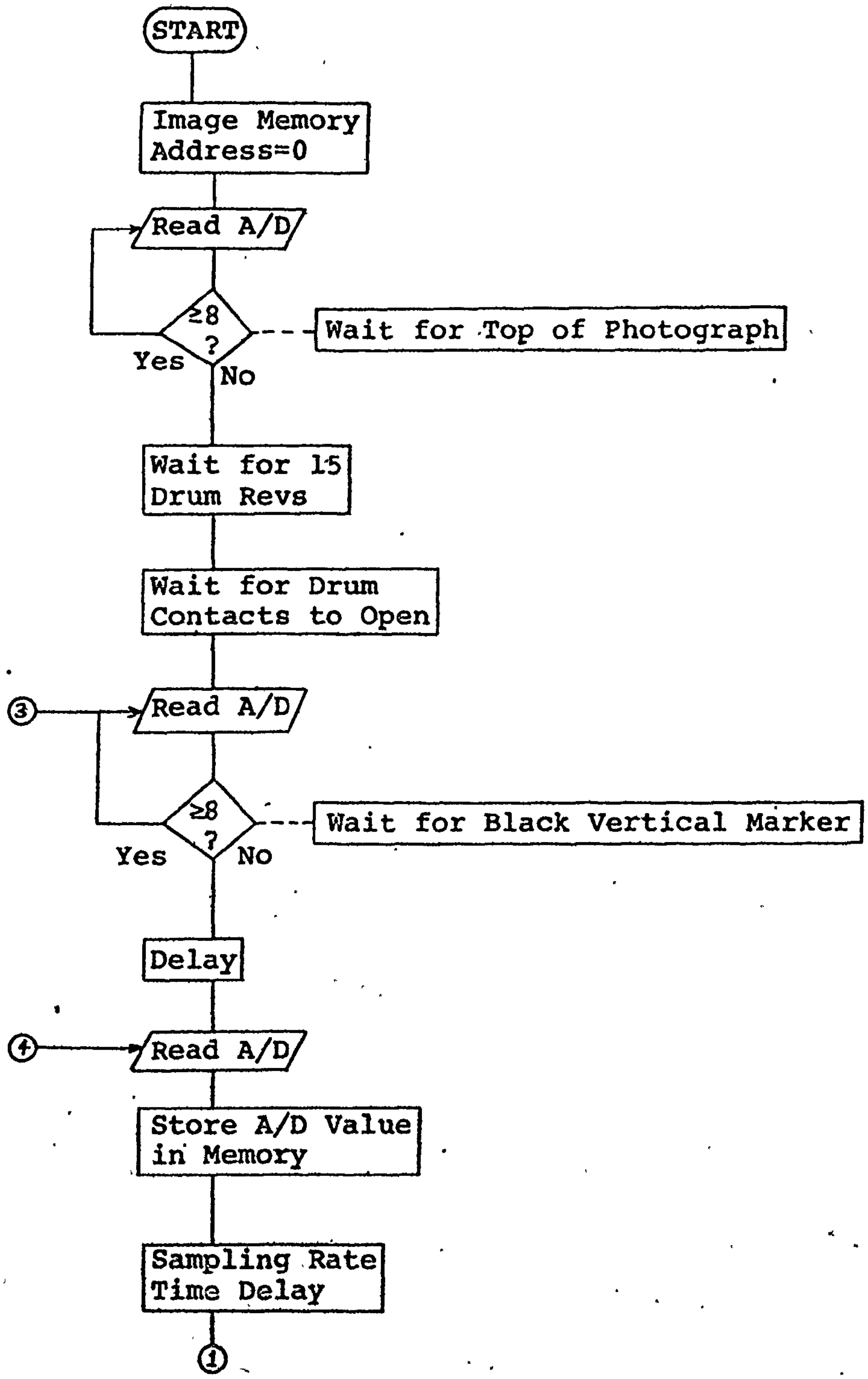


Fig. 4.10 Photgraph Digitizing Program Flowchart

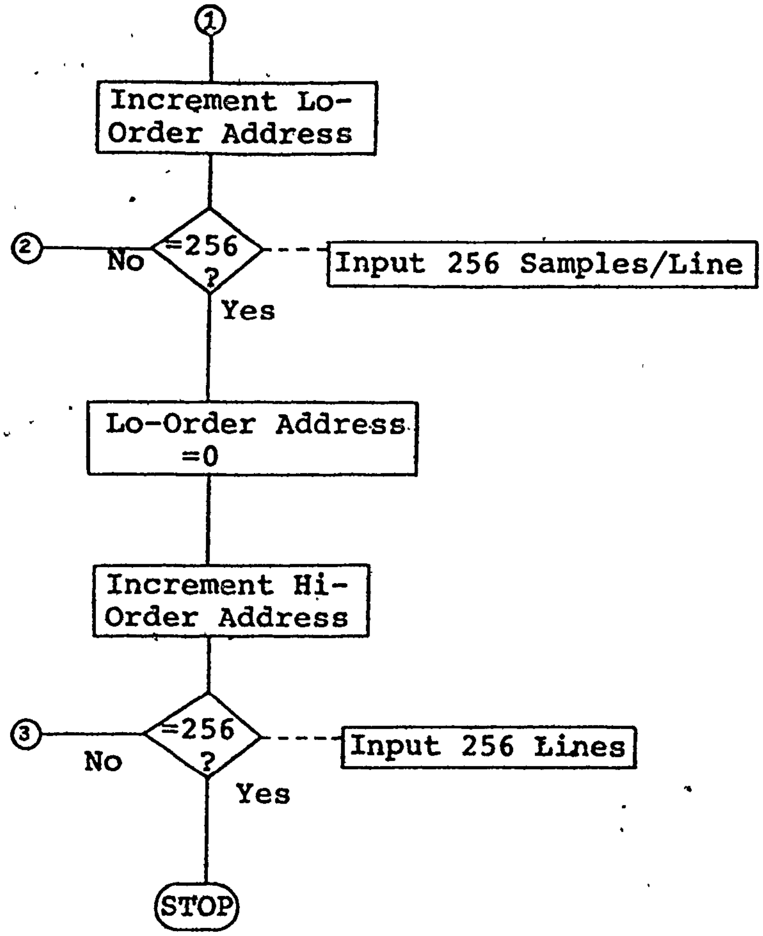


Fig. 4.10 Continued

output cable of channel 1, which is disconnected from the memory. This is not required for the memory since only channel zero is used for this operation.

The photograph was taped to the drum with a white marker band around the top of the photograph. As the lateral feed advances, the A/D converter is sampled, waiting for the white or high video level to change to dark, indicating the end of the marker and the top of the photograph. The marker band can be simply moved to change the starting point of the digitization.

Once the digitization has been initiated, 256 samples are taken every line, until 256 lines have been sampled. The frequency of the lateral feed oscillator determines the feed rate and hence the vertical scale factor. Horizontal synchronization is achieved using the mechanical contacts and a vertical black marker strip down the left edge of the photograph. When the drum rotates towards the beginning of a new line, the contacts open and the flag bit goes high. When the contacts close again and the flag returns to zero, the video signal will be white, from the boarder of the photograph. The video signal is sampled for a low level, indicating the presence of the marker strip.

A delay factor SHIFT determines a software delay time from the marker strip to the start of the actual video sampling of data. The video sampling rate is set by the factor RATE, which sets the time delay between A/D samples. Each sample is written to the image display memory and the address incremented. The program, after sampling 256 points, waits for the next sync flag. RATE determines the horizontal scale factor and SHIFT determines the horizontal position of the photograph.

4.4 GEOMETRIC CORRECTIONS TO PHOTOGRAPH DATA

4.4.1 Geometric Transformations

The digitized photograph image is approximately similar to the satellite image in terms of geometry by not close enough for direct comparison. There are six transformations that must be applied to the photograph image:

(i) x-shift (translation)

(ii) y-shift

(iii) x-scaling

(iv) y-scaling

(v) rotation

(vi) radius shift

Note that the independent x and y-scaling and radius shift cannot be performed optically without special optics. Radius shift is required since the PPI centre does not represent zero range. The radial sweep does not begin until a delay time corresponding to a distance of about 3 nautical miles.

The 20 nmi. range rings were used to measure and compute the geometric corrections required. For each point in the output (transformed) image, the address of the input image point is computed and then the point is mapped into the output image. This is a nearest neighbour technique similar to that used in the scan conversion program. If the address lies outside the image, a zero (black) is used in the output image, except for the centre, where white is used since the ground clutter is white.

If the input image is centre shifted Δx, Δy with respect to the output image centre, then a point in the centered image (corrected for offset) $x_1, y_1 = x_o - \Delta x, \ y_o - \Delta y$ where x_o, y_o is the co-ordinates of the input image. A point in a "square" image $x_2, y_2 = x_1, \ y_1/k = (x_o - \Delta x), (y_o - \Delta y)/k$ where k is a scale factor such that y-axis has the same scale as the x-axis. In polar co-ordinates

$$r_2, \ \theta_2 = \sqrt{x_2^2 + y_2^2}, \quad \text{ATAN2} \ (y_2/x_2)$$

where ATAN2 is a form of an arctangent function which gives non-ambiguous angles for any value of x and y. A rotated image

$$r_2, \ \theta_3 = r_2, \ \theta_2 + \Delta\theta$$

where $\Delta\theta$ is the required rotation, positive values of $\Delta\theta$ correspond to a counter clockwise rotation. Compensating for radius offset

$$r_3, \ \theta_3 = r_2 + \Delta r, \ \theta_2 + \Delta\theta$$

where Δr is the radius (in terms of the square image scale) corresponding to the radar centre point. In rectangular co-ordinates

$$x_3, \ y_3 = r_3 \cos\theta_3, \ r_3 \sin\theta_3$$

Finally, scaling the image to the desired size,

$$x_4, y_4 = k_x x_4, \ k_y y_3$$

The process is reversed in order to compute the co-ordinates x_o, y_o in the input image for each point x_4, y_4 in the output corrected image. Given x_4, y_4,

$$x_3, y_3 = x_4/k_x,\ y_4/k_y$$

~~r_3, θ_3~~ $= \sqrt{x_2^2 + y_3^2},\ \text{ATAN2}\ (y_3/x_3)$

$$x_2,\ y_2 = r_2 \cos\theta_2,\ r_2 \sin\theta_2$$

$$x_1, y_1 = x_2,\ ky_2$$

$$x_o, y_o = x_1 + \Delta x,\ y_1 + \Delta y$$

In general, the co-ordinates x_o, y_o will not be an integer, even if x_4, y_4, Δx, and Δy are integers. It is necessary to round x_o, y_o to the nearest integer since it will form the address of the input image, and x_4, y_4 will form the address of the output image. This nearest neighbour technique, as previously discussed, does not attempt to estimate the value of a desired image point located between actual sample points.

4.4.2 Computation of Correction Factors

The range rings on the PPI photographs were used to determine all of the correction factors except $\Delta\theta$, which was determined from the FORTRAN orbit analysis program. When the image of a PPI photograph was displayed on the display system, a moveable black or white dot was used as a cursor, which could be located over a point of interest (such as a range ring). Its address (x,y) could be displayed on the console.

For the three range rings visible (60 nmi. range), the co-ordinates (address) of four points on each range ring were determined. These points were at the top, bottom, left, and right extremes of a range ring. From this data, the centre point is easily computed. For the radar image corresponding to satellite orbit pass 8001, the centre was located at 124,132. If the centre of the image display is 128,128, then $\Delta x = -4$ and $\Delta y = 4$. The spacing between the range rings was greater on the y-axis, by a factor of k = 1.08887. As seen in Table 3.7, the computer printout for the grid correction factors for orbit 8001, $\Delta\theta$ = -23.46 degrees. By noting the spacing between range rings and comparing it to the distance from the centre to the first range ring (which is less), Δr = 6. In order to achieve the same scale as the satellite image, (0.4754 nmi./element) k_x = 0.390807, k_y = 1.00606.

4.4.3 Program Implementation

The computer was programmed to perform the geometric corrections on a radar image so that the resulting image could then be compared to the satellite image. Due to the nature of the computation required, the program was written in BASIC, which like FORTRAN, contains trigonometric functions in the form of intrinsic functions as well as complete floating point arithmetic. BASIC, the most popular high-level language run on microcomputers, is an interpreter and hence suffers from a speed deficiency. Table 4.1 is the BASIC program. This program requires on the order of 18 hours to run, where each image point is transformed in about 1 second. If the algorithm were implemented using assembly language coding, at least an order of magnitude of speed improvement could be achieved. However, the program would be long and not practical since only two images were transformed.

The transformation parameters are contained in lines 40, 50, 60, 70 and 80. The values shown in the listing are for orbit 8001. Only the parameters $\Delta\theta$, Δx, and Δy change for similar conditions for taking and digitizing the PPI photographs. Note that the parameters k_x, k_y, k were fairly close to unity and Δx and Δy more fairly close to zero. Except for the rotation, the geometric corrections were small since the digitization parameters were close to the correct values as possible.

```
1 REM PROGRAM FOR IMAGE TRANSLATION, MAGNIFICATION,
2 REM ROTATION, AND EXPANSION FROM CENTRE
3 REM IMAGE CAN BE TRANSLATED AND MAGNIFIED IN
4 REM BOTH X AND Y AXIS
5 REM INPUT IMAGE IN CHAN 1
6 REM TRANSFORMED IMAGE IN CHAN 0 OF MEMORY
7 REM PROGRAM CORRECTS DIGITIZED PPI PHOTOGRAPHS
8 REM FOR SATELLITE OVERLAY
9 REM TRANSFORMATION USES NEAREST NEIGHBOUR TECHIQUE
10 P1=3.14159
20 P2=P1/2
30 R0=180/P1
35 REM K2=Y SCALE FACTOR
36 REM K1=X SCALE FACTOR
40 K2=1.00606 :K1=.930807
45 REM D1=X TRANSLATION DISTANCE
46 REM D2=Y TRANSLATION DISTANCE
50 D1=-4 :D2=4
55 REM D3=RANGE (NORMALIZED) THAT PPI SCAN
56 REM STARTS FROM CENTRE
60 D3=6
65 REM D4= ROTATION ANGLE (DEGREES)
70 D4=-23.46/R0
75 REM K=PPI PHOTOGRAPH X,Y SCALE CORRECTION FACTOR
80 K=1.08887
100 C0=128
110 D1=D1+C0
120 D2=C0-D2
130 C2=.5
140 C1=1
150 C3=255
160 C4=4
170 H=129
180 L=130
190 D=131
195 REM COMPUTE ADDRESS OF INPUT IMAGE CORRESPONDING
196 REM TO I,J POINT IN TRANSFORMED IMAGE
200 FOR J=-128 TO 127
210 Y=J/K2
220 S=Y*Y
230 J3=127-J
```

Table 4.1   Geometric Transformation Program

```
240 FØR I=-128 TØ 127
250 X=I/K1
260 R=SQR(X*X+S)-D3
270 IF R<0 THEN 3000
275 REM COMPUTE ATAN2 ØF (Y,X)
280 IF X=0 THEN 1000
290 T=ATN(Y/X)
300 IF X<0 THEN T=T+P1*SGN(Y)
310 IF Y=0 AND X<0 THEN T=P1
500 T=T-D4
510 X=R*CØS(T)+D1
520 Y1=R*SIN(T)*K+D2
525 REM RØUND X TØ NEAREST INTEGER
530 N=INT(X) :F=X-N :X=N
540 IF F>C2 THEN X=X+C1
545 REM RØUND Y TØ NEAREST INTEGER
550 N=INT(Y1) :F=Y1-N :Y1=N
560 IF F>C2 THEN Y1=Y1+C1
565 Y1=C3-Y1
567 REM TEST IF CØMPUTED ADDRESS ØUT ØF RANGE
570 IF X<0 ØR X>C3 THEN 2000
580 IF Y1<0 ØR Y1>C3 THEN 2000
585 REM SET UP ADDRESS AND READ INPUT PØINT
590 ØUT C0,C4
600 ØUT H,Y1 :ØUT L,X
610 S1=INP(D) :S2=INP(D) :S3=INP(D)
620 IF S1<>S2 ØR S1<>S3 THEN 610
630 REM WRITE INPUT PØINT TØ TRANSFØRMED PØINT
700 ØUT C0,0
710 ØUT H,J3 ;ØUT L,I+C0
720 ØUT D,S1 :ØUT D,S1 :ØUT D,S1
730 NEXT I
740 NEXT J
750 END
1000 T=P2*SGN(Y)
1010 GØTØ 500
2000 S1=0
2010 GØTØ 700
3000 S1=C3
3010 GØTØ 700
9999 END
```

Table 4.1 Continued

SECTION 5

PROCESSING AND ANALYSIS OF SATELLITE AND RADAR IMAGES

5.1 PRE-PROCESSING AND ENHANCEMENT OF SATELLITE IMAGES

5.1.1 Linear Contrast Enhancement

The satellite image data has considerable dynamic range due to the sensor requirements. The infra-red and visual sensors must accommodate the total range of temperature and illumination conditions on the earth. In a given image, covering a small section of the earth, the dynamic range is normally smaller than the total possible range, particularly for infra-red images. Local variations are smaller than global variations. On display systems, where the available dynamic range is considerably less, the image will appear too-dark, too-light or "flat". Of the 16 levels available in a 4-bit display systems, only a few levels will be used by most images, if only the four most significant bits of the raw image data are used. Fig. 5.1 is a video level histogram of 65536 samples for a typical satellite image. It is divided into 256 amplitude "bins". Over the image, the binary count ranges from 119 to 181, which occupies 24.3% of the total possible range. If a 16-level display system is used, only 4 levels would be used.

Dynamic range compression effectively shifts and expands the histogram horizontal axis so that the minimum count of 119 is represented by level 0 on the display and 181 is represented by level 15. Brightness and contrast controls on a television monitor perform this function for analog video. A linear transfer function, $y_i = ax_i+b$, can be applied to the image data. x_i is an input sample and y_i is the scaled output, which ranges from 0 to 15. a is the gain or expansion

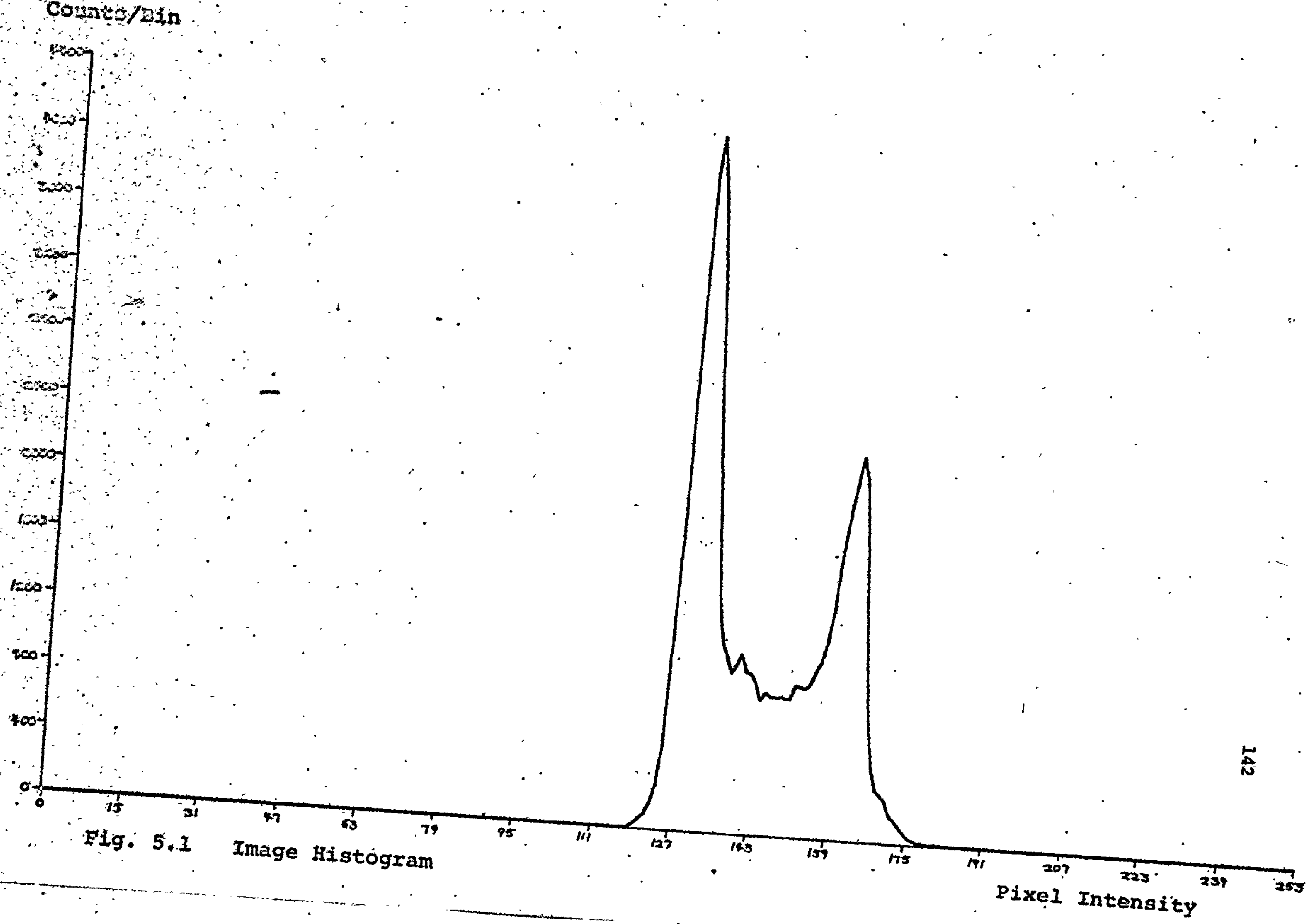


Fig. 5.1 Image Histogram

factor and b is the shift factor. The parameters a and b can be simply computed:

$$y_{max} = a\,x_{max} + b$$

$$y_{min} = a\,x_{min} + b$$

$$a = (y_{max} - y_{min})/(x_{max} - x_{min})$$

$$b = y_{min}$$

For an N-bit display system, $y_{min} = 0$ and $y_{max} = 2^N-1$. Although a and b can be automatically computed, a and b are often adjusted so that image detail is maximized, at the expense of "washing and blackening-out" some parts of the image.

Appendix 11 is a program for linear contrast enhancement. The raw, unprocessed image is stored in channel 1. Locations GAIN and SHIFT are contrast and level parameters. The adjusted value $y_i = (x_i$ - SHIFT) • GAIN/256, where x_i has a value between 0 and 255 and y_i between 0 and 15. If y_i is to be set to the four most significant bits of x_i, SHIFT=0 and GAIN=16. Both SHIFT and GAIN have a value between 0 and 255.

Fig. 5.2 is a flowchart of the algorithm. The above equation is used to form the transfer function look-up table for all 256 x_i values, which contains 256 4-bit y_i values. For each image element x_i, y_i is obtained by reading the table location with address TABLE + x_i or y_i = [TABLE + x_i]. If the look-up table technique was not used, y_i would have to be computed 65536 times, using the transfer function equation, compared to 256 times to generate the look-up table. If y_i is computed to be negative or greater than 15, y_i is "limited" to 0 or 15 respectively.

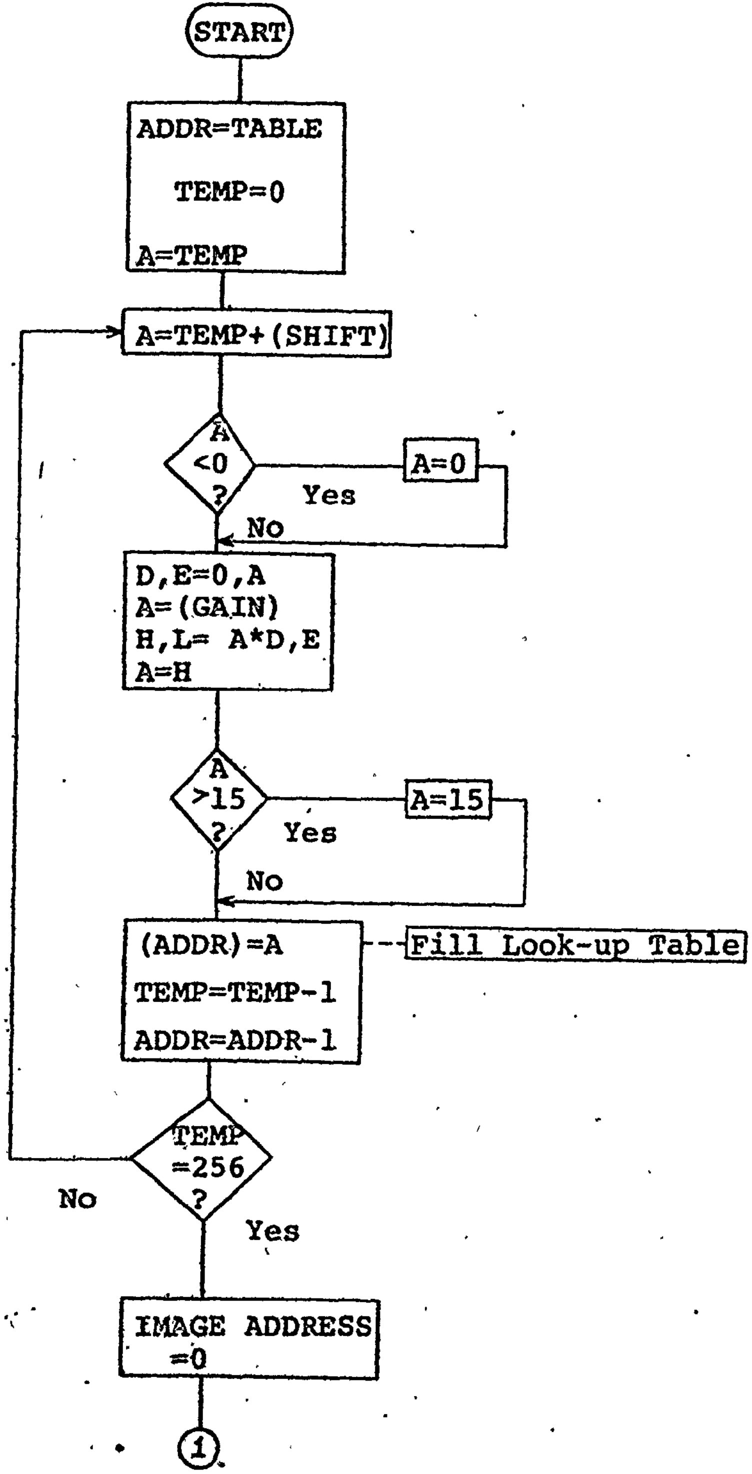


Fig. 5.2 Flowchart for Linear Contrast Enhancement

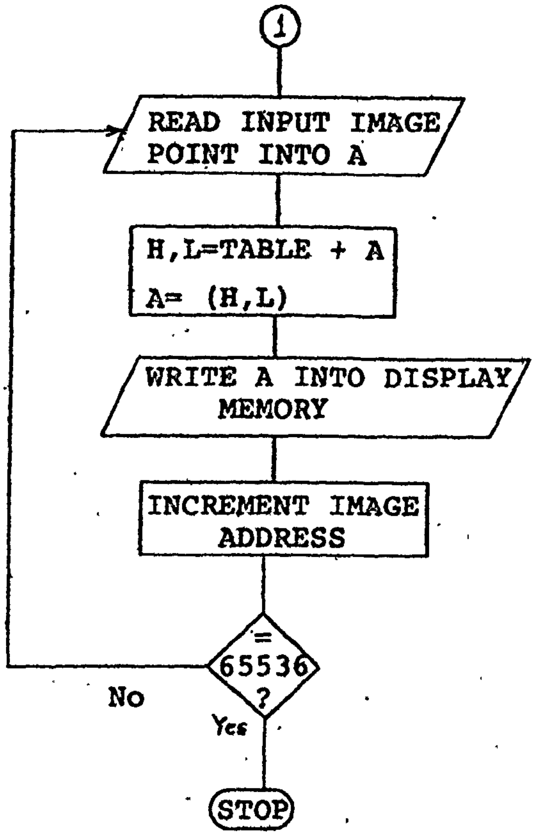


Fig. 5.2 Continued

5.1.2 Histogram Flattening

As shown in Fig. 5.1, the image histogram has two major peaks. The peaks at 131 is due to the surface temperature and the peak at 163 is due to clouds. The surface and cloud temperatures are fairly constant within themselves and hence the image may be classified into two major classes of objects. In a typical infrared image, land and water temperature is uniform over the image, more so than clouds. Different cloud types with different heights will have different temperatures.

The histogram flattening algorithm enhances details by assigning any given display level with an equal number of picture elements. In essence, the image histogram is "flat". This prevents picture elements from clustering about a few levels. Depending on the particular image, detail may be enhanced to be positive or negative degree in various sections of the image. Examples of histogram flattening as shown in Appendix 5. In all cases, this algorithm automatically compresses the dynamic range of the raw image to be suitable for the display.

The histogram flattening algorithm uses the cumulative distribution as the transfer function. The cumulative distribution is the integral of the histogram. It is monotonically increasing, as the linear transfer function, but is generally non-linear. The resulting image will not have an absolutely flat histogram, particularly if the input histogram has large slopes.

Appendix 11 is a program for enhancing an image using the hisstogram flattening algorithm. Fig. 5.3 is the corresponding flowchart. The histogram of the input image is generated which is then replaced by the cumulative distribution. The values of the cumulative

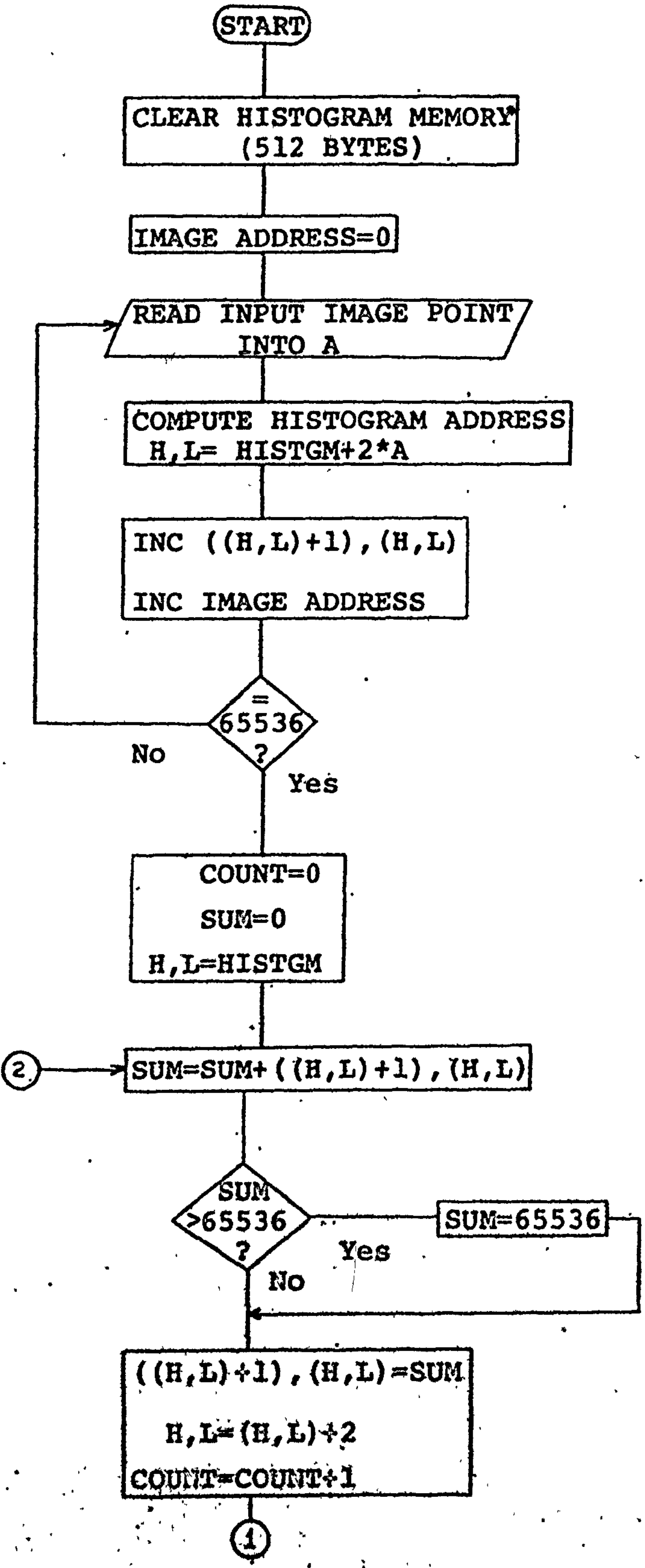


Fig. 5.3 Flowchart for Histogram Flattening

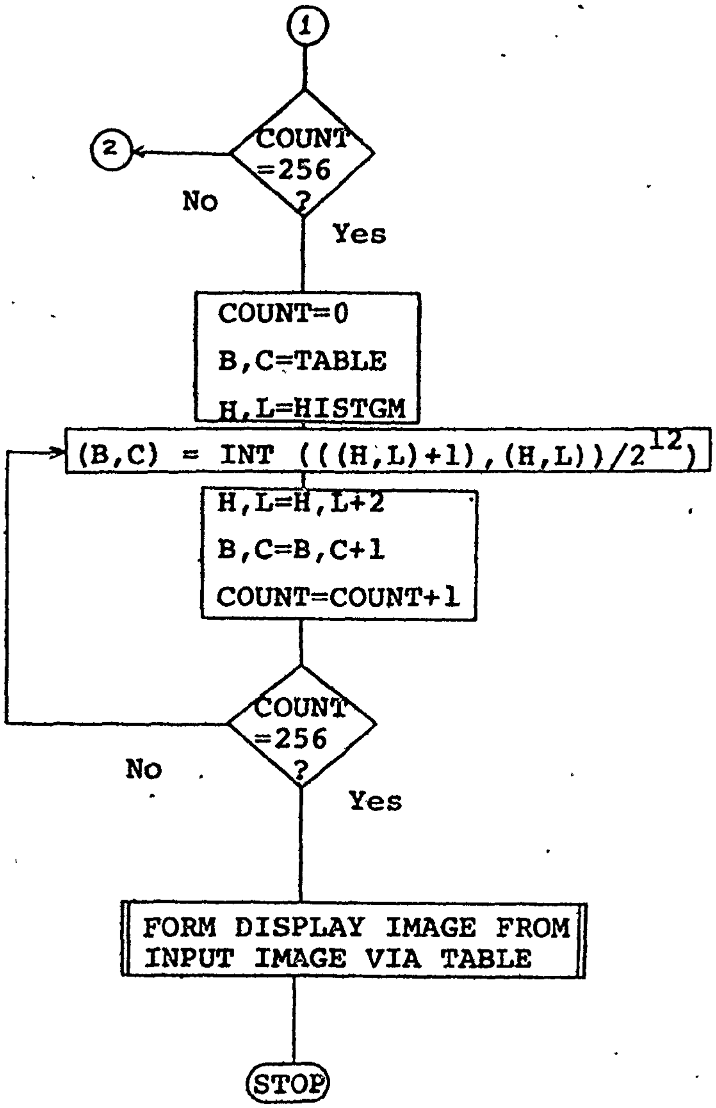


Fig. 5.3 Continued

distribution are truncated to 4-bit values and is used in the same manner as the look-up table in the linear enhancement program.

5.2 COMPARISON OF SATELLITE AND RADAR IMAGES

5.2.1 Contouring

In order to compare clouds with echos, the two images must be combined in some way to show both the similarities and differences between the two. Simple addition of the two images will display a confusing result. The most common technique used in comparing features of images is to draw contour lines from one image over the other image. In Fig. A5.14 and A5.39, contours for a particular temperature (representing cloud boundaries) have been overlaid on the radar image as black lines. Areas above the threshold are shaded in grey. The cloud areas can be seen without degrading the clarity of the radar image. The contour lines and shading are best visible in colour. The shading is achieved by filling in the cloud areas with black dots at even columns on even lines and odd columns on odd lines. One half of the radar image elements are destroyed, resulting in an apparent brightness loss. Very small echos may be lost if they occupy only a few elements. For maximum clarity, any black elements must be first changed to the next level, purple, so that the black level is reserved for the thresholded overlay.

Appendix 12 is a program for performing the threshold and contour overlay. Fig. 5.4 is the corresponding flowchart. The location LEVEL is set to the cloud threshold, with a range from 0 to 255. Some clouds, particularly thin cirrus, may have an apparent temperature or brightness near that of land or water and hence will not have contours drawn around them. However, these clouds usually do not have

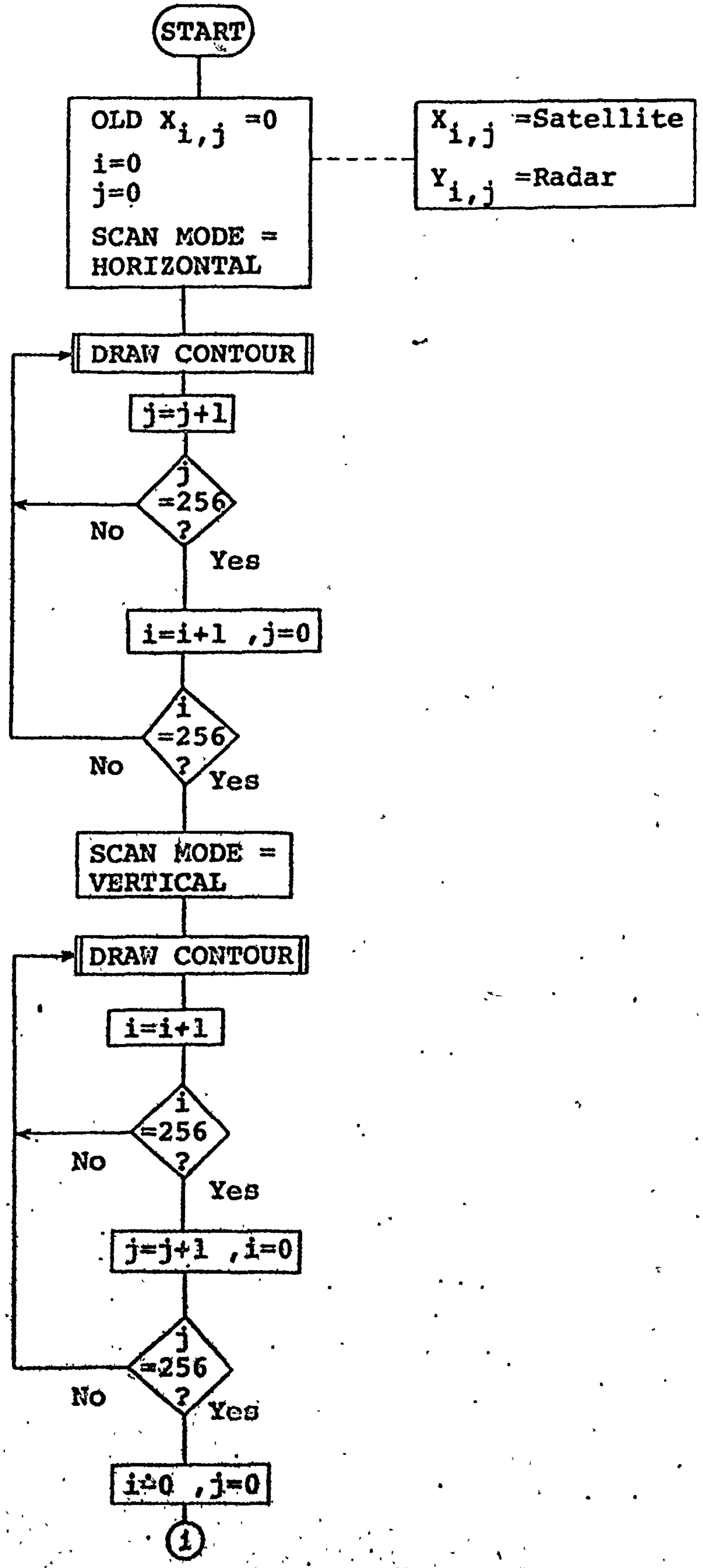


Fig. 5.4 Flowchart for Thresholding and Contour Overlay

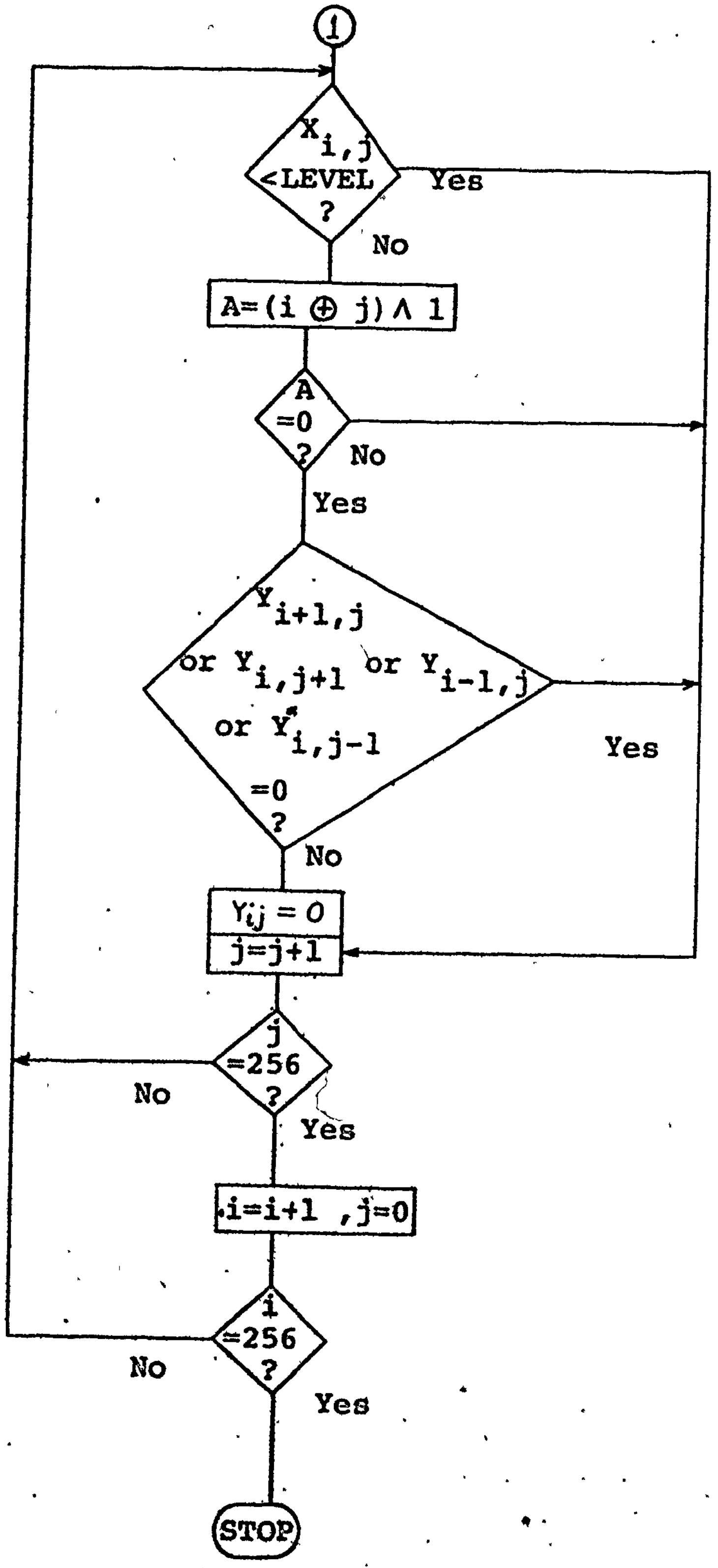


Fig. 5.4 Continued

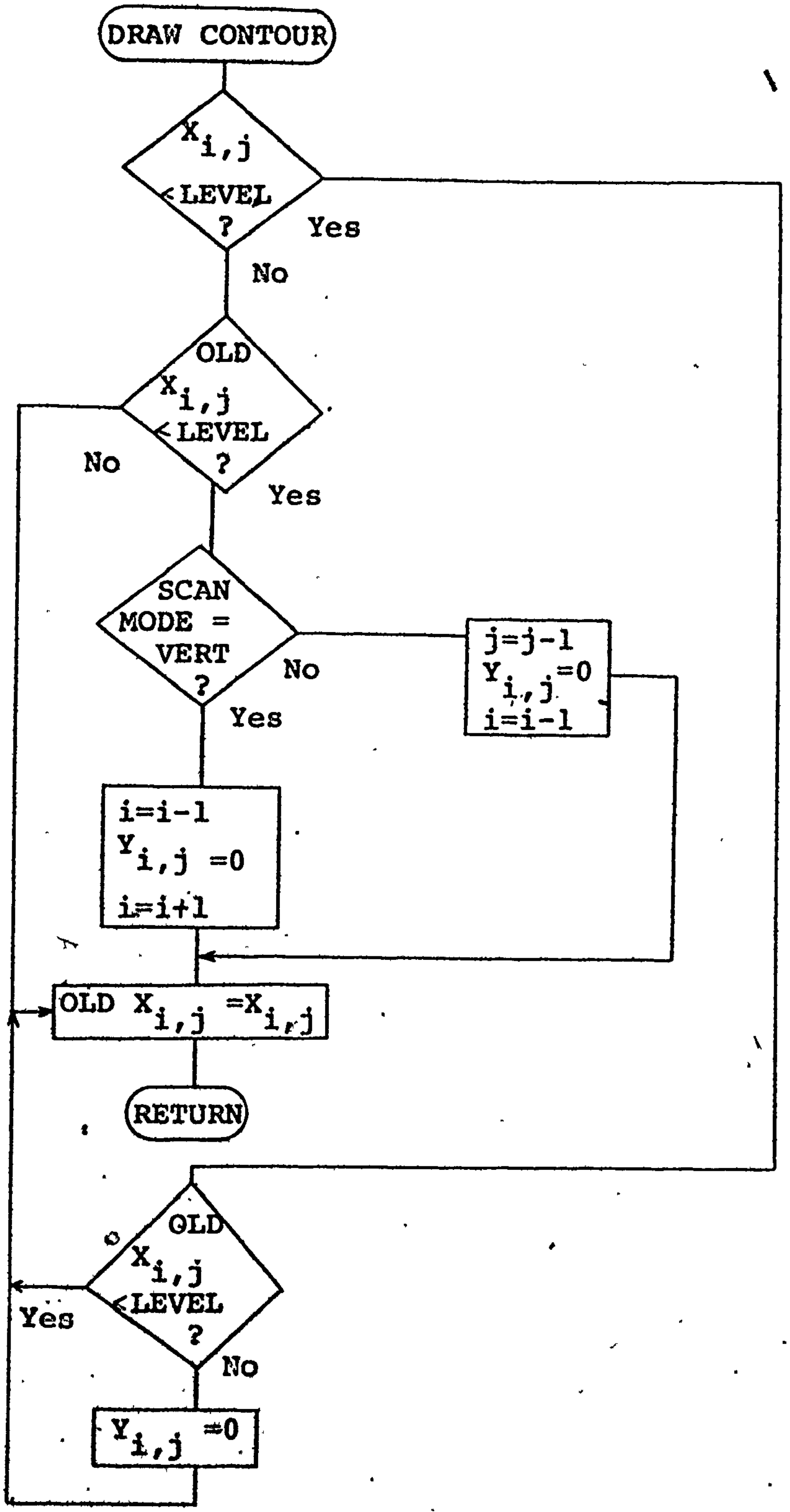


Fig. 5.4 Continued

corresponding echos since they contain little water.

Appendix 12 is a program from producing contours of radar echos over a satellite image. The result is shown in Fig. A5.12 and A5.38. The procedure is similar to the previous program. This comparison method is generally superior to the previous one since radar echos are isolated easier than clouds. The main problem is ground clutter. Although MTI processing will eliminate ground clutter, it usually eliminates slow-moving clouds as well.

5.2.2 Thresholding

Fig. A5.40 is the result of thresholding both the satellite and radar images and assigning a colour to each of the following conditions:

DARK BLUE = No cloud and no echo
LIGHT BLUE = Cloud and no echo
LIGHT PINK = Cloud and echo
RED = No cloud and echo

Appendix 12 and Fig. 5.5 is a program for the performing the thresholding. The value of CLOUD and TARG set the threshold levels for cloud and echo discrimination. The value of CLOUD can range from 0 to 255 and TARG can range from 0 to 15. If the four colours are chosen carefully, similarities and differences between clouds and radar echos may be seen. The main disadvantage of this method is that some "faint" clouds may not be visible since they fall below the threshold level. If the cloud threshold level is lowered, land and water areas will be incorrectly classified as clouds.

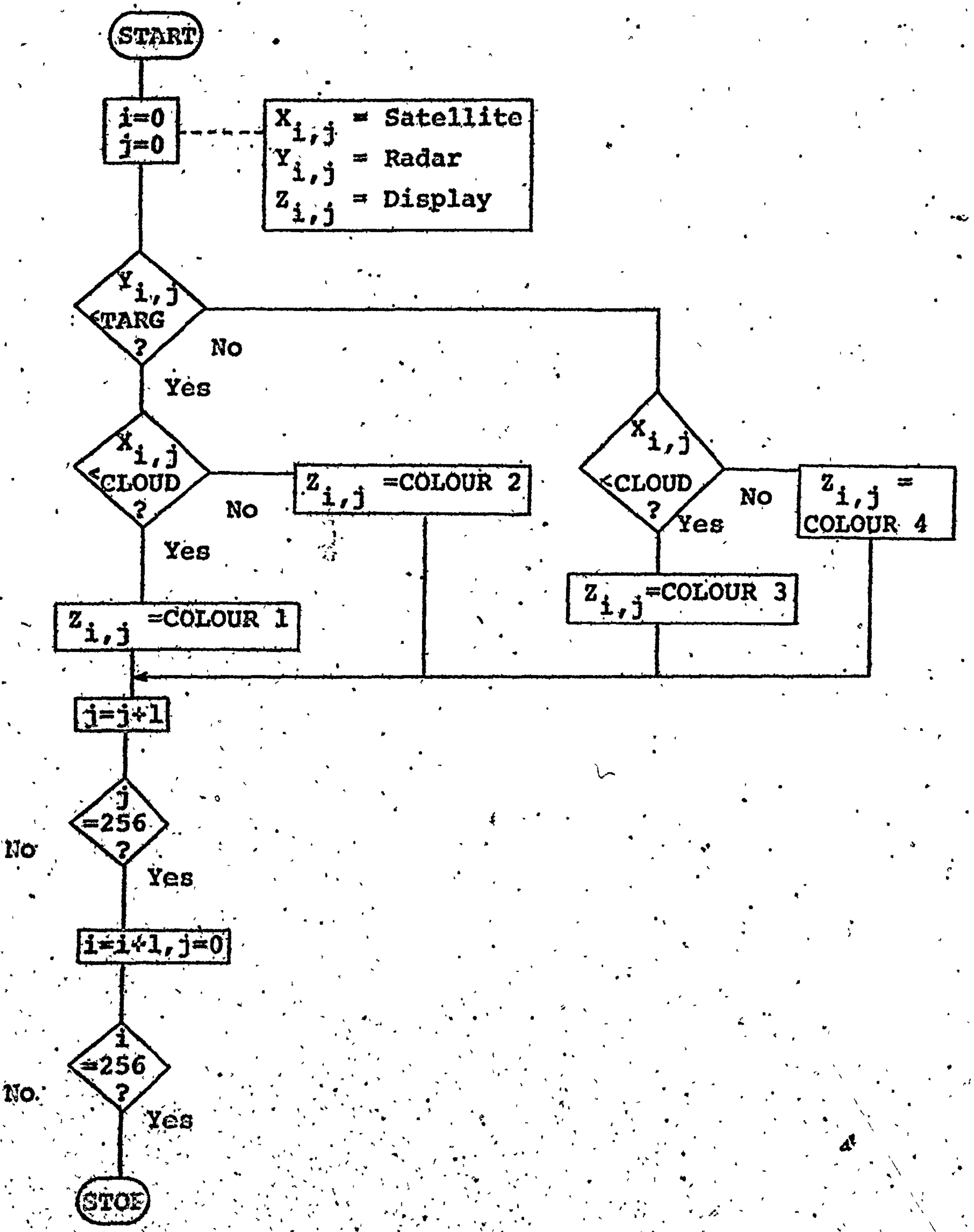


Fig. 5.5 Flowchart for Thresholding

5.2.3 Contouring by Gradient

Another technique of isolating objects is by edge detection rather than a simple level detection. An edge has a higher gradient than the background. The gradient for an image (two-dimensional) is

$$\text{grad } f = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}$$

Of interest is the magnitude of the gradient for the discrete case

$$|\text{grad } x \ | = \left[\left(\frac{\Delta x_{ij}}{\Delta i}\right)^2 + \left(\frac{\Delta x_{ij}}{\Delta j}\right)^2 \right]^{1/2}$$

This may be approximated by

$$|\text{grad } x_{ij}| \approx ((x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2)^{1/2}$$

Using a further approximation

$$|\text{grad } x_{ij}| \approx |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|$$

This approximation has a maximum error of 41% when $|x_{i+1,j} - x_{i,j}| = |x_{i,j+1} - x_{i,j}|$. (1.5 dB) The computational simplicity and speed makes it attractive. This technique was used in Fig. A15.5, where the element intensity is proportional to the magnitude of the gradient. Although clouds are outlined, this gradient method is not reliable enough to allow automatic cloud detection solely on the basis of the magnitude of the gradient being greater than some fixed value. Clouds not having sharp edges or faint clouds will not produce a high gradient.

5.2.4 Cross-Correlation

The conventional technique of measuring similarity between two signals is cross-correlation. For the two-dimensional discrete case,

$$R(i,j) = \sum_{y=1}^{N} \sum_{x=1}^{N} S(x,y) * T(x-i, y-j)$$

where S is input image 1

and T is input image 2

x and y are the element column and row addresses.

Since S and T are both unipolar images, the mean $\bar{S}$ and $\bar{T}$ should be subtracted or $R(i,j)$ will contain a bias component which may be very large, particularly if the correlation between the images is small. An unbiased $R_o(i,j)$ will require fewer bits to represent it. Due to memory restrictions, S and T will be restricted to 4 bits per element and $R_o(i,j)$ will be assigned 8 bits.

Decomposing S and T into an unbiased and mean components

$$R(i,j) = \sum_{y=1}^{N} \sum_{x=1}^{N} (S_o(x,y) + \bar{S}) \cdot (T_o(x-i,\ y-j) + \bar{T})$$

$$= \sum_{y=1}^{N} \sum_{x=1}^{N} [S_o(x,y) \cdot T_o(x-i,\ y-j) + \bar{S} \cdot T(x-i,\ y-j) + \bar{T} \cdot S(x,y) + \bar{S} \cdot \bar{T}]$$

$$\text{where } \bar{S} = \frac{1}{N^2} \sum_{y=1}^{N} \sum_{x=1}^{N} S(x,y)$$

$$\bar{T} = \frac{1}{N^2} \sum_{y=1}^{N} \sum_{x=1}^{N} T(x,y)$$

since unbiased images S_o and T_o are defined to have a mean of zero,

$$R(i,j) = \sum_{y=1}^{N} \sum_{x=1}^{N} [S_o(x,y) \cdot T_o(x-i,\ y-j) + \bar{S} \cdot \bar{T}]$$

$$= \sum_{y=1}^{N} \sum_{x=1}^{N} S_o(x,y) \cdot T_o(x-i,\ y-j) + N^2 \bar{S} \cdot \bar{T}$$

$$= R_o(i,y) + N^2 \bar{S} \cdot \bar{T}$$

$$R_o(i,j) = \sum_{y=1}^{N} \sum_{x=1}^{N} S(x,y) \cdot T(x-i, y-j) - N^2 \bar{S} \cdot \bar{T}$$

Although the means $\bar{S}$, $\bar{T}$ can be subtracted from S and T first, the correlation would still be biased since S and T are limited to 4-bit accuracy and the mean is a 20-bit number. Since S and T have 4-bit elements, a multiplication look-up table may be used, considerably faster than conventional multiplication. The product of S and T result in an 8-bit number and the double summation of products produces a 24-bit number, for N=256.

Appendix 13 is a program for correlation and Fig. 5.6 is the flowchart. R_o is computed in the program as follows

$$R(i,j) = |[\sum_{y=1}^{N} \sum_{x=1}^{N} S(x,y) * T(x-i, y-j) - BASE]|/2 ** FACTOR$$

$$\text{where } BASE = [\sum_{y=1}^{N} \sum_{x=1}^{N} S(x,y)] \cdot [\sum_{y=1}^{N} \sum_{x=1}^{N} T(x,y)]/2^{16}, \; N = 256$$

FACTR is a scale factor chosen so that the maximum of R_o does not exceed 256. The program first computes $R_o(0,0)$ to give an estimate of the maximum of R_o, although the peak does not necessarily occur at 0,0. The computation of R_o is lengthy, requiring 18 seconds for each point. To find the actual maximum of R_o, all points would have to be computed. If FACTR is not large enough, R_o will overflow the maximum of 255.

$$\text{If } FACTR = INT\left(\log_2\left(\frac{P}{15}\right)\right)$$

$$\text{where } P = \sum_{y=1}^{N} \sum_{x=1}^{N} (S(x,y) \cdot T(x,y)) - BASE$$

then $R_o(0,0)$ will lie between 15 and 31, leaving considerable headroom for larger values of R_0.

Fig. A5.16 and A5.56 is the cross-correlation between the satellite and radar images used in previous comparisons. i and j range from -32 to +32. This computation required about 21 hours. The correlation produces 8-bit numbers, which are then scaled for the 4-bit display format using the linear contrast enhancement program, with SHIFT=0. GAIN will depend on the actual maximum of R_0 obtained.

As seen in Fig. A5.16, the two images do not appear to be correlated. This is mainly due to the large area of ground clutter, with no corresponding cloud. The correlation technique gives little information about differences and similarities between various regions of the image, and in addition, is a very lengthy procedure.

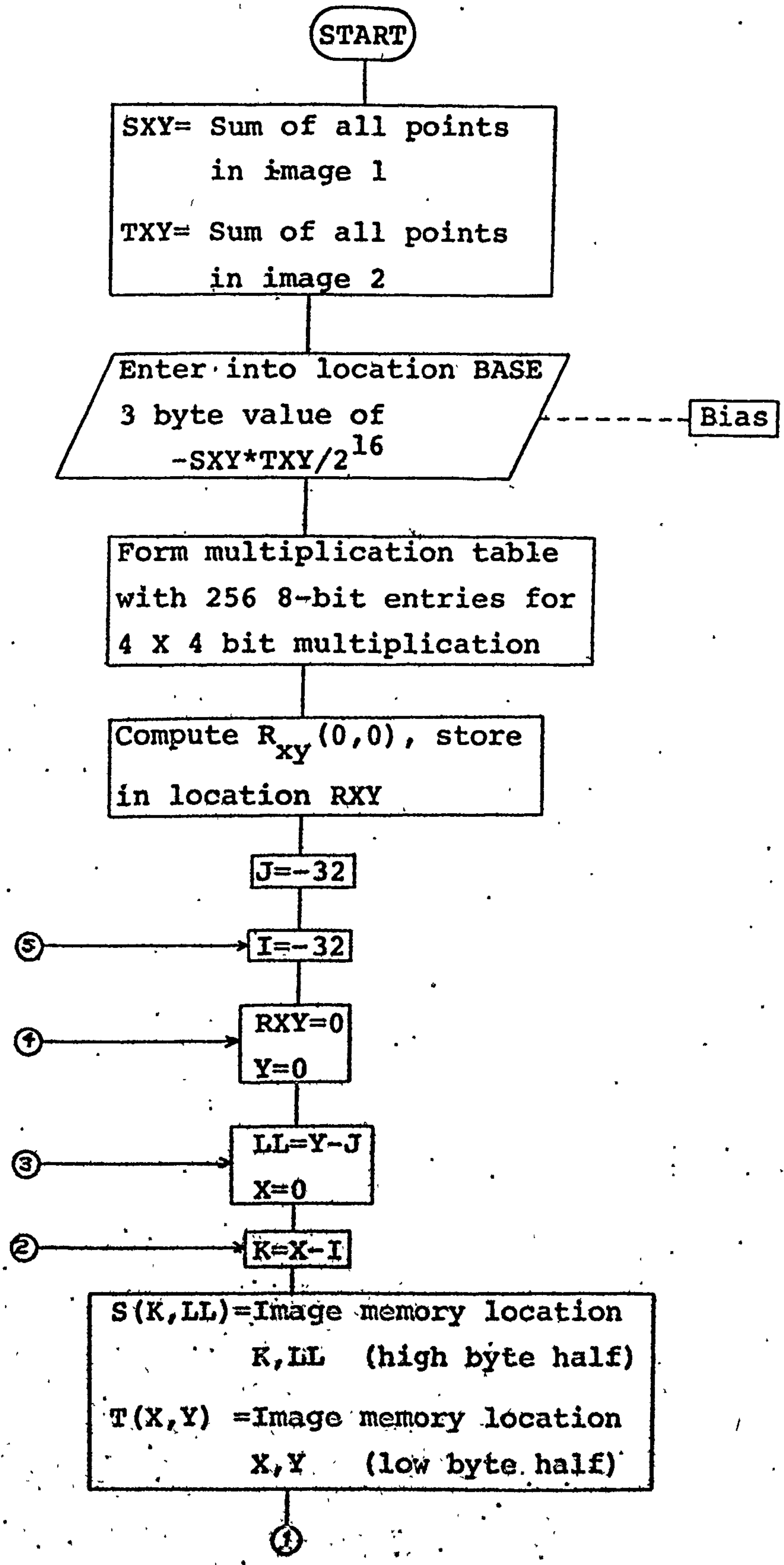


Fig. 5.6 Flowchart for Correlation

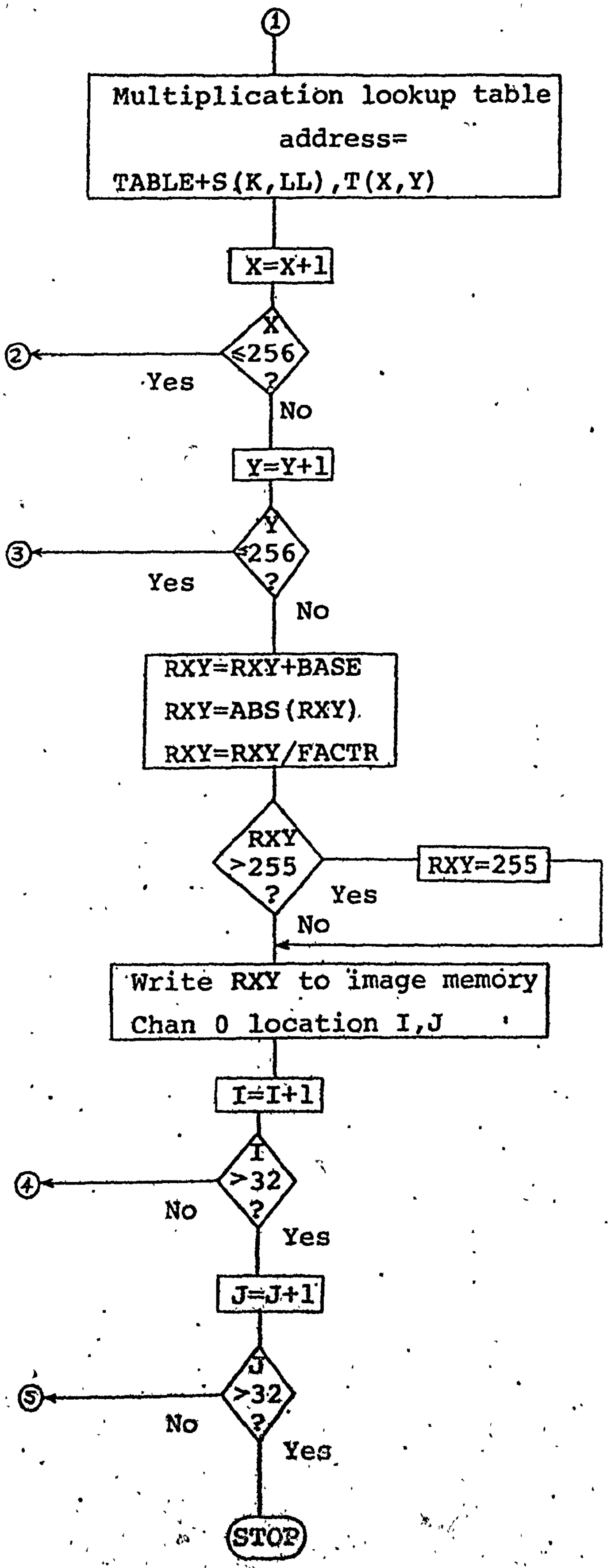


Fig. 5.6 Continued

CHAPTER 6

CONCLUSIONS

6.1 SUMMARY OF RESULTS

The result of work in the hardware area was to extend a general purpose microcomputer system into an image display and processing system. This was done with a large RAM memory, a television monitor and an interface board. This interface of the image display system to the microcomputer was simplified by treating the image memory as a peripheral device, controlled by I/O ports. The major problem in hardware design was the memory interface and display refresh timing. The RAM memory unit employed was essentially a core memory replacement and was not convenient to interface with, as compared to the memory chips themselves. Since the memory was an external unit physically, many cables were required, slowing down the memory transfer time and reducing reliability. Failures in the connecting cables and memory chips were encountered.

The software development required the most time of any task. Although the task was conceptually simple, the implementation was not. The lack of convenient mass storage was the single most complicating factor. Secondly, the raw data was stored in completely different formats. The radar signal was stored inn analogue form, and had to be digitized. The other major complicating factor was the lack of software development tools, in view of the fact that a considerable amount of assembly language code was written.

Both satellite and radar data were processed to be in a mutually compatable form, although the task was long and complex, in terms of computer and operator time and the number of operations required. The individual images were processed to enhance various features. The results are generally evaluated subjectively, and the considerable variability between images requires that different enhancement techniques be used at the discretion of the user. No enhancement technique was found to be consistently better. Colour and black and white displays, with linear or non-linear contrast adjustment each have advantages depending on the nature of the image or areas in the image. Colour displays show the different levels very clearly, although noise is much more apparent than in black and white. Since the images contain only luminance (brightness) information, the display is in a "false" colour, although the human eye does not see black and white either. However, infrared radiation is invisible.

Satellite and radar images were compared for orbits 7989 and 8001. No bird echos were apparent in either of the radar images. No significant cloud echos were received during orbit 8001. Orbit 7989 contained some clouds with corresponding echos, although these clouds were not as clearly visible as others. These clouds were probably low clouds, due to their higher apparent temperature (low contrast from the ground). The colder and higher clouds were probably missed by the radar beam. Ground clutter was the worst problem. MTI processing removed the ground clutter but also most of the clouds.

6.2 CONCLUSIONS

An image display system was built and was used to process, compare, and display satellite and radar imagery. The system successfully demonstrated that it could be used for general purpose image processing not requiring complex operations. The display resolution of 256 elements square with four-bit coding was found to be adequate for most purposes, provided dynamic range compression and contrast enhancement was employed. Large non-display memory workspace was found to be necessary, mainly for the retention of input images since disc storage was not available.

The considerable variability of images requires that processing and enhancement be done on an interactive basis since no enhancement algorithm or display mode offers consistently better results. The programming of the display system is simple and straightforward so that programs for various processing algorithms may be written quickly and easily in assembly language. The programs for formatting the data from its source are more complex, although these are generally fixed, provided that the data is always obtained from the same source and format.

The correlation between radar echos and clouds was of limited success, mainly due to the limited number of data sets available. For each data set, both visual and infrared (daytime) should be available. The radar data should be processed to remove the ground clutter only, with a long range as possible using the polarization mode that best displays cloud echos.

6.3 RECOMMENDATIONS FOR FUTURE WORK

There is a considerable amount of work that could be done in both the display hardware area and the processing of satellite and radar images. Colour displays have considerable potential for the uniquely displaying data from three channels by using red, green, and blue video signals.

The hardware can be easily extended to display two channels, four bits each. The addition of the second channel could control the amplitude (hence brightness) of all three video signals, giving 16 colours, each with 16 brightness values. Combining two (or three) monochrome images produces a colour composite image. The hardware could be flexible (programmable) in the manner the channels are combined, to produce the best composite image as far as the user is concerned. Some of the channels that could be combined in various manners include satellite visual and infrared images, and radar signals of various polarizations.

The speed of processing could be improved by at least an order of magnitude by employing advanced microprocessors such as the 8086. The 8086 offers an expanded address space, so that the image and display memory could be directly addressable, along with program code memory.

A minicomputer such as the HP1000 and an array processor interfaced to an image display would be a very powerful combination. However, the interface between the display and the computer (and software) would be more difficult than a dedicated microprocessor system. It is extremely desirable to have the display memory a part of the computer address space rather than to treat it as a peripheral device. This is clumsy and difficult to implement using a conventional microprocessor. Most commercially available image display systems have good display capability but limited processing capability.

More satellite and radar data needs to be examined. Orbiting satellites are not convenient since only two passes are available each day. Geosynchronous satellites should be seriously considered, since they provide continuous coverage. The radar and satellite images should be compared under different weather conditions. Convective cloud systems probably produce the best correlation.

APPENDIX 1

OPERATING THE SELF-CONTAINED SYSTEM (SCS)

Loading SCS: Load the SCS object paper tape in the normal fashion, using the RO command.

Starting SCS: SCS is initially started at address 0E50. Consult SCS manual for commands. It is recommended that 1500 is used as the first file starting address e.g. FILE/TEST/1500. Make certain that source file and object code to be created (and SCS program) do not conflict. The SCS may be run using the teletype or CRT as a terminal, simply by console device assignment using the MDS monitor.

SCS Differences: The SCS uses the same assembly language statements and syntax as the Intel assembler with the following differences:

(i) the colon after a label is optional but not required.

(ii) the maximum length of a label is 5 characters.

(iii) a comment line must start with an asterisk.

(iv) a comment after a statement does not require a semi-colon.

(v) octal and binary constants are not supported.

(vi) character strings are not supported with the DB pseudo-op. The DW pseudo-op can define a 2-character string, but the characters are entered in reverse order.

(vii) register names PSW and SP are not recognized. They should be defined by statements

```
PSW EQU 6

SP EQU 6
```

(ix) the dollar sign, used for relative symbolic addressing, refers to the current location counter which points to the next instruction after the present instruction (in contrast to the Intel assembler, where the current location point to the first byte of the current instruction).

Exiting SCS: Control may be passed to the Intellec monitor by EXEC 0. The Intellec monitor is recommended for use rather than the SCS monitor functions it has available. To re-enter the SCS, the starting address is 0E67. The cold start (0E50) will destroy the file directory and hence access to the source file.

Source storage: The source files are saved on paper tape. The tape is punched on the teletype. In order to use the teletype, the console must be defined for teletype. If the CRT is normally used, change device status with the AC=T. Source tapes are normally loaded with the paper tape reader. To load a source tape:

(i) enter SCS, define a file.

(ii) return to monitor.

(iii) change location 0EF7 to 06 (defines reader instead of keyboard as input to SCS).

(iv) if the terminal is the teletype, enter 00 into locations 0106, 0107, and 0108. This disables input echo. If the CRT is used, the source will be printed as it is read in.

(v) load source tape in reader

(vi) start SCS at 0E67, tape will read in.

(vii) when the tape stops, press interrupt 0.

(viii) change location 0EF7 back to 03 and 0106-0108 if necessary.

(ix) restart SCS with G.

(x) source file should be present.

The source file should be saved after the program is typed in, in case of system crashes. Corrections can be appended to the tape or by simply typing the corrections on the teletype console with the tape punch on.

Additions to SCS: Due to available memory space in SCS, additional functions have been added to SCS.

(i) symbol table dump. EXEC 0D70 prints the symbol table on the console, useful for debugging purposes.

(ii) source lines can be renumbered, starting at 0000 with an increment of 0010. To renumber, EXEC 0F68.

(iii) the CUST command has been defined for setting up hex-object read and write via the CRT port. Executing CUST initializes the punch and read devices as user-defined device #1 (AR=1, AP=1). Devices can be reassigned with the monitor "A" command. Hex-object can be stored on tape via a modem connected to the CRT port.

(iv) EXEC 0039 generates a symbolic link table. Symbols and their absolute values are printed along with the EQU pseudo-op and line number (starting at 8500). This allows program modules to be linked if the entire program source file is too large. If the symbol table is punched on paper tape, this table can be read into another source file to resolve external references.

(v) a HLT instruction resides in location 38 to serve as an "error trap". If due to a programming error, the program jumps to a location with undefined op codes or to non-existent memory, an RST 7 (CALL 38) will be executed since these locations normally contain FF (RST 7).

Since the source file is always resident in RAM, there is a limit to the size of program SCS can handle. Documentation is a problem, since comments decrease buffer space available for program statements, and increase size of source tape and listing time. Normal horizontal tabs are not available and spaces must be used instead. Since spaces also increase source file length, statements without labels should start at character position 2.

A source listing or current input line can be cancelled with a "control-x". An assembly list can only be halted with a zero level interrupt and a warm start at 0E67.

I/O Drivers: The SCS uses the [illegible] monitor for ASCII terminal I/O. Locations 0105, 0104, 0107 contain a call to the output routine with the character to be printed in register C. The calling address may be changed for special drivers. Locations 0EF6, 0EF7, 0EF8 contain a call to the input routine, with the character read in register A.

Locations 0EF0 and 0F01 contain port addresses for UART data and status read respectively. This is only used in conjunction with stopping a LIST with CONTROL-X. These locations should be changed to F6 and F7 respectively for a CRT instead of a TTY.

File Directory: RAM locations 1024-1030 contain the present file directory in the following manner.

1024-1028 - file name (ASCII), with unused character positions zero-filled.

1029,102A - text buffer start address (LO,HI)

102B,1026 - text buffer and address (LO,HI)

102D,1030 - last line number (4 ASCII digits)

Text Format: The text is stored as ASCII characters in the RAM buffer area specified in the directory. Each line is terminated with a carriage return but no line feed character. All lines include a line number (4 characters). Each line starts with a byte indicating the number of characters in the line, including line number and carriage return. The text buffer is terminated with an end of text (ETX=01) character.

APPENDIX 2

OPERATING BASIC

Loading:

(i) place Basic tape in reader.

(ii) use command RO and the bootstrap loader will be read in.

(iii) use Command G to run Basic loader. Basic will automatically start.

(iv) If the CRT is being used as a console device, after bootstrap loader is read in, change location 3E42 to F7 and 3E5D to F6.

(v) Basic will not operate unless memory locations 0010-1FFF have been set to FF. When the computer is turned on, all RAM locations are set to FF.

Console Devices: If the CRT is being used and hard copy or paper tape is required, console devices must be reassigned. The following program will change between TTY or CRT.

```
8000 INPUT "CRT OR TTY"; A$

8010 IF A$ = "CRT" THEN 8100

8020 IF A$ = "TTY" THEN 8200

8030 GOTO 8000

8100 K1 = 247 :K2 = 246

8110 GOTO 8210

8200 K1 = 245 :K2 = 244

8210 POKE 1217,K1

8220 POKE 1228,K1

8230 POKE 1406,K1

8240 POKE 1498,K1

8250 POKE 1225,K2

8260 POKE 1235,K2

8270 END
```

APPENDIX 3

MICRORAM 3400N INTERFACE UNIT

Table A3.1

Connector	Pin No.	Signal Name
Data in	1	DI-0
	2	DI-1
	:	:
	8	DI-7
	9	DI-8
	10	DI-9
	:	:
	17	DI-16
	18	DI-17
	25	Ground
Data out	1	DO-13
	2	DO-14
	3	DO-15
	4	DO-16
	5	DO-9
	6	Do-10
	7	DO-11
	8	DO-12
	9	DO-8
	10	Do-4
	11	DO-5
	12	DO-6
	13	Do-7
	14	DO-0
	15	Do-1
	16	DO-2
	17	DO-3
	18	DO-17
	25	Ground
Address	1	AI-0
	2	AI-1
	:	:
	15	AI-14
	25	Ground
Memory Control	1	RP input
	2	ECL-1 input
	3	ECL-2 input
	4	DA output
	5	DB output
	6	CR input
	9	Ground

Table A3.2

Signal Name	Description
DI-0 to DI-17	Data input, 18 lines, for memory write
DO-0 to DO-17	Data output, 18 lines, from memory read
AI-0 to AI-14	Address input, 15 lines
RP	Cycle initiate pulse, HI for at least 50 nsec
BCL-1	Byte control 1, for LO-order byte (DI-0 to DI-8 and DO-0 to DO-8) BCL-1 = LO for read BCL-1 = HI for write
BCL-2	Byte control 2, for HI-order byte (DI-9 to DI-17 and DO-9 to DO-17) BCL-2 = LO for read BCL-2 = HI for write
DA	Data available, LO when data out is valid
MB	Memory busy, HI when memory can accept a new RP
Ground	Logic, chassis, and power line ground
GR	General reset, Lo for normal operation

APPENDIX 4

IMAGE DISPLAY INTERFACE BOARD CONNECTOR PIN LIST

Pin	Function	Pin	Function
1	GND	2	
3	MB Chan 1	4	CR Chan 0
5	DA Chan 1	6	RP Chan 0
7	BCL-2 Chan 1	8	BCL-1 Chan 0
9	BCL-1 Chan 1	10	BCL-2 Chan 0
11	RP Chan 1	12	DA Chan 0
13	CR Chan 1	14	MB Chan 0
15		16	
17	GND	18	RED VIDEO
19	GND	20	SYNC
21	GND	22	BLUE VIDEO
23	GND	24	GREEN VIDEO
25	GND	26	BLACK-AND-WHITE VIDEO
27		28	
29	A0	30	
31	A1	32	
33	A2	34	A13
35	A3	36	A14
37	A4	38	A8
39	A4	40	A9
41	A6	42	A10
43	A7	44	A11
45		46	A12
47		48	
49		50	GND
51		52	DI0,DI9 Chan 0 and 1
53		54	DI1,DI10 "
55		56	DI2,DI11 "
57		58	DI3,DI12 "
59		60	DI4,DI13 "
61		62	DI5,DI14 "
63		64	DI6,DI15 "
65		66	DI7,DI16 "
67	DO7 Chan 1	68	DO12 Chan 1
69	DO6 Chan 1	70	DO11 Chan 1
71	DO5 Chan 1	72	DO10 Chan 1
73	DO4 Chan 1	74	DO9 Chan 1
75	DO3 Chan 1	76	DO16 Chan 1
77	DO2 Chan 1	78	DO15 Chan 1
79	DO1 Chan 1	80	DO14 Chan 1
81	DO0 Chan 1	82	DO13 Chan 1
83	GND	84	GND
85	DO0 Chan 0	86	DO13 Chan 0
87	DO1 Chan 0	88	DO14 Chan 0
89	DO2 Chan 0	90	DO15 Chan 0

Pin	Function	Pin	Function
91	DO3 Chan 0	92	DO16 Chan 0
93	DO4 Chan 0	94	DO9 Chan 0
95	DO5 Chan 0	96	DO10 Chan 0
97	DO6 Chan 0	98	DO11 Chan 0
99	DO7 Chan 0	100	DO12 Chan 0

Notes: (i) All pins except 18,20,22,14,26 are memory signals.

(ii) Data inputs for channel 0 and 1 are connected in parallel.

(iii) Data inputs for HI and LO bytes of word are connected in parallel.

APPENDIX 5

PHOTOGRAPHS FROM IMAGE DISPLAY

Figure A5.1 to A5.56 are photographs taken from the display. Most of the black and white photographs have a corresponding colour photograph which shows the difference in detail visible between the two display modes. The nature of the detail of interest will determine which display mode is more suitable. The colour mode allows sharp discrimination between the various levels. The black and white mode allows easier visualization of the relative and absolute levels since the colour mode produces a "false colour" and not what the visible colour is. Infra-red and radar images are only a representation of electromagnetic radiation invisible to the human eye.

The photographs were taken with a model 195 Polaroid Land Camera using a model 1951 close-up lens. The camera was located 25 inches from the monitor screen using a camera mount and light-protective hood. Since the focus range using the close-up lens was too short for the required distance (so that the entire screen filled the camera field of view), the camera bellows were clamped in a closer position from the fully extended position. Frosted plastic was mounted at the film position for framing and focusing adjustments. The black and white photographs were taken at a shutter speed of 1/15 sec at f 11. The colour photographs were taken at a shutter speed of 1/2 sec. at f 5.6. The recommended development time was used. Type 107 film was used for black and white and Polacolour 2 film was used for colour.

Each photograph has a vertical grey or colour scale down the left side and a title across the bottom. Both were generated by software. Due to distortion in the monitor, the edges of the display image are not parallel to the edges of the photographs. In addition, the camera is close to the monitor which results in perspective distortion. The camera should be a long distance away, with a telephoto lens. Another problem, particularly noticeable in colour, is the non-uniform intensity in the photographs. The photographs are darker near the edges. This is probably due to a significant difference in distance from the camera to the centre of the monitor and to the edges of the monitor, or a reduction in intensity off-axis from the monitor.

The photographs were taken from four passes:

7971	May 4, 1978	11:14:19	CST
7989	May 5, 1978	21:42:56	CST
8001	May 6, 1978	20:42:58	CST
8063	May 11, 1978	21:12:36	CST

Orbit 7971 is the only daytime pass and hence both visual and infrared images were available. Images for each pass as shown, both in black and white and colour formats. In addition, large and small scale images are shown, corresponding to 150 and 60 nautical mile radius from the centre to the edges.

The following is a description of each photograph:

Fig. A5.1	As indicated by the title, this is a visual image from orbit 7971 with a range of 60 nautical miles. Some scattered clouds are present. Both Lake Manitoba and Lake Winnipeg are visible, with the latter being mostly cloud covered. The contrast between the water and the land is low, which is generally the case for typical sun angles. Most of sun light is reflected off the lake in one direction.
Fig. A5.2	Histogram flattening has been applied, but with little improvement in detail, "washing out" most of the clouds.

Fig. A5.3	This is the infrared version of Fig. A5.1. The lakes are clearly visible, since the water is much colder than the land. The cloud structure is clearer than in the visible image.
Fig. A5.4	Histogram flattening enhances darker cloud areas, which were not visible without enhancement.
Fig. A5.5	This is a view of a larger area, with a range of 120 nautical miles. Although the scene is mostly dark, Lake of the Woods is visible near the right edge. There are only a few bright clouds.
Fig. A5.6	Histogram flattening provides considerable enhancement of faint clouds, although the contrast between the brightest clouds and the background is lost. Brighter clouds generally have a higher radar reflectivity.
Fig. A5.7	The infrared version of Fig. A5.5 clearly shows the clouds and the lakes. Water is generally characterized by sharp water-land interfaces and uniform brightness or temperature.
Fig. A5.8	Histogram flattening has little effect since Fig. A5.7 is already "flat". Note that all images are linearly enhanced unless otherwise specified.
Fig. A5.9	The Winnipeg area is mostly cloud covered on orbit 7989. This image is slightly overexposed.
Fig. A5.10	A larger view of the area.
Fig. A5.11	Histogram flattening has enhanced both the darker and brighter areas. The cloud brightness (temperature) is fairly uniform and histogram flattening enhances the small brightness variation within the clouds.
Fig. A5.12	The thresholded radar image has been overlaid, showing the areas of echos. The colour version is much better for comparisons.
Fig. A5.13	The PPI photograph corresponding to orbit 7989 was digitized and displayed here, with geometric corrections. The area near the centre is ground clutter.
Fig. A5.14	In this image, the thresholded clouds have been overlaid onto the radar image. The cloud areas are much more visible in colour.
Fig. A5.15	The gradient of Fig. A5.9 shows the cloud edges quite well.
Fig. A5.16	This is the cross-correlation between Fig. A5.9 and Fig. A5.13. The correlation was computed over a total horizontal and vertical shift of 64 elements. The degree

of correlation appears to be the least in the centre. This would normally indicate the images were not geometrically matched. However, the radar ground clutter is the cause of apparent mismatch.

Fig. A5.17	A generally cloudless night shows the Red River and Assiniboine River with extreme clarity. The dark area surrounding the junction of the rivers is the heat generated by Winnipeg.
Fig. A5.18	The larger area reduces the visibility of the rivers, due to the lower resolution.
Fig. A5.19	On this orbit (8001), the lakes are also clearly visible. Notice that the southern tip of Lake Winnipeg is much warmer than the rest of the lake. The signal to noise ratio was poor on this pass. This was caused by a reduction of satellite power level for a telemetry transmission on another channel.
Fig. A5.20	Histogram flattening offers little enhancement in this case.
Fig. A5.21	A larger scale view shows the incoming storm from the West.
Fig. A5.22	Histogram flattening brightens the background detail while destroying some of the cloud structure detail.
Fig. A5.23	The radar image shows the ground clutter blending into cloud echos. The range rings are clearly visible.
Fig. A5.24	The radar echo overlay is not clearly visible in black and white due to the dark background of the satellite image.
Fig. A5.25	This image is the colour version of Fig. A5.1. It is easier to pick out the clouds from the background, which is consistently blue. The colour images show much better amplitude detail since each amplitude level is associated with a distinct colour.
Fig. A5.26	In general, histogram flattening for colour creates a lesser degree of enhancement than it does for black and white images since colour images already give better amplitude discrimination. Noise is more apparent in colour, particularly if large areas of the image are of uniform level.
Fig. A5.27 to A5.35	These are colour versions of Fig. A5.3 to Fig. A5.11.
Fig. A5.36	This shows a larger area of that seen in Fig. A5.35.

Fig. A5.37	This colour radar image shows the echo intensity variation within the echoes much better than the black and white version. However, the background noise is much more apparent in colour.
Fig. A5.38	The echo areas from Fig. A5.37 have been overlaid on Fig. A5.35. The large central area is ground clutter, as well as the region near the top. The cluster of three echos in the top-right region have clouds associated with them, although the clouds are not bright.
Fig. A5.39	Here, the clouds from Fig. A5.35 have been overlaid on Fig. A5.37. Note that most of the clouds do not have echos associated with them.
Fig. A5.40	Both clouds and echos have been thresholded and combined. The red represents areas where clouds have corresponding echos. If is not possible to determine if clouds correspond to echos near the centre since the ground clutter is much brighter than echos contained in it. The only significant areas are the three echos and clouds near the upper right.
Fig. A5.41 to A5.46	These are colour versions of Fig. A5.19 to Fig. A5.24. There are no significant echos during orbit 8001.
Fig. A5.47	B-scan radar display digitized from videotape. Each horizontal range bin (line) contains 128 elements, where each line is an average of 8 range sweeps. There are 256 lines, averaged from 2048 range sweeps. The top left corner corresponds to zero range and azimuth, and the lower right corner corresponds to maximum range and 360 degrees azimuth.
Fig. A5.48	The image, in PPI format, was scan converted from Fig. A5.47. The video has not been D.C. resotred.
Fig. A5.49	Colour version of Fig. A5.17.
Fig. A5.50	Histogram flattening shows the rivers with greater clarity.
Fig. A5.51	Colour version of Fig. A5.18.
Fig. A5.52	Histogram flattening offers little improvement.
Fig. A5.53	Same as Fig. A5-33 except all black elements have been changed to purple.
Fig. A5.54	The gradient shown in colour does not show the cloud edges as clearly as the black and white version Fig. A5.15.

Fig. A5.55 The colour scale shows the large variation in brightness over the monitor screen.

Fig. A5.56 Colour version of cross-correlation.

FIG. A5.1

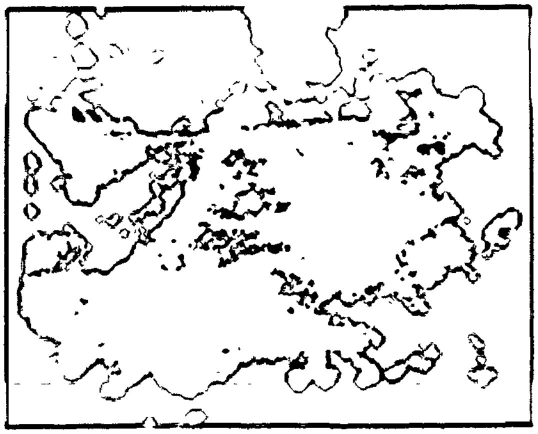


FIG. A5.2

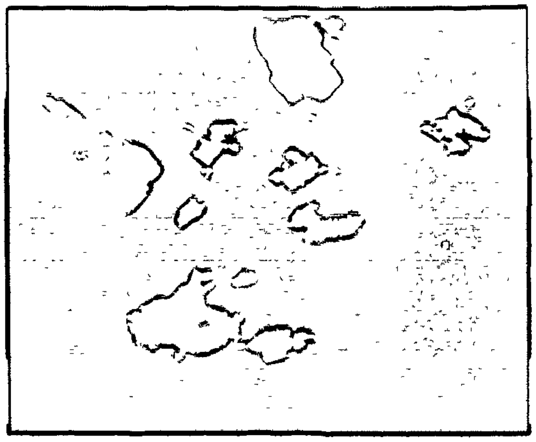


FIG. A5.3

FIG. A5.4

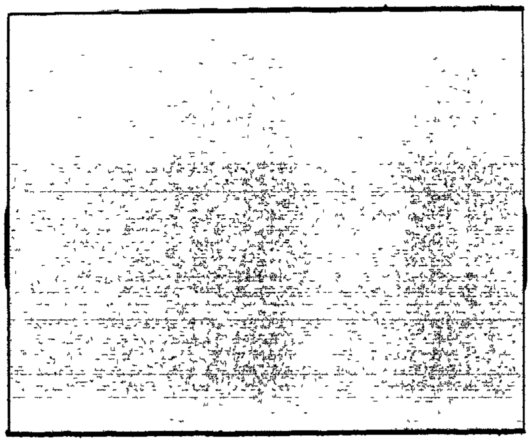


FIG. A5.5

FIG. A5.6

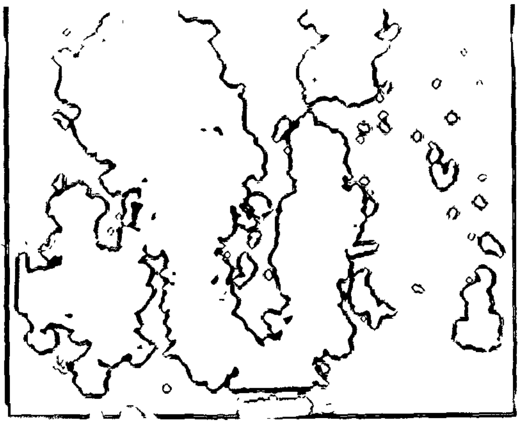


FIG. A5.9

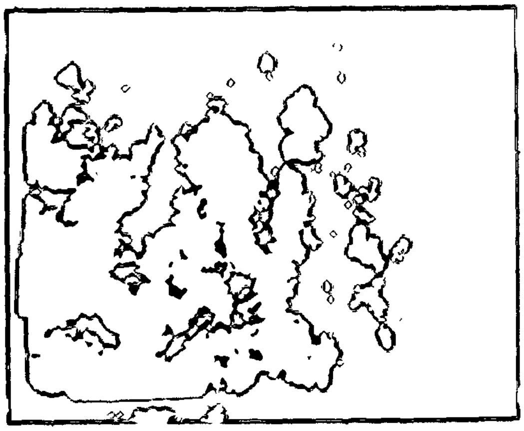


FIG. A5.10

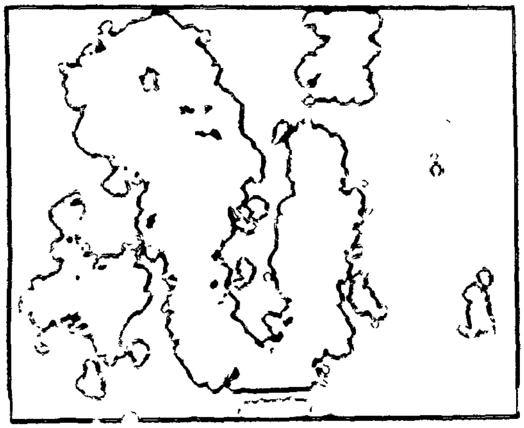


FIG. A5.11

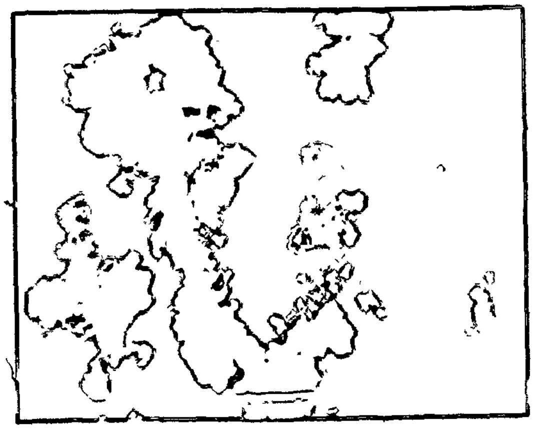


FIG. A5.12

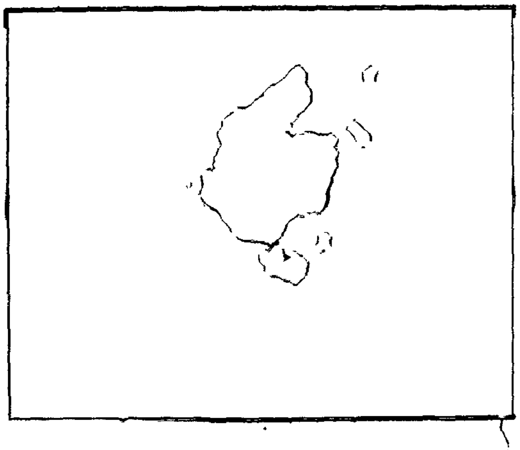


FIG. A5.13

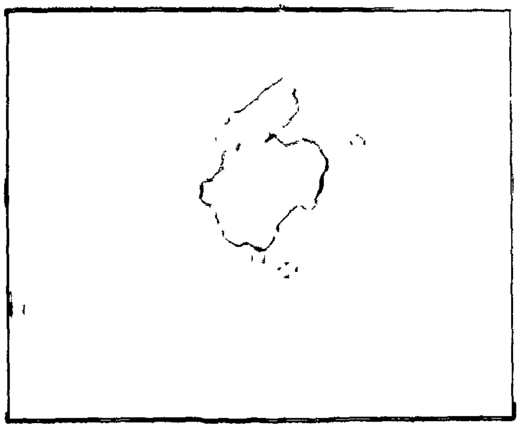


FIG. A5.14

FIG. A5.15

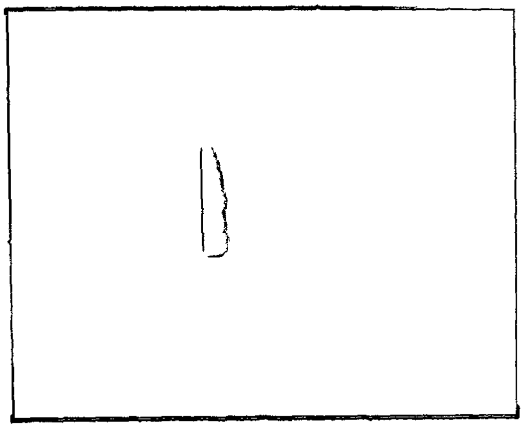


FIG. A5.16

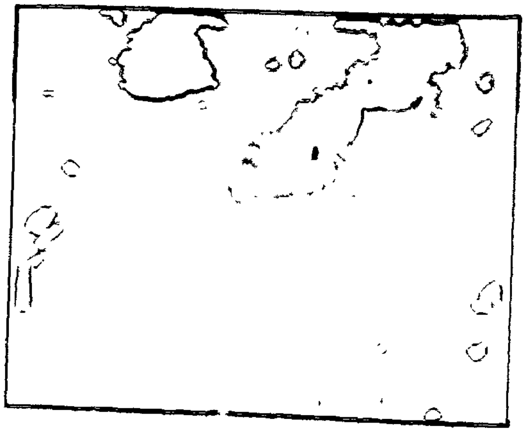


FIG. A5.17

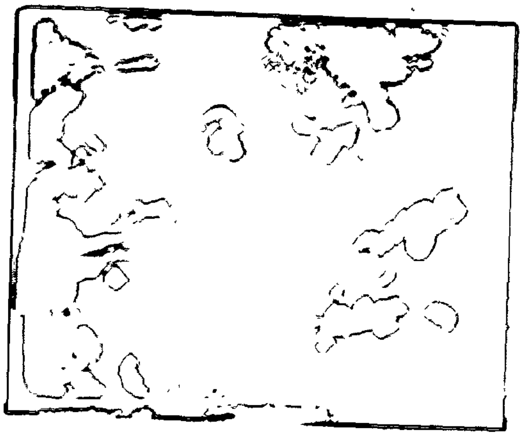


FIG. A5.18

FIG A5.19

FIG. A5.20

FIG. A5.21

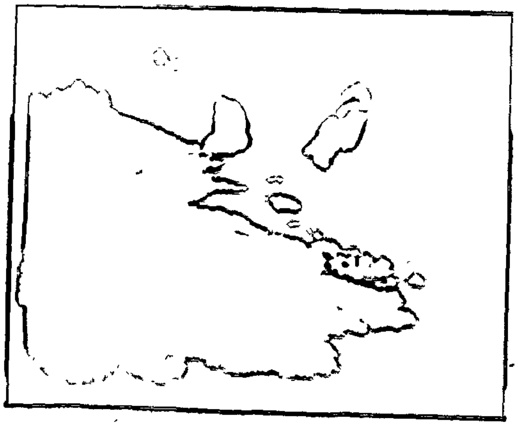


FIG. A5.22

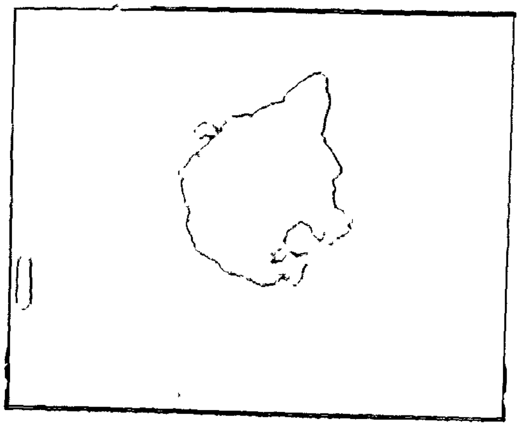


FIG. A5.23

FIG. A5.24

FIG. A5.25

FIG. A5.26

FIG. A.27

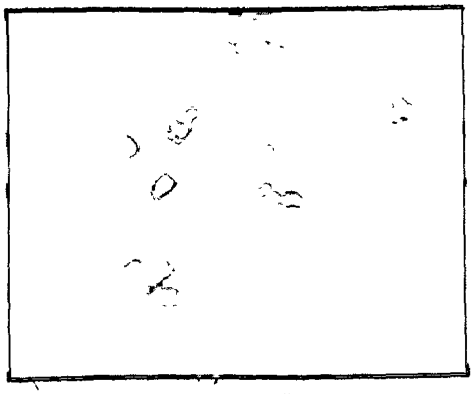


FIG. A5.28

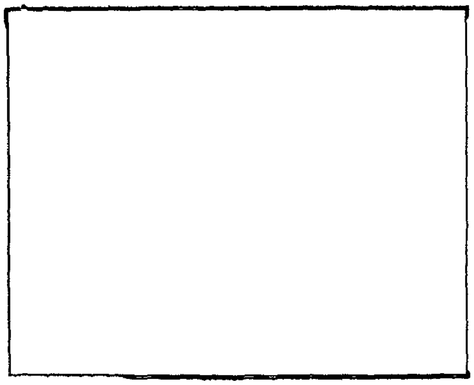


FIG. A5.29

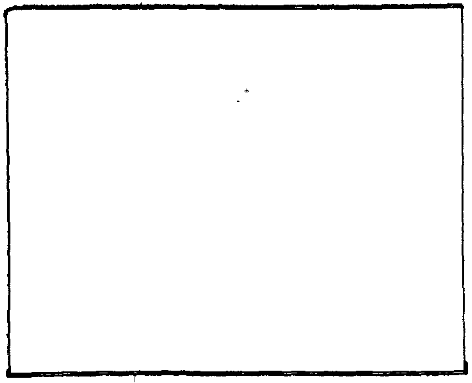


FIG. A5.30

FIG. A5.31

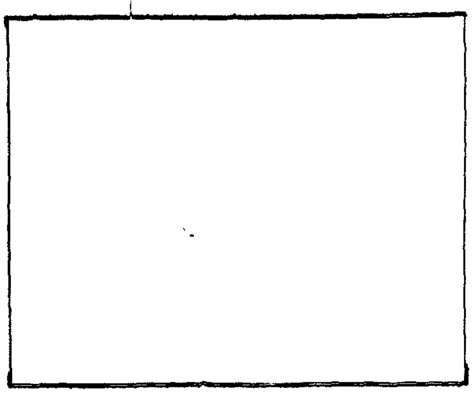


FIG. A5.32

FIG. A5.33

FIG. A5.34

FIG. A5.35

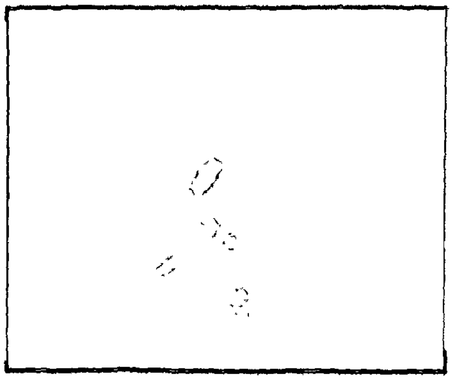


FIG. A5.36

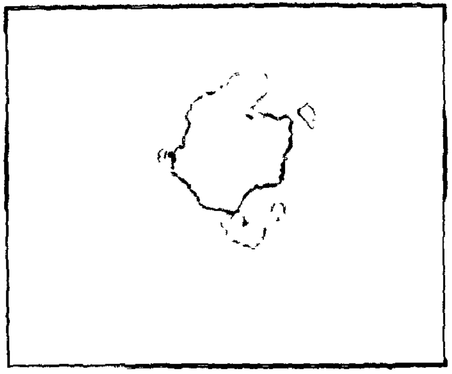


FIG. A5.37

FIG. A5.38

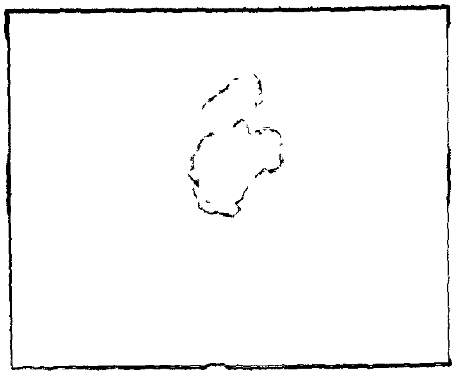


FIG. A5.41

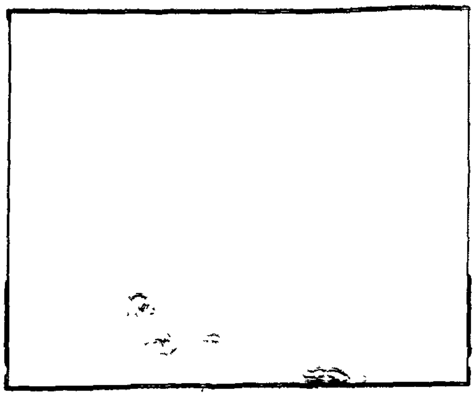


FIG. A5.42

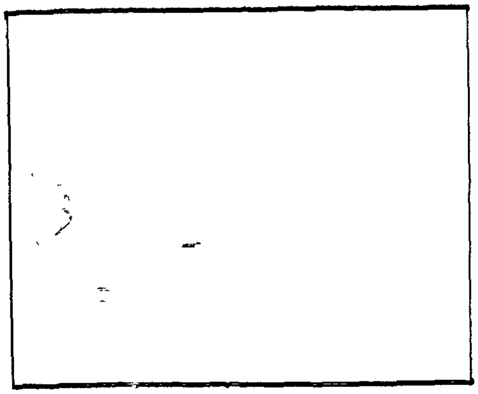


FIG. A5.43

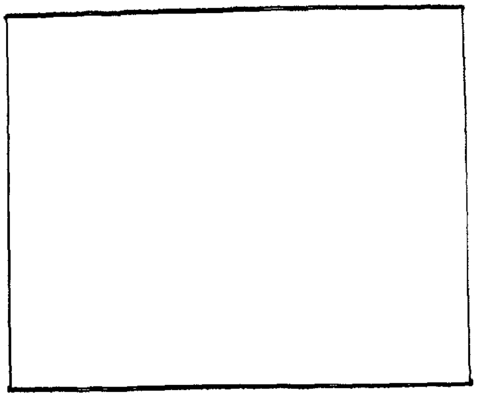


FIG. A5.44

FIG. A5.45

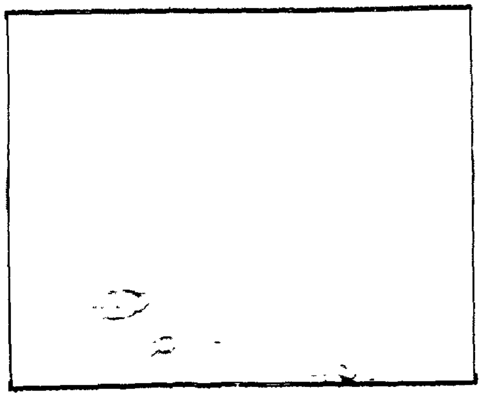
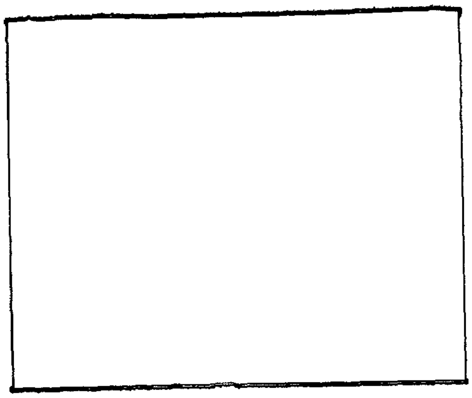
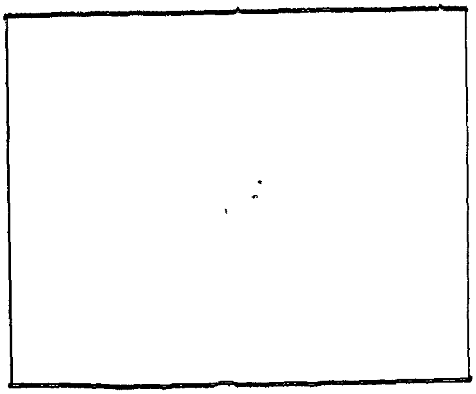


FIG. A5.46





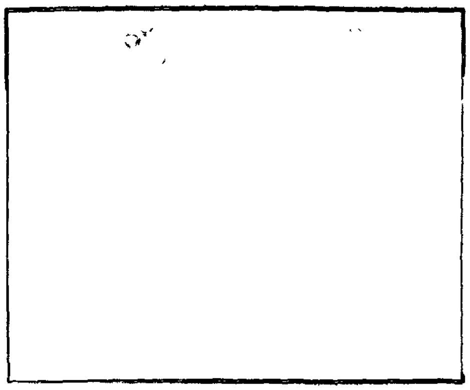


FIG. A5.49

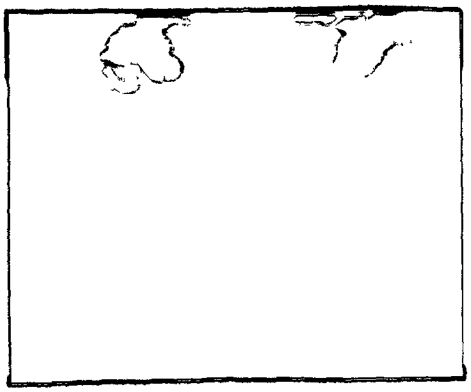


FIG. A5.50

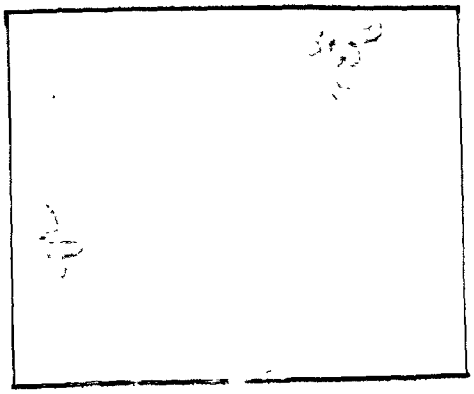


FIG. A5.51

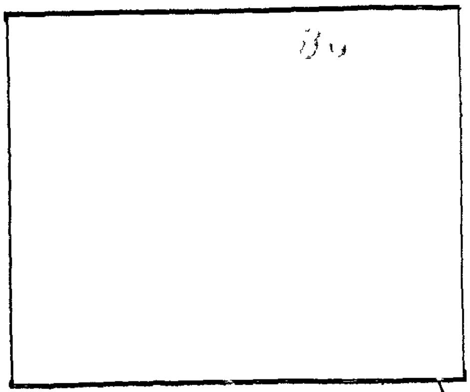


FIG. A5.52

FIG. A5.53

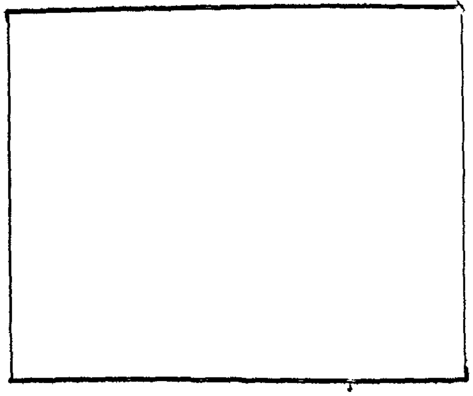


FIG. A5.54

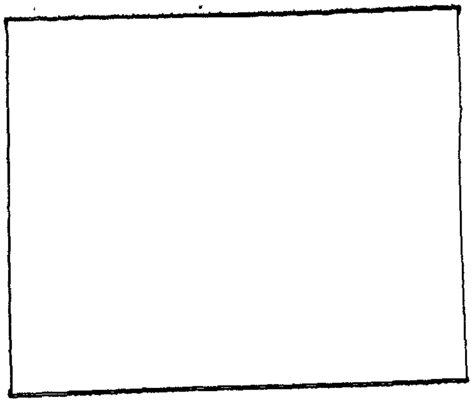


FIG. A5.55

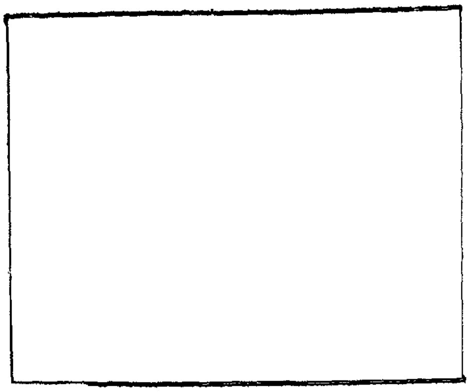


FIG. A5.56

APPENDIX 6

FORTRAN PROGRAM LISTING FOR SATELLITE ORBITAL CALCULATIONS

```
      PROGRAM TST (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1)
C     COMPUTES ORBITAL PREDICTIONS, SATELLITE POSITION,
C     AND IMAGE REGISTRATION
C     COMPUTES CORRECTION FACTORS FOR IMAGE TRANSFORMATION TO AND FROM
C     LAMBERT'S EQUIAZIMUTH PROJECTION
C     COMMON VARIABLES
C     AANOM  =SEMI-MAJOR AXIS OF ANOMALOUS ORBIT
C     ARGPER   =ARGUMENT OF PERIGEE
C     ECC   =ORBITAL ECCENTRICITY
C     EROT    =EARTH ROTATION RATE
C     INC   =ORBITAL INCLINATION
C     OMEGA   =LONGITUDE OF ASCENDING NODE
C     PNODAL    =ORBITAL NODAL PERIOD
C     PROT    =ROTATION RATE OF PERIGEE  (RAD/HR)
C     SROT   =ORBITAL ROTATION RATE (RAD/HR)
C     TAUPER   =TIME FROM PERIGEE TO ASCENDING NODE
C     TOMEGA   =TIME OF ASCENDING NODE
      REAL INC,LAMTRC,LAMMAP,LAMSAT
      REAL LINE,LAT,LONG
      INTEGER D,Y,SHAPE,DOW,H
      INTEGER HP1,ORNUMI,ORNUM
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /2/ LAMSAT,PSISAT,RSAT
      COMMON /3/ PI
      COMMON /4/ RTOD
      DIMENSION SATLAT(17,17),SATLON(17,17),RADLAT(17,17),RADLON(17,17)
     1,DIFLAT(17,17),DIFLON(17,17)
      DIMENSION AZAR(17,17),RAR(17,17),XCORD(17,17),YCORD(17,17)
      DIMENSION CORA(17,17),CORR(17,17)
      DIMENSION DAY(7),MONTH(12)
      DATA DAY/3HMON,3HTUE,3HWED,3HTHU,3HFRI,3HSAT,3HSUN/
      DATA MONTH/3HJAN,3HFEB,3HMAR,3HAPR,3HMAY,3HJUN,3HJUL,3HAUG,3HSEP,
     13HOCT,3HNOV,3HDEC/
C     SCAN RATE IN LINES/HR
      LNRATE=24000
      PI=ABS(4.0*ATAN(1.0))
C     RADIANS TO DEGREES CONVERSION FACTOR
      RTOD=180.0/PI
C     RADIUS TO EDGE OF DESIRED IMAGE (MILES)
      RANGE=150.0
C     NUMBER OF PIXELS BETWEEN SUB-SYNC MARKERS
      WIDTH=1951.0
C     NUMBER OF PIXELS CENTRE IS SHIFTED IN 2000 BYTE RECORD
      SHIFT=30.0
C     RADIUS IN CIRCULAR ARC ON THE EARTH'S SURFACE
      DELTA=RANGE/60.0/RTOD
C     NUMBER OF POINTS IN GEOMETRIC CORRECTION ARRAY
      L=16
      H=16
```

```
C       READ EARTH PARAMETERS
        READ (5,*) BE,OBLECC,EROT
C       READ SATELLITE PARAMETERS
        READ (5,*) PANOM,ECC,INC,ARGPER
C       READ REFERENCE ORBIT DATA   ORNUMI=ORBIT NUMBER
        READ (5,*) OMEGA,TOMEGA,D,M,Y,ORNUMI
C       COMPUTE JULIAN DAY NO. OF REFERENCE ORBIT
        JDNE=JDN(D,M,Y)
        JDNREF=JDNE
        TOSAV=TOMEGA
C       READ TRACKING STATION COORDINATES
        READ (5,*) PSITRC,LAMTRC,SLHTRC
C       READ COORDINATES OF CENTRE OF IMAGE
        READ (5,*) LAMMAP,PSIMAP
C       READ CIRCULAR OR ELIPTICAL MODE
        READ (5,*) SHAPE
C       COMPUTE GEOCENTRIC LATITUDE OF TRACKING STATION
        CALL OBLATE (BE,OBLECC,PSITRC,RTGC,SLHTRC)
C       COMPUTE GEOCENTRIC LATITUDE OF IMAGE CENTRE
        CALL OBLATE (BE,OBLECC,PSIMAP,RTGC1,SLHTRC+.25)
C       COMPUTE SATELLITE ELEMENTS
        CALL ELEMENT (PANOM,SHAPE)
C       READ DAYTIME ASCENDING NODE LIMITS
        READ (5,*) OMEGAD1,OMEGAD2
C       READ NIGHTIME ASCENDING NODE LIMITS
        READ (5,*) OMEGAN1,OMEGAN2
C       COMPUTE LONGITUDE INCREMENT BETWEEN ORBITS
        DOMEGA=PNODAL*SROT
        C=PNODAL*SROT/2.0/PI
C       COMPUTE TIME FOR SCANNING ACROSS RADAR AREA (HR)
        TCOV=2.0*PNODAL*(RANGE/60.0)/360.0
        COSINC=COS(INC)
C       COMPUTE TIME BETWEEN MATRIX POINTS
        DELTAT=TCOV/FLOAT(H)
        HP1=H+1
        LP1=L+1
C       COMPUTE SCANNER ARC BETWEEN SAMPLES ON SCAN
        SEPPIX=60.0/WIDTH/RTOD
C       COMPUTE EARTH ARC BETWEEN GRID POINTS OF IMAGE
        SEPGRD=DELTA/FLOAT(L/2)
C       READ NO. OF SETS OF PREDICTIONS (JOB) AND MODE
C       MODE=0 IF ONLY PREDICTION TIMES ARE REQUIRED
        READ (5,*) JOB,MODE
```

```
      DO 10 JOBI=1,JOB
C     READ STARTING DATE FOR PREDICTIONS
      READ (5,*) D,M,Y
C     COMPUTE JULIAN DAY NUMBER
      JDN1=JDN(D,M,Y)
      K=JDN1
C     PRINT HEADING
      WRITE (6,100) MONTH(M),Y
C     READ ENDING DATE FOR PREDICTIONS
      READ (5,*) D,M,Y
      JDN2=JDN(D,M,Y)
C     READ DAY OR NIGHT PASS
      READ (5,*) KKK
C     KKK=1 FOR DAY , =2 FOR NIGHT
      IF (KKK.EQ.2) GOTO 73
C     SET ASCENDING NODE LIMITS
      OMEGA1=OMEGAD1
      OMEGA2 =OMEGAD2
      GOTO 74
   73 OMEGA1=OMEGAN1
      OMEGA2=OMEGAN2
   74 CONTINUE
C     COMPUTE INTEGER PART OF ORBITAL REVOLUTIONS
C     FROM DAY OF REFERENCE ORBIT TO STARTING DAY
    2 REVS=FLOAT(IFIX(FLOAT(JDN1-JDNE)/PNODAL*24.))
C     SET EPOCH TO STARTING DAY
C     OLD EPOCH IS REFERENCE ORBIT
      CALL PREDCT (REVS,JDNE,SHAPE)
      REVS=0.0
      X=OMEGA
C     KEEP DECREMENTING ORBIT UNTIL ASCENDING NODE
C     LESS THAN OMEGA1 (EAST LIMIT)
    3 IF(X.LT.OMEGA1) GOTO 1
      X=X-DOMEGA
      REVS=REVS-1.0
      GOTO 3
C     SET NEW EPOCH
    1 IF (REVS.LT.0.0) CALL PREDCT (REVS,JDNE,SHAPE)
C     KEEP INCREMENTING ORBIT UNTIL ASCENDING NODE
C     WEST OF EAST LIMIT
    5 CALL PREDCT (1.0,JDNE,SHAPE)
C     NEXT DAY IF ASCENDING NODE DOES NOT FALL
C     BETWEEN LIMITS
      IF (OMEGA.LT.OMEGA1) GOTO 5
      IF (OMEGA.GT.OMEGA2) GOTO 4
C     COMPUTE HR,MIN,SEC
      CALL HMS (TOMEGA,NH,NM,NS)
C     COMPUTE ORBIT NUMBER
      ORNUM=ORNUMI+IFIX(ROUND((FLOAT(JDNE-JDNREF)*24.0+TOMEGA-TOSAV)/
     1PNODAL))
C     COMPUTE TIME SCAN CROSSES CENTRE OF IMAGE
```

```
C     =TML OF RADAR
C     SET GUESS TO INITIAL VALUE DEPENDING ON
C     DAY OR NIGHT PASS
      IF (KKK.EQ.1) GOTO 75
      GUESS=0.2333
      GOTO 76
   75 GUESS=0.7333
   76 TML=TOMEGA+TAUML(GUESS,.000001,.00005,LAMMAP,PSIMAP,SHAPE)
C     COMPUTE SATELLITE SKY POSITION
      CALL STRACK (AZ,EL,LAMSAT,LAMMAP,PSISAT,PSIMAP,RSAT,RTGC1,SEP)
      TAU0=TML-TOMEGA
C     COMPUTE NEW POSITION AFTER 1 SECOND
      CALL GTRACK(SHAPE,.000001,TAU0+1.0/3600.0,RSAT,PSI1,Y1)
      CALL STRACK(AZ1,EL1,Y1,LAMSAT,PSI1,PSISAT,RSAT,RTGC1,SEP1)
C     COMPUTE TIME FOR SCANNING ACROSS RADAR AREA (HR)
      TCOV=2.0*RANGE/60.0/RTOD/ABS(SEP1)/3600.0
C     COMPUTE TIME BETWEEN VERTICAL GRID POINTS
      DELTAT=TCOV/FLOAT(H)
      SEP=ABS(SEP)
C     COMPUTE SENSOR ANGLE FROM NADIR TO IMAGE CENTRE
      SENSA=SENANG(RTGC1,SEP,RSAT,KKK,LAMSAT,LAMMAP)
C     COMPUTE SENSOR ANGLE TO RIGHT EDGE OF IMAGE
      SENSB=SENANG(RTGC1,SEP+DELTA,RSAT,KKK,LAMSAT,LAMMAP)
C     COMPUTE SENSOR ANGLE TO LEFT EDGE OF IMAGE
      SENSC=SENANG(RTGC1,SEP-DELTA,RSAT,KKK,LAMSAT,LAMMAP)
C     COMPUTE HR,MIN,SEC AT TML
      CALL HMS(TML,MH,MM,MS)
C     COMPUTE DATE
      CALL DATE (JDNE,D,M,Y,DOW)
C     IF FIRST DAY OF MONTH, PRINT HEADING
      IF (K.NE.JDNE.AND.D.EQ.1) WRITE (6,100) MONTH (M),Y
      K=JDNE
      TML=TML - TOMEGA
C     COMPUTE LINE NUMBER AT TML SINCE EQUATOR
      LINE=TML*FLOAT(LNRATE)
C     COMPUTE ARC ON EARTH SURFACE CORRESPONDING
C     TO DISTANCE FROM NADIR TO SUB-SYNC MARKER
      SEPMAX=ABS(ASIN(RSAT*0.5/RTGC1)-30.0/RTOD)
C     COMPUTE COLUMN NO. TO CENTRE OF IMAGE
      COLUMN=WIDTH/2.0*(1.0+SEP*SGN(SENSA)/SEPMAX)+SHIFT
      WRITE (6,110) D,DAY(DOW),NH,NM,NS,OMEGA*RTOD,MH,MM,MS,
     1SENSB*RTOD,SENSA*RTOD,SENSC*RTOD,ORNUM
     1,LINE,COLUMN
      IF (MODE.EQ.0) GOTO 26
      SEP=SEP*SGN(SENSA)
C     COMPUTE GRID POINTS
      DO 20 LOOP=1,HP1
      TMLI=TML+FLOAT(LOOP-(1+H/2))*DELTAT
C     COMPUTE NEW SATELLITE POSITION FOR EACH VERTICAL POINT
      CALL GTRACK(0,.000001,TMLI,RSAT,PSISAT,LAMSAT)
      COSPSI=COS(PSISAT)
```

```
C     COMPUTE AZIMUTH OF SCAN LINE
      AZSCAN=ATAN(SQRT(COSPSI**2-COSINC**2)/(C*COSPSI-COSINC/COSPSI)
     1/ABS(COSPSI))
      AZSCAN=ABS(AZSCAN)
      AZSTR=AZSCAN
      DO 20 LSCAN=1,LP1
C     FOR EACH SATELLITE POSITION, COMPUTE
C     HORIZONTAL GRID POINTS
      SENSP=FLOAT(LSCAN-(L/2+1))
C     COMPUTE ARC FOR GRID POSITION
      SEPSCN=SEPGRD*SENSP +SEP
C     COMPUTE SCAN AZIMUTH, RESOLVING AMBIGUITIES
      AZSCAN=AZSTR
      IF (SEPSCN.GT.0.0) AZSCAN=PI-AZSTR
      IF ((KKK.EQ.1.AND.SEPSCN.LT.0.0).OR.(KKK.EQ.2.AND.SEPSCN.GT.0.0))
     1AZSCAN=-AZSCAN
C     COMPUTE GROUND COORDINATES FOR GRID POINT
C     ON SATELLITE IMAGE
      CALL COORD(ABS(SEPSCN),AZSCAN,PSISAT,LAMSAT,LAT,LONG,DLONG)
C     COMPUTE ACTUAL LONGITIUDE
      DLONG=ABS(DLONG)
      LONG=LAMSAT+SGN(SEPSCN)*FLOAT(2*KKK-3)*DLONG
C     FILL ARRAY WITH SATELLITE IMAGE COORDINATES
      IF (KKK.EQ.2) GO TO 25
      SATLAT(LOOP,LSCAN)=LAT
      SATLON(LOOP,LSCAN)=LONG
      GO TO 20
C     IF NIGHT PASS, TRANSPOSE ARRAY
   25 SATLAT((H+2-LOOP),(L+2-LSCAN))=LAT
      SATLON((H+2-LOOP),(L+2-LSCAN))=LONG
   20 CONTINUE
C     PRINT ARRAY OF GROUND COORDINATES
      CALL ARQUT (SATLAT,SATLON,HP1)
      SINMAP=SIN(PSIMAP)
      COSMAP=COS(PSIMAP)
      DO 40 LOOP=1,HP1
      DO 40 LSCAN=1,LP1
      A=FLOAT(H/2+1-LOOP)
      B=FLOAT(LSCAN-(L/2+1))
C     COMPUTE RANGE FROM CENTRE OF EACH GRID POINT
      RADSEP=(DELTA/FLOAT(H/2))*SQRT(A**2 + B**2)
      IF (RADSEP.EQ.0.0) A=1.0E-06
C     COMPUTE AZIMUTH OF EACH GRID POINT
      RADAZ=ATAN2(B,A)
C     COMPUTE CORRESPONDING GROUND COORDINATE
C     OF EACH GRID POINT
      CALL COORD(RADSEP,RADAZ,PSIMAP,LAMMAP,LAT,LONG,DLONG)
C     FILL ARRAY OF GROUND CORDINATES
      RADLAT(LOOP,LSCAN)=LAT
      RADLON(LOOP,LSCAN)=LONG
C     COMPUTE DIFFERENCE BETWEEN GROUND
```

```
C     AND SATELLITE IMAGE COORDINATES
      DIFLAT(LOOP,LSCAN)=LAT-SATLAT(LOOP,LSCAN)
      DIFLON(LOOP,LSCAN)=LONG-SATLON(LOOP,LSCAN)
C     COMPUTE AZIMUTH AND RANGE FROM CENTRE
C     TO SATELLITE IMAGE COORDINATES
      CALL STRACK(AZ,EL,SATLON(LOOP,LSCAN),LAMMAP,SATLAT(LOOP,LSCAN),
     1PSIMAP,RSAT,RTGC1,RSEP)
      AZAR(LOOP,LSCAN)=AZ
      RAR(LOOP,LSCAN)=RSEP*60.0
C     COMPUTE ROTATION OF RADAR (GROUND) IMAGE
C     RELATIVE TO SATELLITE IMAGE FOR EACH GRID POINT
      XCORD(LOOP,LSCAN)=RADAZ
      IF (XCORD(LOOP,LSCAN).LT.0.0) XCORD(LOOP,LSCAN)=2.0*PI+XCORD(LO
     1OP,LSCAN)
      XCORD(LOOP,LSCAN)=XCORD(LOOP,LSCAN)-AZAR(LOOP,LSCAN)
      IF (ABS(XCORD(LOOP,LSCAN)).GE.PI) XCORD(LOOP,LSCAN)=XCORD(LOOP,
     1LSCAN)-2.0*PI*SGN(XCORD(LOOP,LSCAN))
C     COMPUTE AZIMUTH CORRECTION FOR RADAR GRID POINTS
      CORA(LOOP,LSCAN)=XCORD(LOOP,LSCAN)
      IF (LOOP.EQ.9.AND.LSCAN.EQ.9) CORA(LOOP,LSCAN)=0.0
      CORR(LOOP,LSCAN)=1.0/RTOD
C     COMPUTE RANGE CORRECTION FACTOR
      IF (RADSEP.EQ.0.0) GOTO 40
      CORR(LOOP,LSCAN)=RSEP/RADSEP/RTOD
   40 CONTINUE
C     PRINT ARRAYS
      CALL AROUT(RADLAT,RADLON,HP1)
      CALL AROUT(DIFLAT,DIFLON,HP1)
      CALL AROUT(AZAR,RAR,HP1)
      CALL AROUT(CORA,CORR,HP1)
   26 CONTINUE
      REVS=1.
      GOTO 5
    4 JDN1=JDN1+1
      IF (JDN1.LE.JDN2) GOTO 2
   10 CONTINUE
C     PRINT NO. OF SCAN LINES FOR FULL COVERAGE
      SIZEV=TCOV*FLOAT(LNRATE)
C     PRINT NO. OF HORIZONTAL ELEMENTS ACROSS
      SIZEH=WIDTH*DELTA/SEPMAX
      PRINT*,SIZEV
      PRINT*,SIZEH
  100 FORMAT('1',14X,A3,I5,/,'0',' DATE  ASCENDING NODE   MAP LINE
     1   MAX ANG   CENTRE    MIN ANG       ORBIT LINE NO   COLUMN',/,
     1' ',9X,'TIME         LONG        TIME',/)
  110 FORMAT (' ',I2,A4,I3,I3,I3,F8.2,I4,I3,I3,5X,3F10.4,5X,I6
     1,2F9.2)
```

```
C     LEAVE BLANK LINE BETWEEN PAIRS OF ARRAY POINTS
      COMMON /4/ RTOD
      INTEGER HP1
      REAL LAT(HP1,HP1),LONG(HP1,HP1)
      WRITE (6,140)
      DO 50 II=1,HP1
      WRITE (6,150) (RTOD*LAT(II,J),J=1,HP1)
      WRITE (6,160) (RTOD*LONG(II,J),J=1,HP1)
   50 CONTINUE
  140 FORMAT (1H1)
  150 FORMAT (1H0,17F7.2)
  160 FORMAT (1H ,17F7.2)
      RETURN
      END
```

```
      SUBROUTINE OBLATE (BE,OBLECC,PSITRC,RTGC,SLHTRC)
C     COMPUTE TRACKING STATION COORDINATES ON A GEOCENTRIC SYSTEM
C     APPOXIMATES AN OBLATE EARTH
C     BE   =POLAR RADIUS OF EARTH
C     OBLECC   =ECCENTRICITY OF EARTH
C     PSITRC   =GEODETIC LATITUDE OF TRACKING STATION
C     PSITRC   =GEOCENTRIC LATITUDE OF TRACKING STATION
C     RTGC   =GEOCENTRIC RADIUS OF TRACKING STATION
C     SLHTRC   =HEIGHT ABOVE SEA LEVEL OF TRACKING STATION
      C=(OBLECC*COS(PSITRC))**2
      RTGD=BE*(1. +C/2. +3./8.*C**2 +5./16.*C**3)
      X=OBLECC**2*SIN(2*PSITRC)/2./(1.-(OBLECC*SIN(PSITRC))**2)
      PSITRC=PSITRC - X*RTGD/(RTGD+SLHTRC)
      RTGC=RTGD+SLHTRC*(1.-X**2*RTGD/2./(RTGD+SLHTRC))
      RETURN
      END
```

```
      SUBROUTINE ELEMENT (PANOM,SHAPE)
C     COMPUTE ORBITAL ELEMENTS
C     PANOM   =ANOMALISTIC PERIOD
C     SHAPE=/ FOR CIRCULAR APPROXIMATION,  =1 FOR ELLIPTICAL ORBIT
C     E   =ECCENTRIC ANOMALLY
      REAL INC
      INTEGER SHAPE
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /3/ PI
      C1= 5.643912758E-6
      C2= 2.839674972E10
      AANOM=((PANOM/C1)**2)**(1./3.)
      C=C2/(1.-ECC**2)**2/SQRT(AANOM**7)
      COSI=COS(INC)
      SROT=C*COSI+EROT
      PROT=C/2.*(5.*COSI**2-1.)
      PNODAL=PANOM*(1.-PROT/PI)
      IF(SHAPE.EQ.0) RETURN
      E=ACOS((ECC+COS(ARGPER))/(1.0+ECC*COS(ARGPER)))
      IF (E.GT.PI) E=2.*PI-E
```

```
      STOP
      END
      FUNCTION ROUND(ARG)
C     ROUND REAL NUMBER TO NEAREST INTEGER
      SIGN=SGN(ARG)
      ARG=ABS(ARG)
      FRAC=ARG-FLOAT(IFIX(ARG))
      ROUND=FLOAT(IFIX(ARG))
      IF (FRAC.GE.0.5) ROUND=ROUND+1.0
      ROUND=SIGN*ROUND
      RETURN
      END
      FUNCTION SGN(ARG)
C     SIGN FUNCTION
      IF (ARG.GE.0.0)  GO TO 56
      SGN=-1.0
      GO TO 57
   56 SGN=1.0
   57 RETURN
      END
      FUNCTION SENANG(RTGC1,SEP,RSAT,KKK,LAMSAT,LAMMAP)
C     COMPUTE SENSOR ANGLE FROM NADIR GIVER GROUND COOORDINATES
C     AND SUB SATELLITE POINT
      REAL LAMSAT,LAMMAP
      SENANG=ATAN(RTGC1*SIN(SEP)/(RSAT-RTGC1*COS(SEP)))
      SENANG=SENANG*FLOAT(2*KKK-3)*SGN(LAMMAP-LAMSAT)
      RETURN
      END
      SUBROUTINE COORD(GCA,AZ,PSI,LAM,LAT,LONG,DLONG)
C     COMPUTE LATITUBE (LAT) AND LONGITUDE (LONG) GIVEN
C     GCA (GREAT CIRCLE ARC) AND AZ (AZIMUTH)
      COMMON /3/ PI
      REAL LAM,LAT,LONG
      IF (AZ.EQ.0.0.OR.AZ.EQ.PI) AZ=AZ+1.0E-06
      TAZD2=TAN(AZ/2.0)
      GCA=ABS(GCA)
      ARGM=(PI/2.0-PSI-GCA)/2.0
      ARGP=(PI/2.0-PSI+GCA)/2.0
      IF (ARGP.EQ.0.0.OR.ABS(ARGP).EQ.PI.OR.ABS(ARGP).EQ.PI/2.0.OR.
     1ABS(ARGP).EQ.3.0*PI/2.0) ARGP=ARGP+1.0E-6
      ANG1=ATAN(COS(ARGM)/COS(ARGP)/TAZD2)
      ANG2=ATAN(SIN(ARGM)/SIN(ARGP)/TAZD2)
      IF (ANG2.EQ.0.0.OR.ABS(ANG2).EQ.PI) ANG2=ANG2+1.0E-6
      DLONG=ANG1-ANG2
      LAT=PI/2.0-2.0*ATAN(TAN(ARGM)*SIN(ANG1)/SIN(ANG2))
      LONG=LAM-DLONG
      RETURN
      END
      SUBROUTINE AROUT(LAT,LONG,HP1)
C     PRINT TWO ARRAYS, LAT,LONG EACH HP1 X HP1
```

```
      TAUPER=(E-ECC*SIN(E))*PNODAL/PI/2.
      RETURN
      END
      FUNCTION JDN(D,M,Y)
C     COMPUTE JULIAN DAY NUMBER
      INTEGER D,Y
      IF (M.GT.2) GOTO 1
      M=M+13
      Y=Y-1
    1 JDN=IFIX(365.25*FLOAT(Y))+IFIX(30.6001*FLOAT(M+1))+D+1720982
      RETURN
      END
      SUBROUTINE DATE (JDN,D,M,Y,DOW)
C     COMPUTE DAY,MONTH,YEAR AND DAY OF WEEK GIVEN JULIAN DAY NUMBER
      INTEGER D,Y,DOW
      DN=FLOAT(JDN-1720982)
      Y=IFIX((DN-122.1)/365.25)
      I=IFIX(365.25*FLOAT(Y))
      M=IFIX((DN-FLOAT(I))/30.6001)
      D=IFIX(DN)-I-IFIX(30.6001*FLOAT(M))
      M=M-1
      IF(M.GE.14) M=M-12
      IF(M.LT.3) Y=Y+1
      C=(DN+5.)/7.
      DOW=IFIX(7.*(C-FLOAT(IFIX(C))+.01))
      RETURN
      END
      SUBROUTINE HMS(HOURS,H,M,S)
C     COMPUTE HOURS,MIN,SEC FROM TIME IN DECIMAL HOURS
      INTEGER H,S
      H=IFIX(HOURS)
      C=(HOURS-FLOAT(H))*60.
      M=IFIX(C)
      S=IFIX((C-FLOAT(M))*60.)
      REM=(C-FLOAT(M))*60.0 - FLOAT(S)
      IF (REM.GE.0.5) S=S+1
      RETURN
      END
C     COMPUTE NEW EPOCH GIVEN OLD EPOCH AND NUMBER OF ORBITAL REVOLUTIONS
C     FROM OLD EPOCH
C     REVS   =P- NUMBER OF REVOLUTIONS
C     FLAG   =SHAPE
      SUBROUTINE PREDCT(REVS,JDN,FLAG)
      REAL INC
      INTEGER FLAG
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /3/ PI
      OMEGA=OMEGA+SROT*REVS*PNODAL
      OMEGA=ATAN2(SIN(OMEGA),COS(OMEGA))
      IF(FLAG.EQ.0) GOTO 1
```

```
      ARGPER=ARGPER+PROT*REVS*PNODAL
      ARGPER=ATAN2(SIN(ARGPER),COS(ARGPER))
      IF(ARGPER.LT.0.) ARGPER=ARGPER+2.*PI
      C=COS(ARGPER)
      E=ACOS((ECC+C)/(1.+ECC*C))
      IF(ARGPER.GT.PI) E=2.0*PI -E
      TAUPER=(E-ECC*SIN(E))*PNODAL/2./PI
    1 TOMEGA=TOMEGA+PNODAL*REVS
      C=TOMEGA/24.
      JDN=IFIX(C+FLOAT(JDN))
      TOMEGA=24.*(C-FLOAT(IFIX(C)))
      IF(TOMEGA.LT.0.)TOMEGA=TOMEGA+24.
      RETURN
      END
```

```
      FUNCTION TAUML (TAU,EERROR,PERROR,LAMMAP,PSIMAP,FLAG)
C     COMPUTE TIME AFTER ASCENDING NODE WHEN LINE PERPENDICULAR TO
C     SUB-SATELLITE POINT VELOCITY VECTOR WILL INTERSECT WITH A GIVEN MA
C     COORDINATE, USING NEWTON'S METHOD FOR ITERATION
C     TAU   =TIME AFTER ASCENDING NODE
C     EERROR   =ALLOWABLE ERROR IN THE ECCENTRIC ANOMALLY
C     PERROR   =ALLOWABLE ERROR IN PSIMAP
C     LAMMAP   =LONGITUDE OF MAP POINT
C     PSIMAP   =LATITUDE OF MAP POINT
      REAL INC,LAMMAP
      INTEGER FLAG
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /2/ LAMSAT,PSISAT,RSAT
      COMMON /3/ PI
      DELTAT=.001
      SN1=PNODAL/4.
      SN2=3.*SN1
      C=PNODAL*SROT/2./PI
      COSI=COS(INC)
      T1=TAU
      PSI1=PSITLS(TAU,LAMMAP,COSI,FLAG,EERROR,SN1,SN2,C)
    1 TAUML=T1+DELTAT
      PSI2=PSITLS(TAUML,LAMMAP,COSI,FLAG,EERROR,SN1,SN2,C)
      DPSI=PSIMAP-PSI2
      IF (ABS(DPSI).LE.PERROR) RETURN
      DELTAT=DPSI*(T1-TAUML)/(PSI1-PSI2)
      T1=TAUML
      PSI1=PSI2
      GOTO 1
      END
```

```
      FUNCTION PSITLS(TAU,LAMMAP,COSI,FLAG,EERROR,SN1,SN2,C)
C     COMPUTE LATITUDE OF SCAN LINE CROSSING GIVEN TAU
C     PSISAT   =LATITUDE OF SUB POINT
C     LAMSAT   =LONGITUDE OF SUBPOINT
C     DLAM   =DIFFERENCE IN LOGITUDE BETWEEN MAP AND SUB POINT
      REAL LAMMAP,LAMSAT,INC
```

```
      INTEGER FLAG
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /2/ LAMSAT,PSISAT,RSAT
      COMMON /3/ PI
      COMMON /4/ RTOD
      LOGICAL POS
      CALL GTRACK(FLAG,EERROR,TAU,RSAT,PSISAT,LAMSAT)
      DLAM=LAMSAT-LAMMAP
      COSPSI=COS(PSISAT)
      TANAZ=SQRT(COSPSI**2-COSI**2)/(C*COSPSI-COSI/COSPSI)
      TANAZ=TANAZ/ABS(COSPSI)
      AZ=ABS(ATAN(TANAZ))
      POS=TAU.GT.SN1.AND.TAU.LT.SN2
      IF ((POS.AND.DLAM.GE.0.0).OR.(.NOT.POS.AND.DLAM.LE.0.0)) AZ=PI-AZ
      IF (DLAM.LT.0.0) AZ=-AZ
      IF (ABS(PSISAT).EQ.PI/2.0) PSISAT=PSISAT+1.0E-6
      TANCL=TAN((PI/2.0-PSISAT)/2.0)
      AMBD2=(DLAM-AZ)/2.0
      APBD2=(DLAM+AZ)/2.0
      IF (APBD2.EQ.0.0.OR.ABS(APBD2).EQ.PI.OR.ABS(APBD2).EQ.PI/2.0)
     1APBD2=APBD2+1.0E-6
      PSITLS=ATAN(TANCL*COS(AMBD2)/COS(APBD2))-
     1ATAN(TANCL*SIN(AMBD2)/SIN(APBD2))
      PSITLS=PI/2.0-PSITLS
      RETURN
      END
```

```
      SUBROUTINE GTRACK (FLAG,EERROR,TAU,RSAT,PSISAT,LAMSAT)
C     COMPUTE SATELLITE SUBPOINT GIVEN TAU
C     RSAT   =RADIAL DISTANCE OF SATELLITE FROM CENTRE OF EARTH
      REAL INC,LAMSAT,NU
      INTEGER FLAG
      COMMON /1/ AANOM,ARGPER,ECC,EROT,INC,OMEGA,PNODAL,PROT,SROT,
     1TAUPER,TOMEGA
      COMMON /3/ PI
      IF (FLAG.EQ.1.AND.ECC.GT.0.) GOTO 1
      RSAT=AANOM
      C=(TAU+PNODAL)/PNODAL
      GOTO 2
    1 C=(TAU-TAUPER+PNODAL)/PNODAL
    2 PHI=2.*PI*(C-FLOAT(IFIX(C)))
      IF (FLAG.EQ.0.OR.ECC.EQ.0.) GOTO 3
      E=PHI
    4 URSAT=1.-ECC*COS(E)
      D=(PHI-E+ECC*SIN(E))/URSAT
      E=E+D
      IF (ABS(D).LE.EERROR) GOTO 5
      GOTO 4
    5 RSAT=AANOM*URSAT
      NU=ACOS((COS(E)-ECC)/URSAT)
      IF (E.GT.PI) NU=PI*2.-NU
```

```
      PHI=NU+ARGPER
    3 IF (PHI.GT.2.*PI) PHI=PHI-2.*PI
      PSISAT=ASIN(SIN(PHI)*SIN(INC))
      DLAM=ASIN(TAN(PSISAT)/ABS(TAN(INC)))
      IF (PHI.GE.PI/2.0.AND.PHI.LT.1.5*PI) DLAM=PI-DLAM
      IF (INC.LT.PI/2.) DLAM=-DLAM
      LAMSAT=OMEGA+SROT*TAU+DLAM
      LAMSAT=ATAN2(SIN(LAMSAT),COS(LAMSAT))
      RETURN
      END
      SUBROUTINE STRACK(AZ,EL,LAMSAT,LAMTRC,PSISAT,PSITRC,RSAT,RTGC,SEP)
C     COMPUTE AZIMUTH  AND ELEVATION OF SATELLITE GIVEN SUB POINT
C     AND TRACKING STATION COORDINATES
C     AZ   =AZIMUTH OF SATELLITE   EL  =ELEVATION ABOVE HORIZON
      REAL LAMTRC,LAMSAT
      COMMON /3/ PI
      DLAM1=LAMTRC-LAMSAT
      IF (DLAM1.EQ.0.0.OR.ABS(DLAM1).EQ.PI.OR.ABS(DLAM1).EQ.PI/2.0)
     1DLAM1=DLAM1+1.0E-6
      DLAM2=ACOS(COS(DLAM1))
      COSSEP=SIN(PSISAT)*SIN(PSITRC)+COS(PSISAT)*COS(PSITRC)*COS(DLAM1)
      SEP=ACOS(COSSEP)
      IF (SEP.EQ.0.0.OR.ABS(SEP).EQ.PI) SEP=SEP+1.0E-6
      EL=ATAN((RSAT*COSSEP-RTGC)/RSAT/SIN(SEP))
      TANAZ=(TAN(PSISAT)/SIN(DLAM2)-TAN(PSITRC)/TAN(DLAM2))*COS(PSITRC)
      ONE=1.0
      AZ=ATAN2(ONE,TANAZ)
      IF (SIN(DLAM1).LT.0.) AZ=2.*PI-AZ
      RETURN
      END


3949.914015 .081813334 .2625161708
1.938111545 .00101 1.781547709 0.
-2.2019073 7.2333333 20 4 1978 7796
.8 1.407171709 .0615530303
1.6970417 0.8709192
0
-1.6256327 -1.1180248
1.1162795 1.6238873
3  1
4  5  1978
4  5  1978
1
5  5  1978
5  5  1978
2
6  5  1978
6  5  1978
2
```

APPENDIX 7

NOVA PROGRAM TO COPY 9-TRACK TO 9600 BAUD TAPE

```
                .TITLE PEEK
                .ENT START
                .NREL
                .ENT Q
;
START:  INTOS                   ;INTERRUPT DISABLE
        NIOS    TTO
        LDA     0,ZERO          ;SET UP BUFFER PARAMETERS
        LDA     2,STADD
        LDA     1,N3777
        STA     1,CNTR
Q:      LDA     0,STADD
        DOB     0,MTA           ;SEND START ADDRESS TO TAPE DRIVE
        LDA     0,ZERO
        DOC     0,MTA           ;SEND WORD COUNT TO TAPE DRIVE
;
;
        LDA     0,ZERO
        DOAS    0,MTA           ;SEND COMMAND AND START READING
;
        JSR     WATFON
        DIA     0,MTA           ;READ STATUS
        DIB     1,MTA           ;READ WORD COUNTER
        LDA     2,STADD
        SUB     2,1
        STA     1,BUFC          ;REAL WORD COUNT
        ISZ     BUFC
        STA     2,20
        DSZ     20              ;WORD COUNTER IN AUTO INCREMENT LOCATION
        LDA     1,MASK
        AND#    1,0,SZR         ;CHECK STATUS
        JMP     END             ;BAD RECORD READ
        LDA     0,SYNC          ;SEND SYNC BYTE TO SERIAL PORT
        JSR     TTYO
BEMPT:  LDA     0,@20           ;GET DATA FROM BUFFER
        MOVS    0,0             ;SWAP BYTES FOR HI ASCII CHAR
        JSR     TTYO            ;SEND TO SERIAL PORT
        MOVS    0,0             ;GET LO ASCII CHAR
        JSR     TTYO
        DSZ     BUFC            ;TEST FOR BUFFER SEND
        JMP     BEMPT           ;KEEP SENDING DATA
        DIA     0,TTI           ;CHECK FOR ESCAPE FROM TERMINAL
        NIOS    TTI             ;IF NOT PRESENT ,GET NEXT RECORD
        SUB     0,1,SNR         ;IF PRESENT ,FINISH SENDING PRESENT RECORD
        JMP     END
        JMP     START
;
WATFON: STA     3,TEMP
        LDA     0,MINI
        LDA     1,N256
        STA     1,CONTO         ;SEND 256 NULLS (FFH) AS INTER-RECORD GAP
```

```
HEADR:  JSR     TTYO
        DSZ     CONTO
        JMP     HEADR
        SKPBN   MTA         ;CHECK FOR RECORD READY FROM TAPE DRIVE
        JMP     @TEMP
EXTRT:  JSR     TTYO        ;IF NOT, WAIT UNTIL READY
        SKPBZ   MTA         ;IN THE MEANTIME ,SEND NULLS TO SERIAL PORT
        JMP     EXTRT
        JMP     @TEMP
;
TTYO:   SKPBZ   TTO1        ;SEND DATA TO SERIAL PORT FROM ACC #0
        JMP     TTYO        ;WAIT UNTIL READY FOR NEXT CHAR
        DOAS    0,TTO1
        JMP     0,3
;
END:    LDA     1,N5        ;COMPLETE SENDING PRESENT RECORD
        STA     1,CONTO
        LDA     0,MINI
EREC:   JSR     TTYO
        DSZ     CONTO
        JMP     EREC
        HALT
;
MASK:   0400
SYNC:   26
ESCA:   33
MINI:   -1
N256:   256.
N5:     5.
CONTO:  0
CONT1:  0
BUFC:   0
TEMP:   0
;
ASCII:  60
ONEA:   61
ZERA:   60
SPACA:  40
RETA:   15
LFA:    12
;
;
;
STADD:  BUFFER
ZERO:   0
N3777:  3777
CNTR:   0
                .LOC 1000
                .BLK 3777
        .END START
```

APPENDIX 8

PROGRAM TO READ NOAA 5 HEADER RECORD

```
0000 *PRØGRAM TØ READ NØAA 5 HEADER RECØRD
0010 *AND PRINT ØRBITAL DATA ØN CØNSØLE
0020  ØRG 3800H
0030 SP EQU 06H
0040 PSW EQU 06H
0050 GØ EQU 0F809H
0060 STACK EQU 3E80H
0070 ETX EQU 03H
0080 NUL EQU 0FFH
0090 SYN EQU 16H
0100 USTAT EQU 0F7H
0110 USCMD EQU 0F7H
0120 USDAI EQU 0F6H
0130  LXI SP,STACK
0140 *READ HEADER RECØRD
0150  CALL HSYNC
0160  CALL CRLF
0170 *PRINT "ØRBITAL DATA FØR NØAA5"
0180  CALL STRING
0190  DW 'RØ'
0200  DW 'IB'
0210  DW 'AT'
0220  DW ' L'
0230  DW 'AD'
0240  DW 'AT'
0250  DW 'F '
0260  DW 'RØ'
0270  DW 'N '
0280  DW 'AØ'
0290  DW 'SA'
0300  DB ETX
0310  CALL CRLF
0320 *PRINT "ØRBIT NUMBER:"
0330  CALL STRING
0340  DW 'RØ'
0350  DW 'IB'
0360  DW ' T'
0370  DW 'UN'
0380  DW 'EM'
0390  DW 'RE'
0400  DW ' :'
0410  DB ETX
0420 *GET ØRBIT NUMBER (16 BIT SIGNED INTEGER)
0430  LHLD RECØRD
0440  XCHG
0450 *CØNVERT TØ BCD-ASCII AND PRINT
0460  CALL BINBCD
0470  CALL ØUTLZS
0480  CALL CRLF
0490 *PRINT "ASCENDING NØDE:"
0500  CALL STRING
```

```
0510  DW 'SA'
0520  DW 'EC'
0530  DW 'DN'
0540  DW 'NI'
0550  DW ' G'
0560  DW 'ØN'
0570  DW 'ED'
0580  DW ' :'
0590  DB ETX
0600 *GET INTEGER PART ØF ASCENDING NØDE, PRINT
0610  LHLD RECØRD+2
0620  XCHG
0630  CALL BINBCD
0640  CALL ØUTLZS
0650 *PRINT DECIMAL PØINT
0660  MVI C,'.'
0670  CALL CØ
0680 *GET FRACTIØNAL PART ØF ASCENDING NØDE, PRINT
0690 *ASCENDING NØDE IN DEGREES
0700  LHLD RECØRD+4
0710  XCHG
0720  CALL ABS
0730  CALL BINBCD
0740  CALL ØUT2D
0750  CALL CRLF
0760 *PRINT "EQUATØR CRØSSING TIME: "
0770  CALL STRING
0780  DW 'QE'
0790  DW 'AU'
0800  DW 'ØT'
0810  DW ' R'
0820  DW 'RC'
0830  DW 'SØ'
0840  DW 'IS'
0850  DW 'GN'
0860  DW 'T '
0870  DW 'MI'
0880  DW ':E'
0890  DB ' '
0900  DB ETX
0910 *PØINT TØ TIME IN BUFFER
0920  LXI H,RECØRD+6
0930 *PRINT TIME
0940  CALL DHMS
0950  CALL CRLF
0960 *PRINT "ACQUISTIØN TIME: "
0970  CALL STRING
0980  DW 'CA'
0990  DW 'UQ'
1000  DW 'SI'
1010  DW 'TI'
```

```
1020  DW 'OI'
1030  DW ' N'
1040  DW 'IT'
1050  DW 'EM'
1060  DW ' :'
1070  DB ETX
1080  LXI H,RECORD+14
1090 *PRINT TIME
1100  CALL DHMS
1110  CALL CRLF
1120 *PRINT "DATA TAPE TYPE: "
1130  CALL STRING
1140  DW 'AD'
1150  DW 'AT'
1160  DW 'T '
1170  DW 'PA'
1180  DW ' E'
1190  DW 'YT'
1200  DW 'EP'
1210  DW ' :'
1220  DB ETX
1230  LHLD RECORD+28
1240 *PRINT NUMBER
1250  XCHG
1260  CALL BINBCD
1270  CALL OUT2D
1280 *PRINT "TAPE #: "
1290  CALL CRLF
1300  CALL STRING
1310  DW 'AT'
1320  DW 'EP'
1330  DW '# '
1340  DW ' :'
1350  DB ETX
1360  LHLD RECORD+38
1370  XCHG
1380  CALL BINBCD
1390  CALL OUT2D
1400  CALL CRLF
1410 *RETURN TO MONITOR
1420  CALL 0
1430 *SUBROUTINE TO PRINT CHAR STRING
1440 *GET RETURN ADDRESS WHICH POINTS TO CHARCTER STRING
1450 STRING POP H
1460  MOV A,M
1470  INX H
1480  CPI ETX
1490 *LOOP UNTIL ETX CHAR FOUND
1500  JZ RTN
1510  MOV C,A
1520 *PRINT C REG ON CONSOLE
1530  CALL CO
```

```
1540  JMP STRING+1
1550 RTN PUSH H
1560  RET
1570 *SUBRØUTINE TØ CØNVERT 16-BIT BINARY
1580 *TØ BCD-ASCII
1590 *D,E CØNTAINS SIGNED 16-BIT INTE GER
1600 BINBCD CALL SWAP
1610  LXI H,BUFF
1620  MØV A,D
1630  ØRA A
1640 *STØRE 6 CHARACTERS STARTING AT BUFF
1650 *CHECK SIGN BIT
1660 *IF +VE, STØRE SPACE ;IF -VE, STØRE "-".
1670  MVI M,' '
1680  JP ØVER
1690  MVI M,'-'
1700  CALL CØMP
1710 ØVER INX H
1720 *BREAK DE INTØ 5 BCD DIGITS
1730  LXI B,10000
1740  CALL DIGIT
1750  LXI B,1000
1760  CALL DIGIT
1770  LXI B,100
1780  CALL DIGIT
1790  LXI B,10
1800  CALL DIGIT
1810  LXI B,1
1820  CALL DIGIT
1830  RET
1840 *SUBRØUTINE TØ SUBTRACT BC FRØM DE AND INCREMENT
1850 *MEMØRY (CHARACTER BUFFER VIA HL)
1860 *UNTIL RESULT -VE
1870 DIGIT MVI M,30H
1880 DI0 MØV A,E
1890  SUB C
1900  MØV E,A
1910  MØV A,D
1920  SBB B
1930  MØV D,A
1940  JM DI1
1950  INR M
1960  JMP DI0
1970 *RESULT -VE, RESTØRE
1980 DI1 MØV A,E
1990  ADD C
2000  MØV E,A
2010  MØV A,D
2020  ADC B
2030  MØV D,A
2040  INX H
```

```
2050  RET
2060 BUFF DS 6
2070 *SUBRØUTINE TØ SWAP D AND E
2080 SWAP  MØV A,D
2090  MØV D,E
2100  MØV E,A
2110  RET
2120 *SUBRØUTINE TØ PRINT BCD NUMBERS
2130 *WITH LEADING ZERØ SUPPRESSIØN
2140 ØUTLZS LXI H,BUFF
2150  MVI B,5
2160  MØV A,M
2170  INX H
2180 *TEST FØR MINUS SIGN
2190  CPI '-'
2200  JNZ NCHAR
2210  MØV C,A
2220  CALL CØ
2230 *SEARCH FØR NØN ZERØ CHARACTER
2240 *IF LAST (5TH) CHAR IS ZERØ, PRINT IT
2250 NCHAR MØV A,M
2260  INX H
2270  CPI '0'
2280  JNZ NZERØ
2290  DCR B
2300  JNZ NCHAR
2310  MØV C,A
2320  CALL CØ
2330  RET
2340 *PRINT REMAINING DIGITS
2350 NZERØ DCX H
2360 NDIG MØV C,M
2370  INX H
2380  CALL CØ
2390  DCR B
2400  JNZ NDIG
2410  RET
2420 *SUBRØUTINE TØ FIND TWØ'S CØMPLEMENT ØF D,E
2430 CØMP MØV A,D
2440  CMA
2450  MØV D,A
2460  MØV A,E
2470  CMA
2480  MØV E,A
2490  INX D
2500  RET
2510 *SUBRØUTINE TØ FIND ABS(D,E)
2520 ABS MØV A,D
2530  ØRA A
2540  RP
2550  CALL CØMP
2560  RET
```

```
2570 *SUBROUTINE TO OUTPUT ONLY
2580 *2 BCD DIGITS, WITH NO LEADING ZERO SUPPRESSION
2590 OUT2D LXI H,BUFF+4
2600  MOV C,M
2610  CALL CO
2620  INX H
2630  MOV C,M
2640  CALL CO
2650  RET
2660 *SUBROUTINE FOR CR AND LF
2670 CRLF MVI C,0DH
2680  CALL CO
2690  MVI C,0AH
2700  CALL CO
2710 *PRINT TWO NULLS
2720  MVI C,00H
2730  CALL CO
2740  CALL CO
2750  RET
2760 *SUBROUTINE TO PRINT DAY,HR,MIN,SEC
2770 *SAVE BUFFER ADDRESS
2780 DHMS SHLD HLSTR
2790  CALL NPRINT
2800  CALL STRING
2810  DW ' D'
2820  DB ETX
2830  CALL NPRINT
2840  CALL STRING
2850  DW ' H'
2860  DB ETX
2870  CALL NPRINT
2880  CALL STRING
2890  DW ' M'
2900  DB ETX
2910  CALL NPRINT
2920  CALL STRING
2930  DW ' S'
2940  DB 'Z'
2950  DB ETX
2960  RET
2970 *SUBROUTINE TO PRINT SIGNED 16-BIT INTEGER
2980 *IN BCD-ASCII FORMAT ON CONSOLE
2990 *ADDRESS OF INTEGER IN LOC HLSTR
3000 NPRINT LHLD HLSTR
3010 *GET INTEGER INTO DE
3020  MOV E,M
3030  INX H
3040  MOV D,M
3050  INX H
3060 *SAVE ADDRESS OF NEXT NUMBER
3070  SHLD HLSTR
```

```
3080 *CØNVERT INTEGER TØ BCD-ASCII DIGIT STRING
3090  CALL BINBCD
3100 *PRINT WITH LEADING-ZERØ SUPPRESSIØN
3110  CALL ØUTLZS
3120  RET
3130 HLSTR DS 2
3140 *READ HEADER RECØRD
3150 *SYNC UART AND TAPE INTERFACE TØ LEADER
3160 HSYNC CALL SYNC
3170 *PØINT TØ MEMØRY BUFFER
3180  LXI H,RECØRD
3190 *READ 50 BYTES
3200  MVI E,50
3210 *READ BYTE FRØM TAPE
3220 HRD CALL RX
3230 *CY SET IF FRAMING ERRØR
3240  JC HERR
3250  MØV M,A
3260  INX H
3270  DCR E
3280  JNZ HRD
3290  RET
3300 *PRINT "?" IF FRAMING ERRØR AND RETURN TØ MØNITØR
3310 HERR CALL STRING
3320  DB '?'
3330  DB ETX
3340  CALL 0
3350 *SUBRØUTINE TØ FIND START ØF RECØRD
3360 *INITIALIZE UART
3370 SYNC CALL RXEN
3380 *SEARCH FØR AT LEAST 240 NULLS
3390  MVI C,240
3400  CALL RX
3410  JC SYNC
3420  CPI NUL
3430  JNZ SYNC+3
3440  DCR C
3450  JNZ SYNC+5
3460  MVI C,20
3470 WSYNC CALL RX
3480  JC SYNC
3490  CPI SYN
3500 *WAIT FØR SYNC CHAR
3510  RZ
3520  CPI NUL
3530  JNZ SYNC+3
3540  DCR C
3550  JNZ WSYNC
3560 *SET CY IF MØRE THAN 256 NULLS, RETURN
3570 *INDICATES END ØF FILE
3580  STC
3590  RET
```

```
3600 *SUBRØUTINE TØ READ DATA FRØM UART
3610 RX IN USTAT
3620  MØV B,A
3630  ANI 02H
3640 *WAIT FØR DATA READY STATUS
3650  JZ RX
3660  MØV A,B
3670  ANI 20H
3680  JZ $+1
3690 *SET CY IF FRAMING ERRØR
3700  STC
3710 *READ DATA
3720  IN USDAI
3730  RET
3740 *SUBRØUTINE TØ INITIALIZE UART
3750 *TØ 9600 BAUD, 1 STØP BIT, 8 DATA BITS
3760 RXEN XRA A
3770  ØUT USCMD
3780  ØUT USCMD
3790  ØUT USCMD
3800  MVI A,40H
3810  ØUT USCMD
3820  MVI A,4EH
3830  ØUT USCMD
3840  MVI A,16H
3850  ØUT USCMD
3860  RET
3870 *STØRAGE FØR DATA FRØM TAPE
3880 RECØRD DS 50
3890 LAST END
```

```
 10 PRINT "ORBITAL DATA FOR NOAA5"
 20 A=9999
 30 DEF FN W(Z)=256*PEEK(Z)+PEEK(Z+1)
 40 PRINT "ORBIT NUMBER";FN W(A+1)
 50 PRINT "ASCENDING NODE: ";FN W(A+3);".";FN W(A+5)
 60 PRINT "EQUATOR CROSSING TIME: ";
 70 PRINT FN W(A+7);"D ";FN W(A+9);"H ";FN W(A+11);"M ";FN W(A+13);"S Z"
 80 PRINT "ACQUISITION TIME: ";
 90 PRINT FN W(A+15);"D ";FN W(A+17);"H ";FN W(A+19);"M ";FN W(A+21);
100 PRINT "S Z"
110 PRINT "DATA TAPE TYPE: ";PEEK(A+30)
120 PRINT "TAPE #: ";PEEK(40)
130 END
```

APPENDIX 9

PROGRAM TO READ RAW DATA TAPES

```
0000 *PROGRAM FOR SATELLITE DATA ACQUISITION FROM TAPE
0010 *DATA IS STORED ON TAPE ,1 RECORD PER LINE
0020 *AT LEAST 256 NUL CHARACTERS BETWEEN RECORDS
0030 *DATA AT 9600 BAUD, 8 BITS PER CHARACTER
0040 *SYNC BYTE STARTS RECORD
0050 *65536 BYTES ARE READ AND STORED IN IMAGE MEMORY
0060 *256 BYTES PER RECORD AND 256 RECORDS ARE READ
0070 *USING HORIZONTAL AND VERTICAL SCALE FACTORS
0080 *NEAREST NEIGHBOUR TECHNIQUE IS USED TO SELECT
0090 *DATA POINT FROM TAPE
0100 *8 BIT DATA IS READ INTO CHAN 1 AND THE 4 MSB'S
0110 *ARE DISPLAYED ON SCREEN
0120  ORG 3400H
0130 CO EQU 0F809H
0140 CI EQU 0F803H
0150 STACK EQU 3EBFH
0160 CR EQU 0DH
0170 LF EQU 0AH
0180 ETX EQU 03H
0190 NUL EQU 0FFH
0200 SYN EQU 16H
0210 MEMCON EQU 80H
0220 MRD EQU 83H
0230 MWR EQU 83H
0240 LOADR EQU 82H
0250 HIADR EQU 81H
0260 USTAT EQU 0F7H
0270 USCMD EQU 0F7H
0280 USDAT EQU 0F6H
0290 CRLF EQU 0A0DH
0300 SP EQU 6
0310 PSW EQU 6
0320 TTO EQU 0F4H
0330 HLENG EQU 2002
0340 RI EQU 0F806H
0350 *INITIALIZATION
0360 *CLEAR SCREEN MEMORY AND CHAN 1
0370  LXI SP,STACK
0380  MVI A,0
0390  OUT MEMCON
0400  LXI H,0
0410  SHLD VLMA
0420 *CLEAR ONLY LOWER HALF (SCREEN) ON CHAN 0 BYTES
0430 CLRTV CALL READ
0440  ANI 0F0H
0450  CALL WRITE
0460  CALL INCADR
0470  JNZ CLRTV.
0480 *CLEAR CHAN 1
0490  MVI A,04H
0500  OUT MEMCON
```

```
0510 CLRCHI MVI A,0
0520  CALL WRITE
0530  CALL INCADR
0540  JNZ CLRCHI
0550 *ENTER DATA FRØM CØNSØLE
0560  LXI H,S1
0570 DNIN CALL STRING
0580 *PRINT "DAY ØR NIGHT PASS?"
0590  CALL KEYIN
0600 *INPUT CHARACTER "D" ØR "N" AND CARRIAGE RETURN
0610  CPI 'D'
0620  JZ SUID
0630  CPI 'N'
0640  JNZ DNIN
0650 SUID STA DNFLAG
0660 *SET DNFLAG TØ "D" ØR "N"
0670  LXI H,S2
0680 *PRINT "ØRBIT NUMBER="
0690  CALL STRING
0700 *INPUT ØRBIT # AND CØNVERT TØ 16-BIT INTEGER
0710  CALL BCDIN
0720  SHLD ØRNUM
0730  LXI H,S3
0740 *PRINT "START CØLUMN="
0750 ISTCØL CALL STRING
0760 *INPUT AND CØNVERT TØ BINARY
0770  CALL BCDIN
0780  SHLD STCØL
0790  LXI H,S5
0800 *PRINT "LINE # SINCE AØS="
0810  CALL STRING
0820 *INPUT AND CØNVERT TØ BINARY
0830  CALL BCDIN
0840  SHLD LINENØ
0850  LXI H,S7
0860 VIIN CALL STRING
0870 *PRINT "VISUAL ØR IR?"
0880  CALL KEYIN
0890 *INPUT "V" ØR "I"
0900  CPI 'V'
0910  JZ STAVI
0920  CPI 'I'
0930  JNZ VIIN
0940 *SET VISUAL/IR FLAG
0950 STAVI STA VIFLAG
0960  LXI H,S8
0970 *PRINT "RECØRD LENGTH="
0980  CALL STRING
0990 *INPUT AND CØNVERT TØ BINARY
1000  CALL BCDIN
1010  SHLD RLENG
```

```
1020  LXI H,S9
1030 *PRINT "START TAPE G,"
1040  CALL STRING
1050 *RETURN TO MONITOR
1060  CALL 0
1070 *START TAPE AT BEGINNING OF FILE, TYPE G TO
1080 *CONTINUE PROGRAM
1090  MVI A,0
1100  OUT MEMCON
1110 *SYNC TO TAPE AND WAIT FOR NEXT RECORD (HEADER)
1120 HSYNC CALL SYNCI
1130 *READ 1ST TWO DATA BYTES FROM TAPE (ORBIT #)
1140  CALL RX
1150  JC HERR
1160  MOV D,A
1170  CALL RX
1180  JC HERR
1190  MOV E,A
1200  LHLD ORNUM
1210 *COMPARE ORBIT # READ AND REQUIRED (HL-DE)
1220  CALL COMPD
1230  DAD D
1240  MOV A,H
1250  ORA L
1260  JZ ORBOK
1270 *PRINT "WRONG ORBIT"
1280  LXI H,S10
1290  CALL STRING
1300 *RETURN TO MONITOR
1310  CALL 0
1320 *REPOSITION TAPE TO START OF PROPER FILE
1330 *TYPE G AND PROGRAM REPEATS TEST
1340  JMP HSYNC
1350 *CORRECT FILE FOUND
1360 ORBOK MVI A,'R'
1370 *TYPE "R" ON TTY TO INDICATE CORRECT FILE
1380  OUT TTO
1390 *GET HEADER RECORD LENGTH
1400  LXI H,HLENG
1410 *WAIT FOR END OF HEADER
1420 HEND CALL RX
1430  JC HERR
1440  DCX H
1450  MOV A,H
1460  ORA L
1470  JNZ HEND
1480 *INITIALIZE VARIABLES
1490  LXI H,0
1500  MVI A,0
1510  STA LINE
1520  SHLD RSAML
```

```
1530  SHLD RSAML+2
1540  SHLD VLMA
1550  SHLD PREC
1560  LDA DNFLAG
1570 *IF DAY PASS, START AT TØP-LEFT CØRNER ØF SCREEN AND
1580 *READ NØRMALLY
1590 *IF NIGHT PASS, START AT BØTTØM-RIGHT CØRNER
1600 *AND READ INTØ MEMØRY BACKWARDS
1610  CPI 'D'
1620  JZ $+4
1630  DCX H
1640  SHLD VLMA
1650  LHLD RLENG
1660 *SET RLØNG INITIALLY TØ RECØRD LENGTH
1670  SHLD RLØNG
1680 *SEARCH FØR STARTING RECØRD
1690 TAPE LHLD LINENØ
1700 *INITIALIZE RNUM (RECØRD NUMBER) TØ STARTING RECØRD
1710  SHLD RNUM
1720 *SEARCH FØR RECØRD # =RNUM
1730 *SYNC TØ NEXT RECØRD
1740 NXREC CALL SYNC
1750 *CY IS SET IF END ØF TAPE
1760  JC EØT
1770  LHLD RLØNG
1780 *READ 1ST TWØ BYTES ØF RECØRD (RECØRD NUMBER)
1790  CALL RX
1800 *CY IS SET IF FRAMING ERRØR
1810  JC BADRD
1820  DCX H
1830  MØV D,A
1840  CALL RX
1850  JC BADRD
1860  DCX H
1870  MØV E,A
1880 *SAVE PRESENT RECØRD BYTE CØUNT
1890  SHLD BNUM
1900  LHLD RNUM
1910  XCHG
1920  SHLD PREC
1930  CALL CØMPD
1940 *CØMPARE LINE # READ WITH STARTING LINE #
1950  DAD D
1960  JC ZTST
1970 *LINE # READ <STARTING LINE #, KEEP READING RECØRDS
1980  LHLD BNUM
1990 *WAIT FØR END ØF RECØRD
2000 ENDWT CALL RX
2010  JC NXREC
2020  DCX H
2030  MØV A,H
```

```
2550  LXI D,2000
2560 IRSKP CALL RX
2570  DCX H
2580  DCX D
2590  JC DATER
2600  MOV A,D
2610  ORA E
2620  JNZ IRSKP
2630 *SKIP TO COLUMN START
2640 OVERI XCHG
2650  LHLD STCOL
2660  XCHG
2670 COLWT CALL RX
2680  DCX H
2690  DCX D
2700  JC DATER
2710  MOV A,D
2720  ORA E
2730  JNZ COLWT
2740 *START READING LINE
2750  SHLD BNUM
2760  LXI H,0
2770  SHLD RSAMP
2780  SHLD RSAMP+2
2790  MVI A,0
2800  STA COLUM
2810 RDBYT LHLD BNUM
2820  CALL RX
2830  DCX H
2840  SHLD BNUM
2850  STA PIXEL
2860  JC DATER
2870  LXI H,RSAMP
2880 *ROUND INTEGER PART OF RSAMP TO NEAREST VALUE
2890  CALL ROUND
2900  LXI H,RSAMP
2910 *RSAMP AND INCP HAVE 1 BYTE INTEGER, 2 BYTES FOR FRACTION
2920  LXI D,INCP
2930  CALL ADD4
2940  LDA X
2950  MOV B,A
2960  LDA COLUM
2970  CMP B
2980 *READ DATA BYTES UNTIL COLUMN#=INT(RSAMP)
2990  JNZ RDBYT
3000 *STORE DATA BYTE IN CHAN 1 AND ON SCREEN
3010  CALL READ
3020  ANI 0F0H
3030  MOV B,A
3040  LDA PIXEL
3050  ANI 0FH
```

```
2040  ØRA L
2050  JNZ ENDWT
2060 *END ØF RECØRD, GET NEXT RECØRD
2070  JMP NXREC
2080 *TEST IF PREC (PRESENT RECØRD) =RNUM
2090 ZTST MØV A,H
2100  ØRA L
2110  JZ LNØK
2120 *PREC >STARTING RECØRD # (RNUM)
2130 *PRINT "LINE NØT FØUND"
2140  LXI H,S11
2150  CALL STRING
2160 INTØ MVI C,'='
2170 *PRINT VALUE (IN DECIMAL) ØF CURRENT RECØRD #
2180  CALL CØ
2190  LHLD PREC
2200  XCHG
2210  CALL BINBCD
2220  CALL ØUTLZS
2230 *RETURN TØ MØNITØR
2240  CALL 0
2250 *REWIND TAPE TØ BEFØRE STARTING RECØRD WILL BE FØUND
2260  JMP NXREC
2270 *END ØF TAPE
2280 *PRINT "END ØF TAPE"
2290 EØT LXI H,S14
2300  CALL STRING
2310  CALL 0
2320 *LØAD NEXT TAPE AND CØNTINUE
2330  CALL SYNCI
2340  JMP NXREC+6
2350 *READ ERRØR IN LINE CØUNTER
2360 BADRD LXI H,S12
2370  CALL STRING
2380  JMP INTØ
2390 *STARTING RECØRD FØUND
2400 *TYPE NUL FØR EACH RECØRD READ
2410 LNØK MVI A,0
2420  ØUT TTØ
2430 *WAIT FØR IR DATA
2440  LHLD BNUM
2450  MVI E,12
2460 DWAIT CALL RX
2470  DCX H
2480  JC DATER
2490  DCR E
2500  JNZ DWAIT
2510 *IF VIS, SKIP IR
2520  LDA VIFLAG
2530  CPI 'V'
2540  JNZ ØVERI
```

```
3060  0RA B
3070  CALL WRITE
3080 *P0INT T0 NEXT MEM0RY L0CATI0N
3090  LHLD VLMA
3100  INX H
3110  LDA DNFLAG
3120  CPI 'D'
3130  JZ $+2
3140  DCX H
3150  DCX H
3160  SHLD VLMA
3170  LDA C0LUM
3180 *INCREMENT C0LUMN
3190  INR A
3200  STA C0LUM
3210  JNZ RDBYT
3220 *LINE C0MPLETE
3230  LDA LINE
3240  INR A
3250  STA LINE
3260  LXI H,RSAML
3270  LXI D,INCL
3280 *C0MPUTE NEXT REC0RD # (RNUM) NEAREST T0
3290 *DESIRED L0CATI0N
3300 *INCL AND RSAML HAVE 2 BYTE INTEGER PART, 1 BYTE FRAC
3310  CALL ADD4
3320  LXI H,RSAML
3330  CALL R0UND+1
3340 *R0UND INTEGER PART
3350  M0V E,A
3360  INX H
3370  M0V A,M
3380  ACI 0
3390  M0V D,A
3400  LHLD LINEN0
3410  DAD D
3420  SHLD RNUM
3430  LHLD BNUM
3440 *WAIT F0R END 0F REC0RD
3450 E0R CALL RX
3460  DCX H
3470  JC NXREC
3480  M0V A,L
3490  0RA H
3500  JNZ E0R
3510 *TEST F0R LAST LINE
3520  LDA LINE
3530  0RA A
3540  JNZ NXREC
3550 *PRINT "READ D0NE"
3560  LXI H,S15
```

```
3570  CALL STRING
3580  CALL 0
3590 *SCALE FACT0RS F0R IMAGE SIZE 300 X 300 NMI.
3600 *N0AA 5
3610 INCL DW 88D9H
3620  DB 02H
3630 INCP DW 5D7DH
3640  DB 6CH
3650 *DATA READ ERR0R
3660 DATER LXI H,S12
3670  CALL STRING
3680  MVI C,'='
3690  CALL C0
3700  LHLD PREC
3710  XCHG
3720  CALL BINBCD
3730  CALL 0UTLZS
3740  MVI C,' '
3750  CALL C0
3760  LXI H,S16
3770 *PRINT "BYTE" AND BYTE # IN REC0RD
3780  CALL STRING
3790  MVI C,'='
3800  CALL C0
3810  LHLD BNUM
3820  XCHG
3830  CALL BINBCD
3840  CALL 0UTLZS
3850  CALL 0
3860 *BACKUP TAPE AND RE-READ
3870  JMP NXREC
3880 *HEADER READ ERR0R
3890 HERR LXI H,S12
3900  CALL STRING
3910  LXI H,S17
3920  CALL STRING
3930  CALL 0
3940 *"HEADER"
3950 S17 DW 'H '
3960  DW 'AE'
3970  DW 'ED'
3980  DB 'R'
3990  DB ETX
4000 *WAIT F0R REC0RD START
4010 SYNCI CALL RXEN
4020  MVI C,240
4030  CALL RX
4040  JC SYNCI
4050  CPI NUL
4060  JNZ SYNCI+3
4070  DCR C
```

```
4080  JNZ SYNCI+5.
4090 SYNWT CALL RX
4100  JC SYNCI
4110  CPI SYN
4120  JNZ SYNWT
4130  RET
4140 *LINKAGE ADDRESS TABLE
4150 READ EQU 03800H
4160 RAGAI EQU 03809H
4170 WRITE EQU 03813H
4180 WAGAI EQU 0381DH
4190 RCHK EQU 03820H
4200 INCAD EQU 0382EH
4210 KEYIN EQU 03838H
4220 CRIN EQU 0383EH
4230 STRIN EQU 03857H
4240 SPACE EQU 03867H
4250 BCDIN EQU 03876H
4260 FILL3 EQU 0387BH
4270 BACK EQU 03882H
4280 MOVUP EQU 03897H
4290 BCD EQU 038A2H
4300 DIG EQU 038BBH
4310 LOOP2 EQU 038CFH
4320 SKIP EQU 038D4H
4330 X10 EQU 038DEH
4340 SYNC EQU 038E9H
4350 WSYNC EQU 03906H
4360 IDLE EQU 03912H
4370 RXEN EQU 03915H
4380 COMPD EQU 03929H
4390 BINBC EQU 03931H
4400 OVER EQU 03943H
4410 DIGIT EQU 03963H
4420 DI0 EQU 03965H
4430 DI1 EQU 03972H
4440 SWAP EQU 0397AH
4450 OUTLZ EQU 0397EH
4460 NCHAR EQU 0398EH
4470 NZERO EQU 0399EH
4480 NDIG EQU 0399FH
4490 ROUND EQU 039A9H
4500 ADD4 EQU 039B5H
4510 RX EQU 039C6H
4520 S1 EQU 039E6H
4530 S2 EQU 039F9H
4540 S3 EQU 03A07H
4550 S4 EQU 03A15H
4560 S5 EQU 03A1CH
4570 S6 EQU 03A2EH
4580 S7 EQU 03A36H
4590 S8 EQU 03A45H
4600 S9 EQU 03A54H
4610 S10 EQU 03A61H
4620 S11 EQU 03A6DH
4630 S12 EQU 03A7CH
4640 S14 EQU 03A91H
4650 S15 EQU 03A9DH
4660 S16 EQU 03AA7H
4670 *COMMON VARIABLES
4680  ORG 3B00H
4690 VLMA DS 2
4700 DNFLAG DS 1
4710 BCDBUF DS 5
4720 COUNT DS 1
4730 ORNUM DS 2
4740 STCOL DS 2
4750 LINENO DS 2
4760 LENGTH DS 2
4770 ACCUM DS 2
4780 TEMP DS 2
4790 FACTOR DS 2
4800 WIDTH DS 2
4810 VIFLAG DS 1
4820 RLENG DS 2
4830 BUFF DS 5
4840 RNUM DS 2
4850 RLONG DS 2
4860 BNUM DS 2
4870 PREC DS 2
4880 LINE DS 1
4890 RSAML DS 4
4900 RSAMP DS 4
4910 COLUM DS 1
4920 X DS 1
4930 PIXEL DS 1
4940 TIMER DS 1
4950  END
```

```
0000 *SUBRØUITNES FØR SATELLITE DATA ACQUISITIØN PRØGRAM
0010 *LINKED TØ MAIN PRØGRAM THRØUGH SYMBØL TABLE
0020 *AND LINKER PRØGRAM
0030  ØRG 3800H
0040 CØ EQU 0F809H
0050 CI EQU 0F803H
0060 STACK EQU 3EBFH
0070 CR EQU 0DH
0080 LF EQU 0AH
0090 ETX EQU 03H
0100 NUL EQU 0FFH
0110 SYN EQU 16H
0120 MEMCØN EQU 80H
0130 MRD EQU 83H
0140 MWR EQU 83H
0150 LØADR EQU 82H
0160 HIADR EQU 81H
0170 USTAT EQU 0F7H
0180 USCMD EQU 0F7H
0190 USDAI EQU 0F6H
0200 CRLF EQU 0A0DH
0210 SP EQU 6
0220 PSW EQU 6
0230 TTØ EQU 0F4H
0240 HLENG EQU 2002
0250 *SUBRØUTINES
0260 *SUBRØUTINE TØ READ IMAGE MEMØRY VIA VLMA
0270 READ LHLD VLMA
0280  MØV A,H
0290  ØUT HIADR
0300  MØV A,L
0310  ØUT LØADR
0320 RAGAIN IN MRD
0330  MØV B,A
0340  IN MRD
0350  CMP B
0360  JNZ RAGAIN
0370  RET
0380 *SUBRØUTINE TØ WRITE TØ IMAGE MEMØRY VIA VLMA
0390 *WRITE AND READ BACK UNTIL BØTH SAME
0400 WRITE LHLD VLMA
0410  MØV B,A
0420  MØV A,H
0430  ØUT HIADR
0440  MØV A,L
0450  ØUT LØADR
0460 WAGAIN MØV A,B
0470  ØUT MWR
0480 RCHK IN MRD
0490  MØV C,A
0500  IN MRD
```

```
0510  CMP C
0520  JNZ RCHK
0530  CMP B
0540  JNZ WAGAIN
0550  RET
0560 *SUBROUTINE TO INC VLMA AND SET Z FLAG
0570 INCADR LHLD VLMA
0580  INX H
0590  SHLD VLMA
0600  MOV A,H
0610  ORA L
0620  RET
0630 *SUBROUTINE TO INPUT AND ECHO CHAR FROM CONSOLE
0640 *THEN WAIT FOR CARRIAGE RETURN
0650 KEYIN CALL CI
0660  ANI 7FH
0670  MOV B,A
0680 CRIN MOV C,A
0690  CALL CO
0700  CALL CI
0710  ANI 7FH
0720  CPI CR
0730  JNZ CRIN
0740 *ECHO CR AND LF
0750  MOV C,A
0760  CALL CO
0770  MVI C,LF
0780  CALL CO
0790  MOV A,B
0800  RET
0810 *SUBROUTINE TO TYPE CHAR STRING
0820 *H,L POINTS TO START OF STRING
0830 *STRING TERMINATED BY 03 (ETX)
0840 *IF MSB (B7)=1 ,THEN PRINT NO OF SPACES
0850 *= VALUE OF 7 LSB'S
0860 STRING MOV A,M
0870  INX H
0880  CPI ETX
0890  RZ
0900  ORA A
0910  JM SPACES
0920  MOV C,A
0930  CALL CO
0940  JMP STRING
0950 SPACES ANI 7FH
0960  MOV B,A
0970  MVI C,' '
0980  CALL CO
0990  DCR B
1000  JNZ SPACES+3
1010  JMP STRING+1
```

```
1020 *SUBROUTINE TO INPUT ASCII-BCD
1030 *DIGITS AND CONVERT TO 16-BIT BIN
1040 *LAST 5 DIGITS TAKEN AS NUMBER
1050 *LEADING ZEROS NOT REQUIRED
1060 *UNSIGNED INTEGER
1070 *CLEAR CHARACTER INPUT BUFFER TO ZERO
1080 BCDIN LXI H,BCDBUF
1090  MVI B,5
1100 FILL30 MVI M,'0'
1110  INX H
1120  DCR B
1130  JNZ FILL30
1140 BACK LXI H,BCDBUF
1150 *INPUT CHAR AND ECHO
1160  CALL CI
1170  ANI 7FH
1180  MOV B,A
1190  MOV C,A
1200  CALL CO
1210  MOV A,B
1220 *INPUT CHARS TO BUFFER UNTIL CR
1230  CPI CR
1240  JZ BCD
1250  MVI C,5
1260 *MOVE CHARS IN BUFFER UP ONE BYTE AND
1270 *ENTER NEW CHAR IN 1ST POSITION
1280 *ELIMINATE LAST CHAR
1290 MOVUP MOV A,M
1300  MOV M,B
1310  MOV B,A
1320  INX H
1330  DCR C
1340  JNZ MOVUP
1350  JMP BACK
1360 *TYPE LF
1370 BCD MVI C,LF
1380  CALL CO
1390 *CONVERT ASCII-BCD TO 16-BIT BINARY
1400  LXI H,0
1410 *CLEAR INITIAL SUM
1420  SHLD ACCUM
1430  LXI H,1
1440 *INITIALIZE FACTOR TO UNITS DIGIT
1450  SHLD FACTOR
1460  LXI H,BCDBUF
1470 *LOC TEMP HOLDS BUFFFER ADDRESS
1480  SHLD TEMP
1490  MVI B,5
1500 DIG LHLD TEMP
1510 *GET DIGIT FROM BUFFER
1520  MOV A,M
```

```
1530  INX H
1540  SHLD TEMP
1550  LHLD FACTØR
1560  XCHG
1570  LHLD ACCUM
1580 *CØNVERT ASCII TØ BCD
1590  SUI '0'
1600  JZ SKIP
1610 *MULTIPLTY DIGIT VALUE BY FACTØR, ADD TØ SUM
1620 LØØP2 DAD D
1630  DCR A
1640  JNZ LØØP2
1650 SKIP SHLD ACCUM
1660  DCR B
1670  RZ
1680 *MULTIPLY FACTØR BY 10
1690  LXI H,0
1700  MVI A,10
1710 X10 DAD D
1720  DCR A
1730  JNZ X10
1740  SHLD FACTØR
1750  JMP DIG
1760 *SUBRØUTINE TØ FIND START ØF RECØRD
1770 *INITIALIZE UART
1780 SYNC CALL RXEN
1790 *SEARCH FØR AT LEAST 240 NULLS
1800  MVI C,240
1810  CALL RX
1820  JC SYNC
1830  PUSH PSW
1840 *IF TIMER ØVERFLØWS, NØ MØRE DATA (END ØF TAPE)
1850  LDA TIMER
1860  ØRA A
1870  JZ IDLE
1880  PØP PSW
1890  CPI NUL
1900  JNZ SYNC+3
1910  DCR C
1920  JNZ SYNC+5
1930 WSYNC CALL RX
1940 *WAIT FØR SYNC BYTE
1950  JC SYNC
1960  CPI SYN
1970  JNZ WSYNC
1980  RET
1990 IDLE PØP PSW
2000  STC
2010  RET
2020 *SUBRØUTINE TØ INITIALIZE UART
2030 *FØR 9600 BAUD
```

```
2040 RXEN XRA A
2050  OUT USCMD
2060  OUT USCMD
2070  OUT USCMD
2080  MVI A,40H
2090  OUT USCMD
2100  MVI A,4EH
2110  OUT USCMD
2120  MVI A,16H
2130  OUT USCMD
2140  RET
2150 *SUBROUTINE TO COMPLEMENT D,E
2160 COMPD MOV A,D
2170  CMA
2180  MOV D,A
2190  MOV A,E
2200  CMA
2210  MOV E,A
2220  INX D
2230  RET
2240 *SUBROUTINE TO CONVERT 16-BIT BINARY
2250 *TO BCD-ASCII
2260 *BINARY WORD IS IN D,E ,SIGNED INTEGER
2270 *ASCII-BCD DIGITS STORED IN BUFFER
2280 BINBCD CALL SWAP
2290  LXI H,BUFF
2300  MOV A,D
2310  ORA A
2320  MVI M,' '
2330  JP OVER
2340  MVI M,'-'
2350  CALL COMPD
2360 OVER INX H
2370  LXI B,10000
2380  CALL DIGIT
2390  LXI B,1000
2400  CALL DIGIT
2410  LXI B,100
2420  CALL DIGIT
2430  LXI B,10
2440  CALL DIGIT
2450  LXI B,1
2460  CALL DIGIT
2470  RET
2480 DIGIT MVI M,'0'
2490 DIO MOV A,E
2500  SUB C
2510  MOV E,A
2520  MOV A,D
2530  SBB B
2540  MOV D,A
```

```
2550  JM DI1
2560  INR M
2570  JMP DI0
2580 DI1 MOV A,E
2590  ADD C
2600  MOV E,A
2610  MOV A,D
2620  ADC B
2630  MOV D,A
2640  INX H
2650  RET
2660 *SUBROUTINE TO SWAP D AND E
2670 SWAP MOV A,D
2680  MOV D,E
2690  MOV E,A
2700  RET
2710 *SUBROUTINE TO PRINT BCD NUMBERS
2720 *WITH LEADING ZERO SUPPRESSION
2730 *CHARACTERS STORED IN BUFFER
2740 OUTLZS LXI H,BUFF
2750  MVI B,5
2760  MOV A,M
2770  INX H
2780  CPI '-'
2790  JNZ NCHAR
2800  MOV C,A
2810  CALL CO
2820 NCHAR MOV A,M
2830  INX H
2840  CPI '0'
2850  JNZ NZERO
2860  DCR B
2870  JNZ NCHAR
2880  MOV C,A
2890  CALL CO
2900  RET
2910 NZERO DCX H
2920 NDIG MOV C,M
2930  INX H
2940  CALL CO
2950  DCR B
2960  JNZ NDIG
2970  RET
2980 *SUBROUTINE TO ROUND FIXED POINT INTEGER
2990 ROUND INX H
3000  INX H
3010  MOV A,M
3020  RAL
3030  MVI A,0
3040  INX H
3050  ADC M
```

```
3060  STA X
3070  RET
3080 *SUBROUTINE FOR 4-BYTE ADDITION
3090 *TWO 4-BYTE NUMBERS IN MEMORY VIA H,L ;D,E
3100 ADD4 MVI B,3
3110  ORA A
3120  LDAX D
3130  ADC M
3140  MOV M,A
3150  INX H
3160  INX D
3170  DCR B
3180  JNZ ADD4+3
3190  MVI A,0
3200  ADC M
3210  MOV M,A
3220  RET
3230 *SUBROUTINE TO INPUT BYTE FROM UART
3240 RX ORA A
3250 *SET TIMER TO ZERO
3260  MVI A,0
3270  STA TIMER
3280  LDA TIMER
3290  DCR A
3300  STA TIMER
3310 *INCREMENT TIMER EACH TIME UART STATUS IS READ AND
3320 *NOT READY
3330 *RETURN IF TIMER OVERFLOWS (TIME-OUT)
3340  RZ
3350 *TEST UART STATUS
3360  IN USTAT
3370  MOV B,A
3380  ANI 02H
3390  JZ RX+6
3400  MOV A,B
3410  ANI 20H
3420  JZ $+1
3430 *SET CY IF FRAMING ERROR
3440  STC
3450 *READ UART DATA
3460  IN USDAI
3470  RET
3480 *STORAGE FOR CHARACTER STRINGS
3490 *"DAY OR NIGHT PASS?"
3500 S1 DW 'AD'
3510  DW ' Y'
3520  DW 'RO'
3530  DW 'N '
3540  DW 'GI'
3550  DW 'TH'
3560  DW 'P '
```

```
3570  DW 'SA'
3580  DW '?S'
3590  DB ETX
3600 S2 DW 'RØ'
3610 *"ØRBIT NUMBER="
3620  DW 'IB'
3630  DW ' T'
3640  DW 'UN'
3650  DW 'BM'
3660  DW 'RE'
3670  DB '='
3680  DB ETX
3690 *"START CØLUMN="
3700 S3 DW 'TS'
3710  DW 'RA'
3720  DW ' T'
3730  DW 'ØC'
3740  DW 'UL'
3750  DW 'NM'
3760  DB '='
3770  DB ETX
3780 *"WIDTH="
3790 S4 DW 'IW'
3800  DW 'TD'
3810  DW '=H'
3820  DB ETX
3830 *"LINE # SINCE AØS="
3840 S5 DW 'IL'
3850  DW 'EN'
3860  DW '# '
3870  DW 'S '
3880  DW 'NI'
3890  DW 'EC'
3900  DW 'A '
3910  DW 'SØ'
3920  DB '='
3930  DB ETX
3940 *"LENGTH="
3950 S6 DW 'EL'
3960  DW 'GN'
3970  DW 'HT'
3980  DB '='
3990  DB ETX
4000 *"VISUAL ØR IR? "
4010 S7 DW 'IV'
4020  DW 'US'
4030  DW 'LA'
4040  DW 'Ø '
4050  DW ' R'
4060  DW 'RI'
4070  DW ' ?'
```

```
4080  DB ETX
4090 *"RECØRD LENGTH="
4100 S8 DW 'ER'
4110  DW 'ØC'
4120  DW 'DR'
4130  DW 'L '
4140  DW 'NE'
4150  DW 'TG'
4160  DW '=H'
4170  DB ETX
4180 *"START TAPE G,"
4190 S9 DW 'TS'
4200  DW 'RA'
4210  DW ' T'
4220  DW 'AT'
4230  DW 'EP'
4240  DW 'G,'
4250  DB ETX
4260 *"WRØNG ØRBIT"
4270 S10 DW 'RW'
4280  DW 'NØ'
4290  DW ' G'
4300  DW 'RØ'
4310  DW 'IB'
4320  DB 'T'
4330  DB ETX
4340 *"LINE NØT FØUND"
4350 S11 DW 'IL'
4360  DW 'EN'
4370  DW 'N '
4380  DW 'TØ'
4390  DW 'F '
4400  DW 'UØ'
4410  DW 'DN'
4420  DB ETX
4430 *"READ ERRØR IN RECØRD"
4440 S12 DW 'ER'
4450  DW 'DA'
4460  DW 'E '
4470  DW 'RR'
4480  DW 'RØ'
4490  DW 'I '
4500  DW ' N'
4510  DW 'ER'
4520  DW 'ØC'
4530  DW 'DR'
4540  DB ETX
4550 *"END ØF TAPE"
4560 S14 DW 'NE'
4570  DW ' D'
4580  DW 'FØ'
4590  DW 'T '
4600  DW 'PA'
4610  DB 'E'
4620  DB ETX
4630 *"READ DØNE"
4640 S15 DW 'ER'
4650  DW 'DA'
4660  DW 'D '
4670  DW 'NØ'
4680  DB 'E'
4690  DB ETX
4700 S16 DW 'YB'
4710  DW 'ET'
4720  DB ETX
4730 *CØMMØN VARIABLES
4740 *INTERNAL AND EXTERNAL
4750  ØRG 3B00H
4760 VLMA DS 2
4770 DNFLAG DS 1
4780 BCDBUF DS 5
4790 CØUNT DS 1
4800 ØRNUM DS 2
4810 STCØL DS 2
4820 LINENØ DS 2
4830 LENGTH DS 2
4840 ACCUM DS 2
4850 TEMP DS 2
4860 FACTØR DS 2
4870 WIDTH DS 2
4880 VIFLAG DS 1
4890 RLENG DS 2
4900 BUFF DS 6
4910 RNUM DS 2
4920 RLØNG DS 2
4930 BNUM DS 2
4940 PREC DS 2
4950 LINE DS 1
4960 RSAML DS 4
4970 RSAMP DS 4
4980 CØLUM DS 1
4990 X DS 1
5000 PIXEL DS 1
5010 TIMER DS 1
5020  END
```

APPENDIX 10

PROGRAM TO READ AND WRITE MEMORY TO TAPE

```
0000 *READ AND WRITE IMAGE MEMORY TO TAPE
0001 *SIMILAR TO SCREEN MEMORY TRANSFER TO AND
0002 *FROM TAPE
0003 *ENTIRE MEMORY IS TRANSFERRED IN 150 SEC.
0004 *REQUIRES EXTERNAL SUBROUTINES IN
0005 *SATELLITE DATA ACQUISITION PROGRAM
0010  ORG 3C00H
0020 NUL EQU 0FFH
0030 SYN EQU 16H
0040 MEMCON EQU 80H
0050 USTAT EQU 0F7H
0060 USCMD EQU 0F7H
0070 USDAO EQU 0F6H
0080 READ EQU 3800H
0090 WRITE EQU 3813H
0100 STRING EQU 3857H
0110 SYNC EQU 38E9H
0120 RXEN EQU 3915H
0130 VLMA EQU 3B00H
0140 ETX EQU 03H
0150 RX EQU 39C6H
0160 PSW EQU 06H
0161 SP EQU 6
0170 *WRITE MEMORY TO TAPE
0171 *WRITE CHAN 1 THEN CHAN 0
0172  LXI SP,3E80H
0180  LXI H,0
0190  SHLD VLMA
0200  MVI A,04H
0210  OUT MEMCON
0220  STA FLAG
0221 *INITIALIZE UART
0230  CALL RXEN
0240  MVI A,33H
0250  OUT USCMD
0260  LXI H,1000
0261 *WRITE LEADER
0270 LOOP1: MVI A,NUL
0280  CALL UOUT
0290  DCX H
0300  MOV A,H
0310  ORA L
0320  JNZ LOOP1
0321 *WRITE SYNC BYTE
0330  MVI A,SYN
0340  CALL UOUT
0341 *READ FROM MEMORY
0350 LOOP2: CALL READ
0360  CALL UOUT
0361 *INCREMENT MEMORY ADDRESS
0370  LHLD VLMA
0380  INX H
0390  SHLD VLMA
0400  MOV A,H
0410  ORA L
0420  JNZ LOOP2
0421 *TEST IF CHAN 0 WRITTEN
0430  LDA FLAG
0440  ORA A
0441 *RETURN TO MONITOR WHEN DONE
0450  CZ 0
0451 *SET TO CHAN 0
0460  MVI A,0
0470  OUT MEMCON
0480  STA FLAG
0490  JMP LOOP2
0500 *READ FROM TAPE
0501 *READ CHAN 1 THEN CHAN 0
0502  LXI SP,3E80H
```

```
0510 RD: LXI H,0
0520  SHLD VLMA
0530  MVI A,04H
0540  OUT MEMCON
0550  STA FLAG
0551 *SYNCHRONIZE UART TO LEADER
0560  CALL SYNC
0561 *READ DATA BYTE
0570 RDLOP: CALL RX
0571 *HALT IF FRAMING ERROR
0580  JC ERROR
0581 *WRITE TO MEMORY
0590  CALL WRITE
0591 *INCREMENT ADDRESS
0600  LHLD VLMA
0610  INX H
0620  SHLD VLMA
0630  MOV A,H
0640  ORA L
0650  JNZ RDLOP
0660  LDA FLAG
0670  ORA A
0671 *RETURN TO MONITOR WHEN TAPE READ
0680  CZ 0
0681 *SET TO CHAN 0
0690  MVI A,0
0700  STA FLAG
0710  OUT MEMCON
0720  JMP RDLOP
0730 ERROR: LXI H,CHAR
0731 *PRINT "READ ERROR"
0740  CALL STRING
0741 *RETURN TO MONITOR
0750  CALL 0
0760 CHAR: DW 'ER'
0770  DW 'DA'
0780  DW 'E '
0790  DW 'RR'
0800  DW 'RO'
0810  DB ETX
0811 *SUBROUTINE TO WRITE BYTE TO TAPE
0820 UOUT: PUSH PSW
0830  IN USTAT
0840  ANI 01H
0841 *WAIT FOR UART READY
0850  JZ UOUT+1
0860  POP PSW
0870  OUT USDA0
0880  RET
0890 FLAG: DS 1
0900  END
```

APPENDIX 11

PROGRAMS FOR IMAGE CONTRAST ENHANCEMENT

```
0000 *PROGRAM TO ENHANCE IMAGE CONTRAST WITH LINEAR
0010 *TRANSFER FUNCTION
0020 *GAIN AND SHIFT (CONTRAST AND BRIGHTNESS) ARE SET
0030 *SO THAT MIN AND MAX PIXEL VALUES OF IMAGE
0040 *COVER ENTIRE 4-BIT RANGE OF DISPLAY
0050 *DISPLAY PIXEL VALUE (0-15) =
0060 *(INPUT PIXEL VALUE-SHIFT)*GAIN/256
0070 *VALUES OUTSIDE DISPLAY RANGE ARE LIMITED TO
0080 *MIN AND MAX DISPLAY VALUES
0090 *GAIN AND SHIFT ARE PROGRAMMED BY ENTERING
0100 *VALUES INTO THESE MEMORY LOCATIONS
0110 *INPUT IMAGE IS STORED IN CHAN 1 IN 8 BIT FORM
0120 *HI-ORDER CHAN 0 IS NOT AFFECTED
0130  ORG 3200H
0140 MEMCON EQU 80H
0150 SP EQU 6
0160 TABLE EQU 2C00H
0170  LXI SP,3E80H
0180 *FORM TRANSFER FUNCTION LOOK-UP TABLE
0190 *POINT TO START OF TABLE
0200  LXI H,TABLE
0210  SHLD ADDR
0220 *INITIALIZE INPUT VALUE
0230  MVI A,0
0240  STA TEMP
0250 *POINT TO SHIFT
0260 TGEN: LXI H,SHIFT
0270  SUB M
0280 *CHECK IF RESULT IS -VE, IF SO SET TO 0
0290  JNC $+2
0300  MVI A,0
0310  MOV E,A
0320 *MULTIPLY INPUT-SHIFT BY GAIN
0330  LDA GAIN
0340  MVI D,0
0350  LXI H,0
0360  MVI B,8
0370 *MULTIPLICATION LOOP
0380 LOOP: DAD H
0390  RAL
0400  JNC DEC
0410  DAD D
0420  ACI 0
0430 DEC: DCR B
0440  JNZ LOOP
0450 *DIVIDE BY 256 BY GETTING HI-ORDER PRODUCT
0460  MOV A,H
0470 *TEST IF PRODUCT >15
0480  ANI 0F0H
0490  MOV A,H
0500  JZ $+2
```

```
0510 *IF SO, LIMIT TO 15
0520  MVI A,0FH
0530  ANI 0FH
0540  LHLD ADDR
0550 *STORE VALUE IN TABLE FOR TRANSFER FUNCTION
0560  MOV M,A
0570 *POINT TO NEXT TABLE ELEMENT
0580  INX H
0590  SHLD ADDR
0600  LDA TEMP
0610 *INCREMENT INPUT VALUE OF TRANSFER FUNCTION
0620  INR A
0630  STA TEMP
0640 *256 INPUT VALUES SINCE INPUT IMAGE=8 BIT CODE
0650  JNZ TGEN
0660 *MODIFY CONTRAST VIA LOOK-UP TABLE
0670  LXI H,0
0680 *READ INPUT IMAGE POINT IN CHAN 1
0690 *SET UP IMAGE MEMORY ADDRESS
0700 BACK: MVI A,04H
0710  OUT MEMCON
0720  MOV A,H
0730  OUT 81H
0740  MOV A,L
0750  OUT 82H
0760 *READ IMAGE POINT
0770  CALL READ
0780 *COMPUTE TABLE ADDRESS, DATA IS LO-ADDRESS
0790 *TABLE MUST START AT A PAGE BOUNDARY
0800  LXI H,TABLE
0810  MOV L,A
0820 *GET TRANSFORMED VALUE FROM TABLE
0830  MOV E,M
0840 *WRITE TO DISPLAY, READ FIRST TO GET HI-HALF
0850  MVI A,0
0860  OUT MEMCON
0870  CALL READ
0880  ANI 0F0H
0890  ORA E
0900 *WRITE, WITHOUT DESTROYING HI-HALF
0910  OUT 83H
0920  OUT 83H
0930  INX H
0940 *SCAN THROUGH ENTIRE IMAGE
0950  MOV A,H
0960  ORA L
0970  JNZ BACK
0980 *RETURN TO MONITOR
0990  CALL 0
```

```
0991 *SUBROUTINE TO READ FROM MEMORY
0992 READ: IN 83H
0993  MOV B,A
0994  IN 83H
0995  CMP B
0996  JNZ READ
0997  RET
1000 *STORAGE AREA
1010 GAIN: DS 1   ;CONTRAST FACTOR, 8-BIT UNSIGNED
1020 SHIFT: DS 1  ;REDUCES MIN INPUT VALUE
1030 TEMP: DS 1
1040 ADDR: DS 2
1050  END
```

```
0000 *PROGRAM TO ENHANCE IMAGE CONTRAST BY
0001 *HISTOGRAM FLATTENING
0009  ORG 1E00H
0010 SP EQU 6
0020 MEMCON EQU 80H
0030 DATA EQU 83H
0040 HIADR EQU 81H
0050 LOADR EQU 82H
0060 TABLE EQU 2C00H
0061  LXI SP,3E80H
0065 *CLEAR HISTOGRAM (SET 512 BYTES=0)
0070  LXI H,HISTGM
0080  MVI B,0
0090  MVI A,0
0100 CLEAR: MOV M,A
0110  INX H
0120  MOV M,A
0130  INX H
0140  DCR B
0150  JNZ CLEAR
0160 *GENERATE HISTOGRAM
0162 *HISTOGRAM HAS 256 AMPLITUDE BINS, 16 BITS EACH
0164 *INPUT IMAGE IS IN CHANNEL 1, EACH PIXEL=8 BITS
0170  LXI H,0
0175 *POINT TO CHANNEL 1
0180  MVI A,4
0190  OUT MEMCON
```

```
0195 *READ DATA FRØM CHANNEL 1
0200 HIST: CALL READ
0205 *SAVE MEMØRY ADDRESS
0210  PUSH H
0220  LXI D,HISTGM
0225 *HL=PIXEL VALUE*2
0230  MVI H,0
0240  MØV L,A
0250  DAD H
0255 *CØMPUTE ADDRESS ØF AMPLITUDE BIN
0256 *=HISTGM+PIXEL VALUE*2
0260  DAD D
0265 *DØ 16-BIT INCREMENT ØF (HL),(HL+1)
0270  INR M
0280  JNZ $+2
0290  INX H
0300  INR M
0305 *RESTØRE IMAGE MEMØRY ADDRESS
0310  PØP H
0315 *PØINT TØ NEXT PIXEL
0320  INX H
0325 *TEST IF SCANNED ALL PIXELS (HL=0)
0330  MØV A,H
0340  ØRA L
0350  JNZ HIST
0360 *CØMPUTE CUMULATIVE FUNCTIØN (CDF) BY
0361 *INTEGRATING PDF JUST FØRMED
0370  LXI H,HISTGM
0375 *SET INITIAL SUM TØ 0 (CDF(0)=0)
0380  LXI D,0
0390  MVI A,0
0400 CDF: MØV C,M
0410  INX H
0420  MØV B,M
0430  XCHG
0435 *CDF VALUE=PREVIØUS CDF VALUE+PDF VALUE
0440  DAD B
0450  XCHG
0455 *REPLACE PDF VALUE WITH CDF VALUE
0456 *CDF TABLE REPLACES PDF TABLE
0460  DCX H
0470  MØV M,E
0480  INX H
0490  MØV M,D
0500  INX H
0510  DCR A
0515 *SUM ØVER 256 BINS
0520  JNZ CDF
0530 *PDF(255)=2**16
0531 *2**16 ØVERFLØWS A 16-BIT INTEGER
0532 *2**16 = 0 FØR A 16-BIT NUMBER
```

```
0533 *REMOVE OVERFLOW AND LIMIT MAX TO 2**16-1
0534 *START FROM END OF CDF AND REPLACE 0000 WITH FFFF
0540  LXI H,HISTGM+511
0560 TEST: MOV A,M
0580  ORA A
0590  JNZ DONE
0630  MVI M,OFFH
0640  DCX H
0650  MVI M,OFFH
0660  DCX H
0685  JMP TEST
0690 *COMPUTE EQUALIZATION TABLE
0691 *CDF IS NONLINEAR AMPLITUDE TRANSFER FUNCTION
0692 *AND RESULTING IMAGE HAS APPOXIMATELY
0693 *FLAT HISTOGRAM
0694 *OUTPUT IMAGE HAS 4-BIT PIXELS
0695 *FORM TABLE FOR AMPLITUDE TRANSFORMATION
0696 *FROM CDF VALUES/2**12
0700 DONE: LXI B,TABLE
0710  LXI H,HISTGM+1
0720 LOOP: MOV A,M
0730 *DIVISION BY 2**12 EQUIVALENT TO HI-BYTE/16
0780  ANI OFOH
0790  RRC
0800  RRC
0810  RRC
0820  RRC
0830  STAX B
0840  INR C
0842  INX H
0843  INX H
0850  JNZ LOOP
0860 *ADJUST IMAGE ELEMENT VALUES
0870  LXI D,TABLE
0880  LXI H,0
0890 BACK: MVI A,4
0900  OUT MEMCON
0905 *GET INPUT VALUE
0910  CALL READ
0915 *INPUT VALUE FORMS LO-ADDRESS OF TABLE
0920  MOV E,A
0930  MVI A,0
0940  OUT MEMCON
0945 *READ CHANNEL 0 ELEMENT
0950  CALL ERR
0960  ANI OFOH
0970  MOV B,A
0975 *GET TRANSFORMED VALUE
0980  LDAX D
0990  ORA B
0995 *WRITE TO SCREEN WITHOUT DESTROYING UPPER 4-BITS
```

```
1000  OUT DATA
1010  OUT DATA
1020  INX H
1025 *SCAN THROUGH ALL ELEMENTS
1030  MOV A,H
1040  ORA L
1050  JNZ BACK
1060  CALL 0
1065 *SUBROUTINE TO READ FROM IMAGE MEMORY
1066 *HL=ADDRESS
1070 READ: MOV A,H
1080  OUT HIADR
1090  MOV A,L
1100  OUT LOADR
1110 ERR: IN DATA
1120  MOV B,A
1130  IN DATA
1140  CMP B
1150  JNZ ERR
1160  RET
1165 *STORAGE FOR PDF AND CDF
1170 HISTGM: DS 512
1180  END
```

APPENDIX 12

THRESHOLD AND CONTOURING PROGRAMS

```
0000 *PRØGRAM TØ DRAW CØNTØURS ARØUND CLØUDS FRØM
0010 *SATELLITE IMAGE ØNTØ RADAR IMAGE AND
0020 *SHADE CLØUD AREAS WITH BLACK DØTS
0030 *SATELLITE IMAGE IN CHAN 1 (INVISIBLE) -8 BITS
0040 *RADAR IMAGE IN LØW HALF CHAN 0 (SCREEN) -4 BITS
0050 *UPPER HALF CHAN 0 NØT AFFECTED -SHØULD HØLD RADAR
0060 *IMAGE SØ THAT IT CAN BE CØPIED TØ LØ-HALF CHAN 0
0070 *IF NEW CØNTØURS ARE TØ BE DRAWN
0080 *BEFØRE DRAWING CØNTØURS ØN RADAR, BLACK SHØULD BE
0081 *CHANGED TØ VIØLET SØ THAT BLACK REPRESENTS CLØUD
0082 *AREAS ØNLY AND NØT RADAR DATA ITSELF
0083 *SET CØNTENTS ØF LEVEL TØ CLØUD THRESHØLD
0090  ØRG 2800H
0091 PSW EQU 6
0092 SP EQU 6
0093  LXI SP,3E80H
0110 *SET UP DISPLAY MEMØRY ADDRESS
0120  LXI H,0
0130  MVI A,0
0140  ØUT 80H
0150 *SET INITIAL PREVIØUS PIXEL VALUE=0
0160  MVI E,0
0170 *SET TØ HØRIZØNTAL SCAN MØDE
0180  MVI D,0
0190 *SCAN IMAGE LEFT TØ RIGHT AND TØP TØ BØTTØM
0200 LØØP: MØV A,L
0210  ØUT 82H
0220  MØV A,H
0230  ØUT 81H
0240 *CØMPARE SATELLITE IMAGE PØINT TØ THRESHØLD
0250 *AND DRAW CØNTØUR ARØUND CLØUD BØUNDARY
0260  CALL READ
0270  INR L
0280  JNZ LØØP
0290  INR H
0300  JNZ LØØP
0310 *SET SCANNING TØ VERTICAL MØDE
0320  MVI D,1
0330  JMP ØVER
0340 READ: MVI A,04H
0350 *READ DATA FRØM CHAN 1 (SATELLITE IMAGE)
0360  ØUT 80H
0370  IN 83H
0380  MØV B,A
0390  IN 83H
0400  CMP B
0410  JNZ READ
0420  MØV C,A
0430 *CØMPARE PIXEL VALUE TØ THRESHØLD
0440  CALL CMPR
0450  JC BELØW
0460  MØV A,E
```

```
0470 *GET PREVIØUS ADJACENT PIXEL
0480  CALL CMPR
0490 *IF IT IS BELØW THRESHØLD AND PRESENT
0500 *PIXEL IS ABØVE (=CLØUD), THEN PRESENT
0510 *PØINT IS A CLØUD BØUNDARY
0520  JNC NØT
0530  MØV A,D
0540 *TEST IF VERTICAL ØR HØRIZØNTAL SCAN
0550  ØRA A
0560  JZ HØR
0570 *VERTICAL SCAN
0580  DCR H
0590 *DRAW A CØNTØUR PØINT JUST ABØVE CLØUD EDGE
0600  CALL WRITE
0610  INR H
0620  JMP NØT
0630 *HØRIZØNTAL SCAN
0640 HØR: DCR L
0650 *DRAW CØNTØUR PØINT JUST TØ LEFT ØF CLØUD EDGE
0660  CALL WRITE
0670  INR L
0680 *SET PREVIØUS PIXEL VALUE TØ PRESENT
0690 NØT: MØV E,C
0700  RET
0710 *PRESENT PIXEL IS BELØW CLØUD THRESHØLD
0720 *TEST IF PREVIØUS PIXEL IS ABØVE THRESHØLD
0730 BELØW: MØV A,E
0740  CALL CMPR
0750  JC NØT
0760 *IF IT IS, PREVIØUS PIXEL IS AT A
0770 *CLØUD BØUNDARY
0780 *DRAW A CØNTØUR PØINT AT PRESENT PØSITIØN
0790 *SINCE PRESENT PIXEL IS JUST TØ RIGHT (HØRIZØNTAL SCAN)
0800 *ØR BELØW (VERTICAL SCAN)
0810  CALL WRITE
0820  JMP NØT
0830 *SUBRØUTINE TØ WRITE A BLACK DØT ØN SCREEN
0840 *AT PRESENT PØSITIØN
0850 WRITE: MVI A,0
0860 *WRITE TØ CHANNEL 0 (DISPLAY)
0870  ØUT 80H
0880 *READ DATA PØINT AT PRESENT LØCATIØN (HL)
0890  CALL RD
0900 *SET LØ BYTE-HALF TØ BLACK WITHØUT
0910 *DESTRØYING DATA IN UPPER HALF
0920  ANI 0F0H
0930  ØUT 83H
0940  ØUT 83H
0950  RET
```

```
0960 ØVER: MØV A,L
0970  ØUT 82H
0980  MØV A,H
0990  ØUT 81H
1000 *DRAW CØNTØUR ARØUND CLØUD FRØM VERTICAL SCANNING
1010  CALL READ
1020 *SCAN TØP TØ BØTTØM AND LEFT TØ RIGHT
1030  INR H
1040  JNZ ØVER
1050  INR L
1060  JNZ ØVER
1070 *SHADE IN AREA BØUNDED BY CØNTØUR (CLØUD) ØVER RADAR
1080 SHADE: MVI A,04H
1090  ØUT 80H
1091 *READ SATELLITE DATA
1100  CALL RD
1110  MVI A,0
1120  ØUT 80H
1130  MØV A,B
1140  MØV E,A
1141 *CØMPARE SATELLITE PIXEL TØ THRESHØLD
1150  CALL CMPR
1160  JC NEXT
1161 *SHADE IN CLØUD AREA BY SETTING EVERY ØTHER
1162 *SCREEN PIXEL TØ BLACK INSIDE CØNTØUR
1163 *EVEN CØLUMNS ØN EVEN RØWS AND
1164 *ØDD CØLUMNS ØN ØDD RØWS AND CANDIDATES
1165 *FØR SETTING TØ BLACK
1166 *IE IF LSB ØF LØ AND HI ADDRESS ARE SAME
1170  MØV A,L
1180  XRA H
1181 *TEST IF LSB(L)=LSB(H)
1190  ANI 01H
1200  JNZ NEXT
1201 *TEST IF 4 SURRØUNDING NEIGHBØURS (UP, DØWN,
1202 *LEFT, RIGHT) ARE BLACK
1203 *A BLACK NEIGHBØUR PØINT IS A CØNTØUR PØINT
1210  DCR L
1220  CALL TEST
1230  INR L
1240  JC NEXT
1250  DCR H
1260  CALL TEST
1270  INR H
1280  JC NEXT
1290  INR L
1300  CALL TEST
1310  DCR L
1320  JC NEXT
1330  INR H
```

```
1340  CALL TEST
1350  DCR H
1360  JC NEXT
1361 *IF N0 NEIGHB0URS ARE BLACK, THEN SET
1362 *PRESENT P0INT T0 BLACK
1370  CALL WRITE
1380 NEXT: INX H
1390  M0V A,L
1400  0RA H
1410  JNZ SHADE
1411 *WHEN SCANNING C0MPLETE, RETURN T0 M0NIT0R
1420  CALL 0
1430 *SUBR0UTINE T0 READ P0INT IN DISPLAY MEM0RY
1440 *SET UP ADDRESS
1450 RD: M0V A,L
1460  0UT 82H
1470  M0V A,H
1480  0UT 81H
1490 *READ DATA
1500 WAIT: IN 83H
1510  M0V B,A
1520  IN 83H
1530  CMP B
1540  JNZ WAIT
1550  RET
1551 *SUBR0UTINE T0 TEST IF SCREEN DATA (L0 HALF CHAN 0)
1552 *=0 (BLACK)
1559 *READ DATA FR0M CHAN 0
1560 TEST: CALL RD
1570  ANI 0F0H
1580  M0V C,A
1590  M0V A,B
1591 *MASK F0R SCREEN DATA
1600  ANI 0FH
1610  M0V A,C
1611 *SET CY=1 IF SCREEN DATA IS BLACK
1620  RNZ
1630  STC
1640  RET
1650 *SUBR0UTINE T0 C0MPARE VALUE IN A (8-BIT UNSIGNED)
1660 *WITH LEVEL (VALUE 0F CL0UD THRESH0LD)
1670 CMPR: PUSH H
1680  PUSH D
1690 *D0 16-BIT C0MPARE
1700  M0V L,A
1710  MVI H,0
1711  PUSH PSW
```

```
1719 *GET THRESHOLD LEVEL AND NEGATE AND ADD
1720  LDA LEVEL
1721  CMA
1722  INR A
1723  MOV E,A
1724  MVI D,0
1725  POP PSW
1730  DAD D
1740  POP D
1750  POP H
1760  CMC
1770 *SET CY IF DATA
1780  RET
1781 *MEMORY LOCATION FOR THRESHOLD
1782 LEVEL: DS 1
1790  END
```

```
0000 *PROGRAM TO DRAW CONTOURS AROUND RADAR ECHOS FROM
0010 *RADAR IMAGE ONTO SATELLITE IMAGE AND
0020 *SHADE ECHO AREAS WITH BLACK DOTS
0030 *RADAR IMAGE IS STORED IN UPPER BYTES OF CHAN 0
0040 *SATELLITE IMAGE IS STORED IN SCREEN MEMORY
0050 *WHICH IS CONTRAST ENHANCED FROM RAW DATA IN CHAN 1
0060 *AND CHAN 1
0070 *PROCEDURE IS SIMILAR TO CLOUD CONTOUR PROGRAM
0080 *SATELLITE IMAGE HAS BLACK LEVEL CHANGED TO PURPLE
0090 *SET LOC "LEVEL" TO ECHO THRESHOLD
0100 *HIGH ORDER 4 BITS IS THRESHOLD VALUE
0110  ORG 2900H
0120 PSW EQU 6
0130 SP EQU 6
0140  LXI SP,3E80H
0150  LXI H,0
0160  MVI A,0
0170  OUT 80H
0180  MVI E,0
0190  MVI D,0
```

```
0200 *DRAW CØNTØURS ØN SATELLITE IMAGE SCANNING AND
0210 *THRESHØLDING RADAR IMAGE LEFT TØ RIGHT, TØP TØ BØTTØM
0220 LØØP: MØV A,L
0230  ØUT 82H
0240  MØV A,H
0250  ØUT 81H
0260  CALL READ
0270  INR L
0280  JNZ LØØP
0290  INR H
0300  JNZ LØØP
0310 *SCAN TØP TØ BØTTØM AND LEFT TØ RIGHT
0320  MVI D,1
0330  JMP ØVER
0340 *SUBRØUTINE TØ DETERMINE IF PØINT IS A CØNTØUR PØINT
0350 READ: IN 83H
0360  MØV B,A
0370  IN 83H
0380  CMP B
0390  JNZ READ
0400  ANI OFOH
0410  MØV B,A
0420  CALL CMPR
0430  JC BELØW
0440  MØV A,E
0450  CALL CMPR
0460  JNC NØT
0470  MØV A,D
0480  ØRA A
0490  MØV A,E
0500 *TEST SCANNING MØDE
0510  JZ HØR
0520  DCR H
0530  CALL WRITE
0540  INR H
0550  JMP NØT
0560 HØR: DCR L
0570  CALL WRITE
0580  INR L
0590 NØT: MØV E,B
0600  RET
0610 BELØW: MØV A,E
0620  CALL CMPR
0630  JC NØT
0640  MØV A,B
0650  CALL WRITE
0660  JMP NØT
```

```
0670 *SUBROUTINE TO WRITE TO MEMORY
0680 WRITE: PUSH PSW
0690  MOV A,L
0700  OUT 82H
0710  MOV A,H
0720  OUT 81H
0730  POP PSW
0740  OUT 83H
0750  OUT 83H
0760  RET
0770 OVER: MOV A,L
0780  OUT 82H
0790  MOV A,H
0800  OUT 81H
0810  CALL READ
0820  INR H
0830  JNZ OVER
0840  INR L
0850  JNZ OVER
0860 *SHADE AREA BOUNDED BY CONTOUR
0870 SHADE: CALL RD
0880  ANI OFOH
0890  MOV E,A
0900  CALL CMPR
0910  JC NEXT
0920  MOV A,L
0930  XRA H
0940  ANI OIH
0950  JNZ NEXT
0960  DCR L
0970  CALL TEST
0980  INR L
0990  JC NEXT
1000  DCR H
1010  CALL TEST
1020  INR H
1030  JC NEXT
1040  INR L
1050  CALL TEST
1060  DCR L
1070  JC NEXT
1080  INR H
1090  CALL TEST
1100  DCR H
1110  JC NEXT
1120  MOV A,E
1130  CALL WRITE
1140 NEXT: INX H
1150  MOV A,L
1160  ORA H
1170  JNZ SHADE
1180 *RETURN TO MONITOR
1190  CALL 0
```

```
1200 *SUBROUTINE TO READ FROM MEMORY
1210 RD: MOV A,L
1220  OUT 82H
1230  MOV A,H
1240  OUT 81H
1250 WAIT: IN 83H
1260  MOV B,A
1270  IN 83H
1280  CMP B
1290  JNZ WAIT
1300  RET
1310 *SUBROUTINE TO READ AND TEST IF SCREEN
1320 *PIXEL IS SET TO BLACK
1330 *IF =BLACK, SET CY
1340 TEST: CALL RD
1350  ANI 0F0H
1360  MOV C,A
1370  MOV A,B
1380  ANI 0FH
1390  MOV A,C
1400  RNZ
1410  STC
1420  RET
1430 *SUBROUTINE TO COMPARE A WITH THRESHOLD LEVEL
1440 *A AND LEVEL ARE 8-BIT UNSIGNED INTEGERS
1450 CMPR: PUSH H
1460  PUSH D
1470  MOV L,A
1480  MVI H,0
1490  LDA LEVEL
1500 *NEGATE
1510  CMA
1520  INR A
1530  MOV E,A
1540  MVI D,0FFH
1550  DAD D
1560  POP D
1570  POP H
1580  CMC
1590  RET
1600 LEVEL: DB 40H
1610  END
```

```
0000 *PROGRAM TO THRESHOLD SATELLITE AND RADAR IMAGES AND DISPLAY
0001 *SATELLITE IMAGE STORED IN CHAN 1 AS 8-BITS
0002 *RADAR IMAGE STORED IN UPPER HALF CHAN 0
0003 *RESULT WRITTEN ON 1SCREEN
0009  ORG 2A00H
0010 RS EQU 0EH   ;DISPLAY COLOUR FOR CLOUD AND TARGET
0020 RNS EQU 0DH  ;DISPLAY COLOUR FOR NO CLOUD AND TARGET
0030 NRNS EQU 03H ;DISPLAY COLOUR FOR NO CLOUD AND NO TARGET
0040 NRS EQU 04H  ;DISPLAY COLOUR FOR CLOUD AND NO TARGET
0049 *CLOUD THRESHOLD
0050 CLOUD EQU 8BH
0055 *RADAR TARGET THRESHOLD
0060 TARG EQU 04H
0061 SP EQU 6
0062  LXI SP,3E80H
0065 *INITIALIZE IMAGE MEMORY ADDRESS
0070  LXI H,0
0075 *SET UP ADDRESS
0080 MAIN: MOV A,H
0090  OUT 81H
0100  MOV A,L
0110  OUT 82H
0120  MVI A,04H
0130  OUT 80H
0131 *READ SATELLITE PIXEL
0140  CALL READ
0150  MOV C,A
0160  MVI A,0
0170  OUT 80H
0175 *READ RADAR PIXEL
0180  CALL READ
0185 *MASK OFF LOW HALF
0190  ANI 0F0H
0200  MOV B,A
0205 *MOVE HI TO LO HALF
0210  RRC
0220  RRC
0230  RRC
0240  RRC
0245 *TEST FOR TARGET OR NO TARGET
0250  CPI TARG
0260  JNC RETRN
0270 *NO RADAR RETURN
0280  MOV A,C
0285 *TEST FOR CLOUD
0290  CPI CLOUD
0300  JNC CLD
0310 *NO CLOUD, NO TARGET
0320  MVI A,NRNS
0330  JMP WRITE
```

```
0335 *CLØUD, NØ TARGET
0340 CLD: MVI A,NRS
0350  JMP WRITE
0355 *TARGET
0360 RETRN: MØV A,C
0370  CPI CLØUD
0380  JNC YES
0385 *TARGET, NØ CLØUD
0390  MVI A,RNS
0400  JMP WRITE
0404 *TARGET AND CLØUD
0410 YES: MVI A,RS
0415 *ØR IN HI HALF AND WRITE TØ DISPLAY
0420 WRITE: ANI OFH
0430  ØRA B
0440  ØUT 83H
0450  ØUT 83H
0455 *NEXT PØINT
0460  INX H
0470  MØV A,L
0480  ØRA H
0490  JNZ MAIN
0495 *RETURN TØ MØNITØR
0500  CALL 0
0505 *SUBRØUTINE TØ READ FRØM DISPLAY
0510 READ: IN 83H
0520  MØV B,A
0530  IN 83H
0540  CMP B
0550  JNZ READ
0560  RET
0570  END
```

APPENDIX 13

IMAGE CORRELATION PROGRAM

```
0000 *PRØGRAM FØR CØRRELATIØN
0010 *BETWEEN TWØ IMAGES
0020 *FØR AUTØ-CØRRELATIØN, BØTH IMAGES ARE SAME
0030 *IMAGES ARE STØRED IN CHANNEL 1 (INVISIBLE)
0040 *IMAGES HAVE 4-BIT DATA PIXELS
0050 *IMAGE 1 IS STØRED IN UPPER HALF ØF CHAN 1 BYTE
0060 *IMAGE 2 IS STØRED IN LØWER HALF
0070 *ARRAY SIZE IS 256 X 256 ELEMENTS
0080 *CØRRELATIØN FUNCTIØN IS STØRED IN CHANNEL 0
0090 *AS IT IS CØMPUTED AND HENCE IS VISIBLE ØN DISPLAY
0100 *LØWER-ØRDER 4 BITS ARE VISIBLE
0110 *TØ DISPLAY FINAL RESULT, CØPY CHANNEL 0 TØ CHAN 1
0120 *AND USE CØNTRAST PRØGRAM TØ SCALE RESULTS INTØ
0130 *RANGE 0-15 FØR DISPLAY
0140 *INPUT IMAGES ARE IN UNSIGNED 4-BIT FØRM
0150 *BIAS IS REMØVED BY SUBTRACTING
0160 *PRØDUCT ØF SUMS/N**2
0170 *RXY(0,0) IS CØMPUTED TØ ESTIMATE THE MAX
0180 *VALUE ØF RXY(I,J)  FØR SCALING IT TØ RANGE
0181 *0 - 255 ; SCALE FACTØR IS PØWERS ØF 2 ØNLY
0182 *ENTER RANGE ØF I,J INTØ LØCATIØNS MIN,MAX
0183 *IN 2'S CØMPLEMENT FØRM
0189  ØRG 2D00H
0190 DATA EQU 83H
0200 MEMCØN EQU 80H
0210 HIADR EQU 81H
0220 LØADR EQU 82H
0230 SP EQU 6
0240 TABLE EQU 2C00H
0250  LXI SP,3E80H
0260  MVI A,4
0270  ØUT MEMCØN
0280  LXI D,0
0290  LXI H,SXY
0291 *LØC HL CØNTAINS ADDRESS ØF 3-BYTE SUM (24 BITS)
0300  SHLD HL
0301 *CØMPUTE SUM ØF PØINTS IN IMAGE 1
0310  CALL MEAN
0320  LXI H,TXY
0330  SHLD HL
0331 *CØMPUTE SUM ØF PØINTS IN IMAGE 2
0340  CALL MEAN
0341 *RETURN TØ MØNITØR AND ENTER INTØ BASE
0342 *PRØDUCT ØF CØNTENTS ØF SXY AND TXY/2**16
0343 *BASE, SXY ,TXY ARE 3-BYTE VALUES
0344 *WITH LØ-ØRDER BYTE FIRST
0350  CALL 0
0351 *FØRM MULTIPLICATIØN LØØKUP TABLE
0352 *UPPER AND LØWER HALVES ØF LØ ADDRESS
0353 *ARE MULTIPLIER AND MULTIPLICAND
0354 *PRØDUCT IS 8-BITS UNSIGNED
0355 *TABLE STARTS ØN PAGE BØUNDARY SØ THAT
0356 *LØ ADDRESS BYTE IS DATA
```

```
0360  LXI H,TABLE
0361 *SET MULTIPLIER TO 0
0370  MVI C,0
0371 *CLEAR INITIAL PRODUCT
0380 NEXT: MVI A,0
0381 *SET LOOP COUNTER
0390  MVI B,15
0400  MOV M,A
0410  INX H
0420 MULT: ADD C       ;FORM 4X4
0430  MOV M,A       ;TABLE
0440  INX H
0450  DCR B
0460  JNZ MULT
0470  INR C
0480  MOV A,C
0490  CPI 16
0500  JNZ NEXT
0510 *FIND RXY(0,0)
0511 *DE IS USED FOR ADDRESS OF LOOKUP TABLE
0520  LXI D,TABLE
0530  MVI A,0
0540  STA J
0550  STA I
0551 *COMPUTE RXY(0,0) ;I,J=0
0560  CALL SUB
0561 *RETURN TO MONITOR
0562 *MINIMUM FACTR= LOG2(RXY(0,0)/256)
0563 *SHOULD BE LARGER IN CASE RXY(0,0) NOT MAX
0570  CALL 0
0580 *IMAGE CORRELATION
0581 *INITIALIZE INDICIES
0590  LDA MIN     ;J=MIN
0600  STA J
0601 *COMPUTE RXY(I,J) ;I,J OVER RANGE <MIN,MAX>
0610 DOJ: LDA MIN     ;I=MIN
0620  STA I
0630 DOI: CALL SUB
0640  JMP OVER
0650 SUB: LXI H,0       ;RXY=0
0660  SHLD RXY
0670  SHLD RXY+1
0680  MVI A,0      ;Y=0
0690  STA Y
0700 DOY: LDA Y   ;LL=Y-J
0710  LXI H,J
0720  SUB M
0730  STA LL
0740  MVI A,0      ;X=0
0750  STA X
0760 DOX: LDA X   ;K=X-I
0770  LXI H,I
0780  SUB M
0790  OUT LOADR   ;POINT TO S(K,LL)
```

```
0800  LDA LL
0810  OUT HIADR
0820  MVI A,4
0830  OUT MEMCON
0835 *READ S(K,LL)
0840 RD: IN DATA
0850  MOV B,A
0860  IN DATA
0870  CMP B
0880  JNZ RD
0885 *UPPER HALF OF BYTE
0890  ANI 0F0H
0900  MOV C,A
0910  LDA X       ;POINT TO T(X,Y)
0920  OUT HIADR
0930  LDA X
0940  OUT LOADR
0945 *READ T(X,Y)
0950 AGAIN: IN DATA
0960  MOV B,A
0970  IN DATA
0980  CMP B
0990  JNZ AGAIN
0995 *LOW HALF
1000  ANI 0FH
1001 *COMBINE HALES TOGETHER TO FORM
1002 *LOOK-UP TABLE ADDRESS
1010  ORA C
1020  MOV E,A
1030  LDAX D    ;S(K,LL)*T(X,Y)
1040  LXI H,RXY    ;RXY=RXY+S(K,LL)*T(X,Y)
1050  ADD M
1060  MOV M,A
1070  JNC DOWN
1071 *ADD WITH CARRY TO HIGHER BYTES
1080  INX H
1090  INR M
1100  JNZ DOWN
1110  INX H
1120  INR M
1130 DOWN: LXI H,X    ;NEXT X
1140  INR M
1150  JNZ DOX
1160  LXI H,Y       ;NEXT Y
1170  INR M
1180  JNZ DOY
1190  RET
1200 OVER: LDA J
1201 *SET UP DISPLAY ADDRESS FROM I,J
1202 *LET CENTRE OF DISPLAY (8080H) BE RXY(0,0)
1203 *CONVERT SIGNED I,J TO UNSIGNED ADDRESS
1210  ADI 128
1220  OUT HIADR
```

```
1230  LDA I
1240  ADI 128
1250  0UT L0ADR
1260  MVI A,0
1270  0UT MEMC0N
1271 *REM0VE BIAS FR0M RXY(I,J)
1280  CALL N0RML
1290  LDA FACTR
1291 *SCALE RXY T0 8-BIT RANGE
1300  0RA A
1301 *IF FACTR=0 D0 N0T SHIFT (2**0=1)
1310  JZ X1
1320  M0V B,A
1321 *SHIFT RXY (24-BIT) LEFT FACTR TIMES
1330 L00P: LXI H,RXY+2
1340  MVI C,3
1350  0RA A
1360 SHIFT: M0V A,M
1370  RAR
1380  M0V M,A
1390  DCR C
1400  DCX H
1410  JNZ SHIFT
1420  DCR B
1430  JNZ L00P
1440 X1: LDA RXY
1449 *R0UND NUMBER
1450  JNC $+1
1460  INR A
1461 *TEST F0R 0VERFL0W AND LIMIT RXY
1462  STA RXY
1463  LHLD RXY+1  ;GET HIGH 0RDER RXY
1464  M0V A,L
1465  0RA H
1466  LDA RXY
1467  JZ $+2
1468  MVI A,0FFH
1469 *RXY+1 ,RXY+2 SH0ULD BE 0 IF N0 0VERFL0W
1470  0UT DATA
1471 *WRITE RESULT T0 CHAN 0
1480  0UT DATA
1490  LXI H,I
1500  INR M      ;NEXT I
1510  LDA MAX
1520  CMP M
1530  JNZ D0I
1540  LXI H,J     ;NEXT J
1550  INR M
1560  LDA MAX
1570  CMP M
1580  JNZ D0J
1590  CALL 0
1591 *COMPUTE SUM 0F IMAGE ELEMENTS
1592 *ST0RE SUM IN MEMORY INDIRECTLY ADDRESSED
```

```
1593 *BY LOCATION HL
1600 MEAN: LHLD HL    ;CLEAR SUM
1610  MVI A,0
1620  MOV M,A
1630  INX H
1640  MOV M,A
1650  INX H
1660  MOV M,A
1661 *SET UP IMAGE MEMORY ADDRESS
1670 SIGMA: MOV A,D
1680  OUT HIADR
1690  MOV A,E
1700  OUT LOADR
1701 *READ MEMORY
1710 READ: IN DATA
1720  MOV B,A
1730  IN DATA
1740  CMP B
1750  JNZ READ
1760  LHLD HL
1761 *REG H,L POINTS TO SUM
1762 *MASK FOR LO-HALF
1770  ANI OFH
1780  ADD M
1790  MOV M,A
1800  JNC UNDER
1801 *ADD WITH CARRY TO HIGH ORDER BYTES
1810  INX H
1820  INR M
1830  JNZ UNDER
1840  INX H
1850  INR M
1855 *SWAP BYTES HALVES
1860 UNDER: MOV A,B
1870  RLC
1880  RLC
1890  RLC
1900  RLC
1901 *STORE BACK IN MEMORY
1902 *SINCE THIS SUBROUTINE IS CALLED TWICE
1903 *ORDER OF IMAGES IS RESTORED
1904 *BY SWAPPING TWICE
1910  OUT DATA
1920  OUT DATA
1930  INX D
1940  MOV A,D
1950  ORA E
1955 *SCAN ENTIRE IMAGE
1960  JNZ SIGMA
1970  RET
1971 *SUBROUTINE TO SUBTRACT BIAS
1972 *POINT TO RXY (24-BIT)
1980 NORML: LXI H,RXY
1985 *SUBTRACT BIAS BY ADDING COMPLEMENT OF BIAS
```

```
1986 *3-BYTE PRECISION
1990  LXI B,BASE
2000  LDAX B
2010  ADD M
2020  MOV M,A
2030  INX H
2040  INX B
2050  LDAX B
2060  ADC M
2070  MOV M,A
2080  INX H
2090  INX B
2100  LDAX B
2110  ADC M
2120  MOV M,A
2130  RP
2135 *TAKE ABS
2136 *3-BYTE COMPLEMENT
2140  MOV A,M
2150  CMA
2160  MOV M,A
2170  DCX H
2180  MOV A,M
2190  CMA
2200  MOV M,A
2210  DCX H
2220  MOV A,M
2230  CMA
2235 *3-BYTE INCREMENT
2240  INR A
2250  RNZ
2260  INX H
2270  INR M
2280  RNZ
2290  INX H
2300  INR M
2310  RET
2320 *VARIABLE STORAGE
2330 RXY: DS 3    ;ACCUMULATOR FOR SUM OF PRODUCTS
2335 *J,I,Y,X INDEXING LOCATIONS
2340 J: DS 1
2350 I: DS 1
2360 Y: DS 1
2370 X: DS 1
2380 LL: DS 1
2390 HL: DS 2
2400 SXY: DS 3
2410 TXY: DS 3
2420 BASE: DS 3   ;BIAS FACTOR (-VE NUMBER)
2430 FACTR: DS 1  ;EACH 24-BIT RXY IS DIVIDED BY 2**FACTR
2435 *2'S COMPLEMENT NUMBERS
2440 MAX: DS 1
2450 MIN: DS 1
2460  END
```

```
0000 *IMAGE R0TATI0N,MAGNIFICATI0N, AND TRANSLATI0N
0010 *INPUT IMAGE IN UPPER HALF 0F BYTE, CHAN 0
0020 *TRANSF0RMED IMAGE IN L0WER HALF (SCREEN)
0030  0RG 3000H
0040 L0ADR EQU 82H
0050 HIADR EQU 81H
0060 MEMC0N EQU 80H
0070 MRD EQU 83H
0080 MWR EQU 83H
0090 *SET T0 CHAN 0
0100  MVI A,0
0110  0UT MEMC0N
0120  LDA DX    ;-DX*64
0130  CALL M64
0140  SHLD DXP
0150  LDA DY
0160  CALL M64
0170  SHLD DYP
0180 *Y3 AND X3 IS IMAGE PIXEL C0-0RDINATES
0190 *0,0 IS CENTRE 0F IMAGE
0200 *START AT UPPER LEFT C0RNER (-128,-128)
0210 *GIVEN P0INT Y3,X3 IN TRANSF0RMED IMAGE,
0220 *C0MPUTE ADDRESS Y0,X0 0F C0RRESP0NDING P0INT
0230 *IN INPUT IMAGE, T0 NEAREST INTEGER VALUE (NEIGHB0UR)
0240  MVI A,-128
0250  STA Y3
0260 0UTER: LDA Y3
0270  M0V E,A   ;Y3S=Y3*SINDT
0280  LHLD SINDT
0290  M0V B,H
0300  M0V C,L
0310  CALL MULT
0320  SHLD Y3S
0330  M0V A,D
0340  STA Y3S+2
0350  LDA Y3    ;Y3C=Y3*C0SDT
0360  M0V E,A
0370  LHLD C0SDT
0380  M0V B,H
0390  M0V C,L
0400  CALL MULT
0410  SHLD Y3C
0420  M0V A,D
0430  STA Y3C+2
0440  MVI A,-128
0450  STA X3
0460 INNER: LDA X3
0470  M0V E,A   ;X3*C0SDT
0480  LHLD C0SDT
0490  M0V B,H
0500  M0V C,L
```

```
0510  CALL MULT
0520  LXI B,Y3S   ;Y3S+X3*CØSDT
0530  CALL ADD3
0540  CALL D256
0550 * KXP*(Y3S+X3*CØSDT)
0560  MØV B,H
0570  MØV C,L
0580  LDA KXP
0590  MØV E,A
0600  MVI D,0
0610 *SINCE KXP IS UNSIGNED, BYPASS FIRST PART
0620 *ØF MULTIPLICATIØN SUBRØUTINE
0630  CALL PLUS
0640 *NØRMALIZE PRØDUCT TØ 16 BITS
0650  CALL D256
0660 * DXP + KXP*(Y3S+X3*CØSDT)
0670  XCHG
0680  LHLD DXP
0690  DAD D
0700 *NØRMALIZE X ADDRESS TØ 8 BIT
0710  CALL HL64
0720 *CØNVERT SIGNED TØ UNSIGNED (ADD 128)
0730  LXI D,128
0740  DAD D
0750  MØV A,H
0760  ØRA A
0770 *ADDRESS ØUT ØF RANGE IF H NØT 0
0780  JNZ RANGE
0790  MØV A,L
0800 *SET X ADDRESS
0810  ØUT LØADR
0820 *CØMPUTE Y ADDRESS
0830 * X3*SINDT
0840  LDA X3
0850  MØV E,A
0860  LHLD SINDT
0870  MØV B,H
0880  MØV C,L
0890  CALL MULT
0900  LXI B,Y3C
0910 * -X3*SINDT
0920  CALL NGAT
0930 * Y3C - X3*SINDT
0940  CALL ADD3
0950 *NØRMALIZE RESULT
0960  CALL D256
0970  MØV B,H
0980  MØV C,L
0990 * KYP*(Y3C - X3*SINDT)
1000  LDA KYP
1010  MØV E,A
```

```
1020 *KYP IS +VE NUMBER
1030  MVI D,0
1040  CALL PLUS
1050 *NØRMALIZE RESULT
1060  CALL D256
1070  XCHG
1080 * DYP + KYP*(Y3C - X3*SINDT)
1090  LHLD DYP
1100  DAD D
1110  CALL HL64
1120 *NØRMALIZE TØ 8 BIT
1130 *CØNVERT TØ UNSIGNED
1140  LXI D,128
1150  DAD D
1160  MØV A,H
1170  ØRA A
1180 *TEST IF Y ADDRESS IS ØUT ØF RANGE
1190  JNZ RANGE
1200  MØV A,L
1210 *SET Y ADDRESS
1220  ØUT HIADR
1230 *READ INPUT IMAGE PØINT
1240  CALL READ
1250  ANI OFOH
1260 *MØVE INTØ LØ HALF
1270  RRC
1280  RRC
1290  RRC
1300  RRC
1310  MØV C,A
1320 *SET ADDRESS TØ Y3,X3 (ØUTPUT PØINT)
1330 UP: LDA X3
1340  ADI 128
1350  ØUT LØADR
1360  LDA Y3
1370  ADI 128
1380  ØUT HIADR
1390 *WRITE PØINT TØ SCREEN WITHØUT DESTRØYING UPPER HALF
1400  CALL READ
1410  ANI OFOH
1420  ØRA C
1430  CALL WRITE
1440 *INCREMENT Y3,X3
1450  LDA X3
1460  INR A
1470  STA X3
1480  CPI 80H
1490  JNZ INNER
1500  LDA Y3
1510  INR A
1520  STA Y3
```

```
1530  CPI 80H
1540  JNZ OUTER
1550 *RETURN TO MONITOR
1560  CALL 0
1570 RANGE: MVI C,0
1580  JMP UP
1590 *SUBROUTINE FOR SIGNED MULTIPLY H,L=-64*A
1600 *CLEAR CY
1610 M64: ORA A
1620 *ARITHMETIC SHIFT RIGHT A
1630  JP $+1
1640  STC
1650  RAR
1660 *MOVE RESULT (9 BITS) INTO H,L
1670  MOV H,A
1680  MVI A,0
1690  RAR
1700  MOV L,A
1710 *TEST IF H,L POSITIVE NUMBER
1720  MOV A,H
1730  ORA A
1740 *DIVIDE H,L BY -2
1750 *ARITHMETIC SHIFT RIGHT AND TAKE TWO'S COMPLEMENT
1760  JP $+1
1770  STC
1780  RAR
1790  CMA
1800  MOV H,A
1810  MOV A,L
1820  RAR
1830  CMA
1840  MOV L,A
1850  INX H
1860  RET
1870 *SUBROUTINE FOR 16 BY 8 BIT SIGNED MULTIPLICATION
1880 *DHL=BC*E
1890 *TAKE ABS OF E ,STORE SIGN IN D (D=0 IF E +VE)
1900 MULT: MOV A,E
1910  MVI D,0
1920  ORA A
1930  JP PLUS
1940  INR D
1950  MOV A,E
1960  CMA
1970  INR A
1980  MOV E,A
1990 *COMPUTE SIGN OF PRODUCT
2000 PLUS: MOV A,B
2010  ORA A
2020  MVI A,0
2030  JP POS
```

```
2040  INR A
2050 *SIGN ØF RESULT =EXCLUSIVE ØR ØF MULTIPLIER AND
2060 *MULTIPLICAND SIGNS
2070  XRA D
2080  MØV D,A
2090 *TAKE ABS ØF B,C
2100  MØV A,B
2110  CMA
2120  MØV B,A
2130  MØV A,C
2140  CMA
2150  MØV C,A
2160  INX B
2170 PØS: MØV A,D
2180 *SAVE SIGN
2190  STA SIGN
2200  MVI A,8
2210  STA CØUNT
2220 *INITIALIZE PARTIAL PRØDUCT
2230  MVI D,0
2240  LXI H,0
2250 *MULTIPLICATIØN LØØP
2260 *SHIFT D,H,L REGISTERS LEFT
2270 MLØØP: DAD H
2280  MØV A,D
2290  RAL
2300  MØV D,A
2310 *SHIFT MULTIPLIER LEFT TØ TEST BIT IN CY
2320 ØVER1: MØV A,E
2330  RAL
2340  MØV E,A
2350  JNC BZERØ
2360 *ADD MULTIPLICAND TØ PARTIAL PRØDUCT
2370 *PARTIAL PRØDUCT HAS 3 BYTES
2380  DAD B
2390  JNC $+1
2400 *INCREMENT D IF CY FRØM PREVIØUS 2 BYTES
2410  INR D
2420 *LØØP 8 TIMES
2430 BZERØ: LDA CØUNT
2440  DCR A
2450  STA CØUNT
2460  JNZ MLØØP
2470 *RESTØRE SIGN
2480  LDA SIGN
2490  ØRA A
2500  RZ
2510 *IF RESULT -VE, NEGATE D,H,L (ABS ØF SIGNED RESULT)
2520 NGAT: MØV A,L
2530  CMA
2540  ADI 1
2550  MØV L,A
```

```
2560  MØV A,H
2570  CMA
2580  ACI 0
2590  MØV H,A
2600  MØV A,D
2610  CMA
2620  ACI 0
2630  MØV D,A
2640  RET
2650 *SUBRØUTINE TØ ADD D,H,L TØ MEMØRY (B,C)
2660 *SUM IN D,H,L
2670 *D IS HI-ØRDER BYTE
2680 *ADD HLD + (BC)
2690 ADD3: LDAX B
2700  ADD L
2710  MØV L,A
2720  INX B
2730  LDAX B
2740  ADC H
2750  MØV H,A
2760  INX B
2770  LDAX B
2780  ADC D
2790  MØV D,A
2800  RET
2810 *SUBRØUTINE FØR HL=DHL/256 ,SIGNED DIVISIØN
2820 D256: MØV A,L
2830  ØRA A
2840  MØV L,H
2850  MØV H,D
2860 *RØUND RESULT UP IF MSB ØF L =1
2870  RP
2880  INX H
2890  RET
2900 *SUBRØUTINE FØR SIGNED DIVISIØN BY 128
2910 *HL=DHL/128
2920 *ARITHMETIC SHIFT DHL 7 TIMES RIGHT
2930 D128: MVI B,7
2940  MØV A,D
2950  ØRA A
2960  JP $+1
2970  STC
2980  RAR
2990  MØV D,A
3000  MØV A,H
3010  RAR
3020  MØV H,A
3030  MØV A,L
3040  RAR
3050  MØV L,A
3060  DCR B
```

```
3070  JNZ D128+2
3080 *ROUND UP RESULT IF CY SET
3090  RNC
3100  INX H
3110  RET
3120 *SUBROUTINE FOR SIGNED DIVISION OF HL BY 64
3130 HL64: MVI B,6
3140 *ARITHMETIC SHIFT RIGHT HL 6 TIMES
3150  MOV A,H
3160  ORA A
3170  JP $+1
3180  STC
3190  RAR
3200  MOV H,A
3210  MOV A,L
3220  RAR
3230  MOV L,A
3240  DCR B
3250  JNZ HL64+2
3260  RNC
3270 *ROUND HL UP IF CY SET (LAST BIT SHIFTED OUT)
3280  INX H
3290  RET
3300 *SUBROUTINE TO READ IMAGE MEMORY
3310 *ADDRESS GIVEN BY H,L
3320 READ: IN MRD
3330  MOV B,A
3340  IN MRD
3350  CMP B
3360  JNZ READ
3370  RET
3380 *SUBROUTINE TO WRITE TO MEMORY
3390 WRITE: OUT MWR
3400  OUT MWR
3410  RET
3420 *VARIABLES AND CONSTANTS
3430 SIGN: DS 1
3440 COUNT: DS 1
3450 *COS AND SIN OF ROTATION ANGLE
3460 *COSDT =32768*COS(DT)  ,DT IN RADIANS
3470 *COSDT,SINDT ARE SIGNED INTEGERS
3480 COSDT: DS 2
3490 SINDT: DS 2
3500 *X AND Y SHIFT FACTORS
3510 DX: DS 1
3520 DY: DS 1
3530 *X AND Y SCALE FACTORS
3540 KXP: DS 1
3550 KYP: DS 1
```

```
3560 *TEMPØRARY STØRAGE
3570 Y3S: DS 3
3580 Y3C: DS 3
3590 X3: DS 1
3600 Y3: DS 1
3610 DYP: DS 2
3620 DXP: DS 2
3630  END
```

```
10 T=3*4096+8*256
20 PI=3.141593
30 A=2*PI/2048
40 M=3*512
50 K=255
60 FØR I=0 TØ M
70 X=K*SIN(A*I)
80 X=ABS(X)
90 F=X-INT(X)
100 X=INT(X)
110 IF F>=.5 THEN X=X+1
120 PØKE T+I,X
130 NEXT I
140 END
```

APPENDIX 14

SCAN CONVERSION PROGRAM

```
0000 *PROGRAM FOR RADAR SCAN CONVERSION
0010 *READS RECTANGULAR DATA FROM TAPE
0020 *AND WRITES TO DISPLAY IN POLAR FORM
0030 *VIDEO RECORDER DATA IS DIGITIZED AND STORED IN MEMORY
0040 *DATA IS RECORDED ON TAPE
0050 *DATA IS READ FROM TAPE VIA THIS PROGRAM
0060 *PROGRAM READS EACH RADAR SAMPLE, COMPUTES THE
0070 *NEARST IMAGE POINT (SCREEN MEMORY ADDRESS) IN
0080 *PPI FORMAT AND WRITES TO SCREEN
0090 *PROGRAM USES LOOK-UP TABLE FOR SIN(X), COS(X)
0100 *LOOK-UP TABLE IS 1536 BYTES LONG
0110 *TABLE DATA GENERATED BY A BASIC LANGUAGE PROGRAM
0120 NUL EQU OFFH
0130 TABLE EQU 3800H
0140 SYN EQU 16H
0150 MEMCON EQU 80H
0160 MRD EQU 83H
0170 MWR EQU 83H
0180 LOADR EQU 82H
0190 HIADR EQU 81H
0200 USTAT EQU OF7H
0210 USCMD EQU OF7H
0220 USDAI EQU OF6H
0230 SP EQU 6
0240  MVI A,0
0250  OUT MEMCON
0260  LXI SP,3E80H
0270 *CLEAR SCREEN
0280  CALL CLEAR
0290 *RETURN TO MONITOR
0300  CALL 0
0310 *START TAPE AT BEGINNING OF DATA, G
0320 *INITIALIZE UART TO 9600 BAUD
0330 SYNC: MVI A,0
0340  OUT USCMD
0350  OUT USCMD
0360  OUT USCMD
0370  MVI A,40H
0380  OUT USCMD
0390  MVI A,4EH
0400  OUT USCMD
0410  MVI A,16H
0420  OUT USCMD
0430 *SYNCHRONIZE TO TAPE
0440 *SEARCH FOR 240 NULS IN LEADER
0450  MVI C,240
0460 WAIT: IN USTAT
0470  MOV B,A
0480  ANI 02H
0490  JZ WAIT
0500  MOV A,B
```

```
0510  ANI 20H
0520  JNZ SYNC
0530  IN USDAI
0540  CPI NUL
0550  JNZ SYNC
0560  DCR C
0570  JNZ WAIT
0580 *WAIT FØR SYNC BYTE
0590 SYNWT: IN USTAT
0600  MØV B,A
0610  ANI 02H
0620  JZ SYNWT
0630  MØV A,B
0640  ANI 20H
0650  JNZ SYNC
0660  IN USDAI
0670  CPI NUL
0680  JZ SYNWT
0690  CPI SYN
0700  JNZ SYNC
0710  JMP START
0720 *SUBRØUTINE TØ WRITE DATA TØ SCREEN WITHØUT
0730 *DESTRØYING UPPER HALF ØF BYTES IN CHAN 0
0740 SCRN: ANI 0FH
0750  MØV D,A
0760 *SET UP ADDRESS
0770  MØV A,H
0780  ØUT HIADR
0790  MØV A,L
0800  ØUT LØADR
0810 *READ MEMØRY
0820 READ: IN MRD
0830  MØV B,A
0840  IN MRD
0850  CMP B
0860  JNZ READ
0870  ANI 0F0H
0880 *CØMBINE UPPER HALF WITH DATA
0890  ØRA D
0900 *WRITE TØ MEMØRY
0910  ØUT MWR
0920  ØUT MWR
0930  RET
0940 *SUBRØUTINE TØ READ UART
0950 UIN: IN USTAT
0960  MØV B,A
0970  ANI 02H
0980  JZ UIN
0990  MØV A,B
1000  ANI 20H
1010  JNZ STØP
1020  IN USDAI
1030  RET
```

```
1040 *HALT IF FRAMING ERRØR
1050 STØP: HLT
1060 *SUBRØUTINE TØ CLEAR SCREEN
1070 CLEAR: LXI H,0
1080  MVI A,0
1090  CALL SCRN
1100  INX H
1110  MØV A,L
1120  ØRA H
1130  JNZ CLEAR+3
1140  RET
1150 *INITIALIZE SECTØR (2048 SECTØRS ARE READ)
1160 START: LXI H,8080H
1170  SHLD XCØRD
1180  LXI H,0
1190  SHLD SECT
1200 *FØR EACH SECTØR, CØMPUTE X AND Y INCREMENTS FØR
1210 *UNIT RADIUS
1220 *INCX (X INCREMENT) =SIN(A)
1230 * X=R*SIN(A)*128
1240 * X(N+1)=X(N)+INCX
1250 * A=2*PI*SECT/2048
1260 *SIMILARLY FØR Y INNCREMENT
1270 MAIN: LHLD SECT
1280 *CØMPUTE TABLE ADDRESS FØR SIN
1290 *VALUES IN TABLE =ABS(255*SIN(A)) ,UNSIGNED INTEGERS
1300  MØV A,H
1310 *LIMIT ADDRESS BETWEEN TABLE AND TABLE+1024 (1/2 CYCLE)
1320  ANI 03H
1330  MØV H,A
1340  SHLD TEMP
1350  LXI D,TABLE
1360  DAD D
1370  MØV L,M
1380  MVI H,0
1390  LDA SECT+1
1400 *DETERMINE WHICH HALF ØF CYCLE SECTØR IS IN
1410 *IF SECT>1023, CHANGE TØ NEGATIVE #
1420  ANI 04H
1430  JZ $+3
1440  CALL HLCMP
1450  SHLD INCX
1460 *CØMPUTE Y INCREMENT (INCY)
1470 *INCY = -CØS(A)
1480 *GET LIMITED-RANGE SECTØR #
1490  LHLD TEMP
1500 *CØS(A) = SIN(A+PI/2)
1510  LXI D,TABLE+512
1520  DAD D
1530  MØV L,M
1540  MVI H,0
1550  LDA SECT+1
```

```
1560 *COMPUTE TABLE ADDRESS, DETERMINE SIGN
1570  MOV B,A
1580  RAL
1590  XRA B
1600  ANI 04H
1610  JNZ $+3
1620  CALL HLCMP
1630  SHLD INCY
1640 *INITIALIZE CO-ORDINATES TO CENTRE OF SCREEN (R=0)
1650 *XPOS, YPOS ARE CO-ORDINATES, INTEGER AND FRACTIONAL PARTS
1660  LXI H,32768
1670  SHLD XPOS
1680  SHLD YPOS
1690  MVI A,0
1700  STA SWEEP
1710 *UNPACK BYTE INTO 2 RADAR SAMPLES
1720 INNER: LDA SWEEP
1730  ANI 01H
1740  JNZ OVER
1750 *SCAN SAMPLE COUNT (SWEEP) EVEN, READ DATA BYTE FROM TAPE
1760  CALL UIN
1770  STA BYTE
1780 *GET HI (1ST) HALF
1790  RRC
1800  RRC
1810  RRC
1820  RRC
1830  JMP $+3
1840 OVER: LDA BYTE
1850 *WRITE SAMPLE POINT TO SCREEN
1860  LHLD XCORD
1870  CALL SCRN
1880  LHLD INCY
1890  XCHG
1900  LHLD YPOS
1910 *COMPUTE NEXT Y CO-ORDINATE BY ADDING Y INCREMENT
1920 *TO PREVIOUS Y CO-ORDINATE
1930  CALL SUB
1940  SHLD YPOS
1950 *STORE INTEGER PART IN YCORD
1960  STA YCORD
1970 *COMPUTE NEXT X CO-ORDINATE IN SIMILAR MANNER
1980  LHLD INCX
1990  XCHG
2000  LHLD XPOS
```

```
2010  CALL SUB
2020  SHLD XPOS
2030  STA XCORD
2040  LDA SWEEP
2050  INR A
2060  STA SWEEP
2070 *READ 128 SAMPLES PER SCAN
2080  CPI 128
2090  JNZ INNER
2100 *INCREMENT SECTOR NUMBER
2110  LHLD SECT
2120  INX H
2130  SHLD SECT
2140 *TEST IF SECTOR >2047
2150  MOV A,H
2160  CPI 08H
2170  JNZ MAIN
2180 *ALL DATA READ, RETURN TO MONITOR
2190  CALL 0
2200 SUB: DAD D
2210  MOV A,L
2220  ORA A
2230 *GET INTEGER PART AND ROUND TO NEAREST VALUE
2240  MOV A,H
2250  RP
2260  INR A
2270  RET
2280 *SUBROUTINE TO TAKE TWO'S COMPLEMENT OF H,L
2290 HLCMP: MOV A,H
2300  CMA
2310  MOV H,A
2320  MOV A,L
2330  CMA
2340  MOV L,A
2350  INX H
2360  RET
2370 *VARIABLE STORAGE AREA
2380 TEMP: DS 2
2390 INCX: DS 2
2400 INCY: DS 2
2410 SWEEP: DS 1
2420 XCORD: DS 1
2430 YCORD: DS 1
2440 BYTE: DS 1
2450 SECT: DS 2
2460 XPOS: DS 2
2470 YPOS: DS 2
2480  END
```

DIX 15

PROGRAM FOR DIGITIZING PHOTOGRAPHS

```
0000 *PROGRAM TO DIGITIZE PHOTOGRAPHS FROM OPTO-
0001 *MECHANICAL SCANNER INTO 256 BY 256 ARRAY
0002 *4 BITS PER ELEMENT
0003 *PHOTOTUBE VIDEO IS AMPLIFIED AND LOW PASS FILTERED
0004 *FOR A/D CONVERTER
0005 *4 MSB'S OF A/D ARE CONNECTED TO 4 LSB'S OF CHAN 1
0006 *DATA INPUT TO COMPUTER
0009  ORG 2000H
0010 PORT EQU 83H
0011 SP EQU 6
0015 *PHASING CONTACTS OF SCANNER TO B6
0020 SYNC EQU 40H
0030 DATA EQU 0FH
0040 MEMCON EQU 80H
0050 HIADR EQU 81H
0060 LOADR EQU 82H
0070 MWR EQU 83H
0071 MRD EQU 83H
0075 *DELAY FACTOR AFTER HORIZONTAL SCAN START
0080 SHIFT EQU 3500
0090 LINES EQU 15
0095 *SAMPLING RATE TIMING CONSTANT
0100 RATE EQU A5
0101 *INITIALIZE ADDRESS
0110  LXI H,0
0111  LXI SP,3E80H
0115 *TO READ A/D DATA AND SYNC BIT, MODE =CHAN 1
0116 *AND LO ADDRESS MUST BE EVEN
0120  MVI A,04H
0130  OUT MEMCON
0140  OUT LOADR
0145 *WAIT UNTIL A SAMPLE IS ABOVE THRESHOLD
0146 *IE WHEN BLACK MARKER STRIP IS ENCOUNTERED
0150 FEED: IN PORT
0160  CMA
0170  ANI DATA
0180  CPI 8
0190  JC FEED
0200  CALL SUB
0201 *WAIT FOR SYNC PULSE
0210 SYN: IN PORT
0220  ANI SYNC
0230  JZ SYN
0235 *WAIT FOR VERTICAL MARKER STRIP
0240 EDGE: IN PORT
0250  CMA
0260  ANI DATA
0270  CPI 8
0280  JP EDGE
0285 *DELAY AFTER HORIZONTAL EDGE
0290  LXI D,SHIFT
```

```
0300 LOOP: DCX D
0310  MOV A,D
0320  ORA E
0330  JNZ LOOP
0335 *INPUT DATA FROM A/D
0340 INNER: IN PORT
0350  CMA
0360  ANI DATA
0430  MOV B,A
0435 *SET UP MEMORY ADDRESS FOR CHAN 0
0440  MOV A,H
0450  OUT HIADR
0460  MOV A,L
0470  OUT LOADR
0480  MVI A,0
0490  OUT MEMCON
0491 *WRITE TO MEMORY WITHOUT DESTROYING UPPER BYTE HALF
0492 READ: IN MRD
0493  MOV C,A
0494  IN MRD
0495  CMP C
0496  JNZ READ
0497  ANI 0F0H
0500  ORA B
0510  OUT MWR
0520  OUT MWR
0530  MVI A,04H
0540  OUT MEMCON
0550  OUT LOADR
0555 *DELAY BETWEEN SAMPLES
0560  MVI C,RATE
0570 BACK: DCR C
0580  JNZ BACK
0510 *INPUT 256 SAMPLES PER LINE
0620  JNZ INNER
0630  INR H
0640 *SCAN 256 LINES
0650  JNZ SYN
0655 *RETURN TO MONITOR
0660  CALL 0
0570 SUB: MVI B,LINES
0675 *WAIT FOR DRUM SYNC PULSE FROM PHASING CONTACT
0680 PULSE: IN PORT
0590  ANI SYNC
0700  JZ PULSE
0710  LXI D,1000H
0711 *DELAY LOOP
0720 WAIT: DCX D
0730  MOV A,D
0740  ORA E
0750  JNZ WAIT
0755 *LOOP FOR # DRUM REVOLUTIONS =LINES
0760  DCR B
0770  JNZ PULSE
0780  RET
0810  END
```

REFERENCES

1. Digital Picture Processing , A. Rogenfeld and A.C. Kok
1976 Academic Press, Inc.
TA 1630.R67
Chapter 6 Enhancement, Chapter 8 Segementation

2. Digital Image Processing , W.K. Pratt
1978 John Wiley and Sons, Inc.
TA 1632.P7

4. Manual of Remote Sensing , Vol. II. Robert G. Reeves
Chapter 21
American Society of Photogrammetry, 1975
TR 692.5.M35

5. "A First Step Towards Rainfall Estimation from Satellite Cloud Photographs", W.L. Woodley and B. Sancha
Weather, 26, 1971
pp. 279-289

6. "A Satellite Study of Cloud Clusters over the Tropical North Atlantic Ocean", D.W. Mortin and V.E. Suami
American Meterological Society Bulletin, Feb. 1972
pp. 135-156

7. "Digital Image Processing of Earth Observation Sensor Data"
R. Bernstein
IBM Journal of Research and Development
pp. 40-57, Jan. 1976

8. "Electro-Optical Processing of DAPP Meteorlogical Satellite Data"
James R. Blankenship
American Meteorological Society Bulletin, Jan. 1974
pp. 9-15

9. "Correlation of Cloud Brightness and Radiance with Precipitation Intensity", Roy H. Blockmer, Jr., Oct. 75
Standford Research Institute, NEPRF TR 8-75 (SRI)

10. "Rainfall Estimation from Geosynchronous Satellite Imagery During Daylight hours",
C.G. Griffith,
Space Science and Engineering Centre, University of Wisconsin-Madison

11. Orbital Dynamics of Space Vehicles, R. Deutsch
1963 Prentice-Hall, Inc., Englewood Cliffs, N.J.

12. Fundamentals of Celestial Mechanics, J.M.A. Danby
1964 The Macmillan Company, New York