CORE
Metadata, citation and similar papers at core.ac.uk
Provided by NASA Technical Reports Server

NASA
TP
1870
c.1

TECH LIBRARY KAFB, NM
0067836
LOAN COPY:
AFWL TECH

NASA Technical Paper 1870

# A Design Approach to Real-Time Formatting of High Speed Multispectral Image Data

Barry D. Meredith and W. Lane Kelly IV

MAY 1981

NASA

TECH LIBRARY KAFB, NM

0067836

NASA Technical Paper 1870

# A Design Approach to Real-Time Formatting of High Speed Multispectral Image Data

Barry D. Meredith and W. Lane Kelly IV
*Langley Research Center*
*Hampton, Virginia*

NASA
National Aeronautics
and Space Administration

**Scientific and Technical**
**Information Branch**

1981

## SUMMARY

A design approach to formatting multispectral image data in real time at very high data rates is presented for future onboard processing applications. The approach employs a microprocessor-based alternating buffer-memory configuration whose formatting function is completely programmable. Data are read from an output buffer in the desired format by applying the proper sequence of addresses to the buffer via a lookup table memory. Sensor data can be processed using this approach at rates limited by the buffer-memory access time and the buffer-switching-process delay time. This design offers flexible high speed data processing and will benefit from continuing increases in the performance of digital memories.

## INTRODUCTION

NASA space applications and science missions currently generate vast quantities of multispectral image data. These large volumes of data place a great burden on the ground processing facilities, making the task of information extraction both expensive and time consuming. As a result, techniques for performing image data processing onboard the spacecraft are under investigation as a more efficient approach to converting raw image data into useful information. Data formatting is a key processing function that would be included in most onboard systems. The need for data formatting arises from the sensor specific organization of the data produced by multispectral sensors. This organization is usually determined by the sensor configuration, which results from signal-to-noise and bandwidth trade-offs. Formatting reorganizes the sensor output data into the format which provides the most efficient data processing. Additional formatting functions may be required within the onboard data system to preprocess data for processing functions which require specific organization of the data. Unfortunately, the high data rates exhibited by multispectral image data prohibit the use of existing computer software techniques for data formatting. As a result, high speed special-purpose hardware must be developed to perform formatting in real time.

The purpose of this paper is to describe a design approach for real-time data formatting which can be implemented in future onboard processing systems. The approach employs a high speed alternating buffer-memory configuration and both input and output mode controllers. The controllers provide the proper sequence of address and read/write signals to the buffer memories and effect the mode changes. Since it is desirable to develop a programmable formatting system that accommodates a variety of image-data formats, the overall process is placed under microprocessor control. The microprocessor initializes the individual controllers to perform a specific formatting task and monitors the system operation during the data-processing mode.

## DATA FORMATTING

The format of data produced by a multispectral imaging system is usually the result of design trade-offs for the sensor and the processing requirements downstream of the sensor are not taken into consideration. The task of data formatting is to reorganize this sensor data into a form appropriate for efficient data processing. The particular organization of the data depends upon the specific processing function to be performed on the data. Additional formatting may be required at the interface between processing elements if the subsequent function requires a data format that differes from that produced by the previous stage.

Figure 1 illustrates three representative image-data formats: band-interleaved-by-pixel (BIP), band-interleaved-by-line (BIL), and band sequential (BSQ). Each element of the three columns represents a byte of data related to the spectral intensity of a pixel within a specific line of an image frame. The data source for the example is a multispectral scanner with four spectral bands such as the type employed in Landsat missions. Data are generated by the scanner in the BIP format and are formatted into either BIL or BSQ, depending upon the processing requirements of the subsequent functional element. For example, the data are often radiometrically corrected in the initial stage of processing to compensate for the nonuniformities in the photosensor outputs. Since this calibration is performed on a line-by-line basis, a BIL format is desirable. If data classification is to be performed downstream from the calibration process, each pixel from all bands must be spatially registered, which involves returning the image data to the BIP format. When the final product of the processing effort is a film recording, the data must be arranged by spectral bands. This is accomplished by formatting the data into the BSQ organization (ref. 1).

The overall formatting requirements for an onboard image-data-processing system are determined by the sensor configuration, the types of processing functions, and the arrangement of the functions within the system. Therefore, it is desirable to develop a flexible data formatter that is modular in nature and can accommodate a variety of formatting tasks.

## DESIGN CONSIDERATIONS

High speed multispectral image data provides the greatest challenge for an onboard processing system. Currently, imagers generate data at approximately 30 megabits per second (MBPS); with advances in sensor technology, this rate is expected to increase to 100 MBPS by 1985. Such high rates prohibit the use of conventional computer software approaches to data formatting without significant increases in the speed of the central processing unit. Therefore, special-purpose hardware must be dedicated to the task of performing real-time data formatting.

The special-purpose hardware described in this paper consists of a high speed alternating buffer-memory configuration and its dual mode controllers. This configuration takes advantage of the availability of high speed random access memories (RAM) to accommodate the throughput rate necessary for real-time image processing. In order to process a continuous stream of data, the approach employs two memory systems in a configuration that allows the memories to alternate their roles as input buffer and output buffer.

An onboard image-processing system may require a variety of formatting functions. To develop a format-specific design approach for each function would greatly increase the design complexity of the system. Therefore, it is desirable to develop a modular, flexible formatting element which can be configured to perform any formatting task. In the alternating buffer-memory approach, such flexibility is obtained by placing the special-purpose hardware under microprocessor control. The microprocessor generates input parameters for the special-purpose hardware according to a selected formatting algorithm. These parameters determine the organization of the output data for a specific input-data format. Since the architecture of the formatter must be expandable to meet the memory storage requirements of each formatting task, the microprocessor selected for use in the formatter must be capable of addressing the maximum memory range.

Two important considerations for any onboard processing system are system size and power consumption. In the microprocessor-based, alternating buffer-memory technique for data formatting, these considerations depend primarily upon the amount of RAM employed in the approach. The quantity of memory storage required to perform a formatting task is determined both by the sensor configuration and by the task itself. Sensor design specifications such as the number of spectral channels, spatial resolution, and sampling rate affect the formatter memory size. Additional factors include the specific format being generated and whether the formatting occurs on a channel-by-channel basis as a part of a parallel processing architecture. Ultimately, the imaging sensor and processing system architecture must be defined before the formatter memory specifications can be determined. While imaging applications can require significant memory storage, continuing advancements in memory technology in the areas of reduced power consumption and increased packing density tend to make the alternating buffer-memory approach more feasible. System specifications for a hypothetical formatting application are presented in the appendix.

## DATA FORMATTER DESIGN AND OPERATION

### Overall System Description

The objective of the data formatter design is to produce a flexible formatting element capable of processing a continuous stream of high speed multispectral image data in real time. Figure 2 is a block diagram of a microprocessor-based, alternating buffer-memory system which was designed to meet that objective. The bidirectional bus switches connect the two static high speed RAM's in a ping-pong or alternating buffer configuration. In this configuration, the memories are attached to the appropriate bus and controller in order to load input data into one buffer while resequenced output data is read from the other. Upon completion of each input/output (I/O) cycle the bus switches reverse the roles of the memories as either an input or an output buffer. The control word generated to perform this bus switching is produced by the input controller (S1 to S4, C1, and C2). The proper sequence of addresses and read/

write signals are provided to the memories by the controllers. These signals propagate through the address bus switches in the 3 to 1 direction to RAM A and in the 3 to 2 direction to RAM B (fig. 2). Input data are written into memory after passing through the data bus switches in the 1 to 3 direction. Data are read from either memory in the 3 to 2 direction to the output bus. A digital latch is included in the system to hold valid data at the system output for the duration of a data clock cycle. While data are being formatted by the system, the microprocessor monitors the controller status signals and the number of the data block being processed via its sytem I/O. The status signals can be used to determine whether the controllers are active. Also, one output line of the I/O is designated as a mode control to configure the system for initialization or for processing.

Figure 3 describes the sequence of operation of the data formatter. During the initialization mode the microprocessor generates parameters according to a selected format algorithm and loads these parameters into an address lookup table memory located within the output controller. The organization of these parameters in the lookup table determines the output data format (ref. 2). The microprocessor also establishes the length of the data blocks to be processed and the number of blocks to be processed. The microprocessor then outputs a mode command that configures the special-purpose hardware in the data-processing mode. The processor is free at this time to perform other software tasks. When the input controller receives the first data clock, it generates an address, a write command, and a switch control word to load the input data into memory A. The address generator is incremented and data are loaded into RAM A in a continuous cycle until the buffer is full. After the last byte of the data block is loaded, the input controller sends a control word to the switches configuring RAM A as an output buffer and RAM B as the new input buffer. At this time, the output controller is released from its wait state and utilizes the data clock pulses and the lookup table to generate the desired sequence of addresses to RAM A. The reformatted data are read from RAM A while new data are loaded into RAM B, and this ping-pong process continues until the last desired data block is transferred. Upon completion of the final transfer, the microprocessor receives an interrupt and outputs a mode command returning the system to the initialization mode.

## Input Controller

The input controller is that portion of the special-purpose hardware responsible for supplying addresses and write commands to the memory, which is serving as the input buffer. It also provides the control word for the bus switches to reconfigure the buffers and terminates the processing mode when the final data block has been formatted. Figure 4 is a schematic of the input controller design. Memory addresses are generated by a counter, which is incremented by the input data clock. This counter has a modulus of N, where N is the maximum number of addresses for the formatting application having the largest data block. The programmable input/output circuit allows the microprocessor to initialize magnitude comparator A with the address of the final data byte of the data blocks. When an address greater than that value is encountered at the A input of the comparator, the address counter is cleared to begin the next input cycle. The microprocessor programs magnitude comparator B with the number of data blocks to be processed. When a value greater than this programmed value is detected, the comparator interrupts the processor, returning the system to the initialization mode. The initialization process is performed when the mode command from the microprocessor system I/O is high. This mode signal is used to initialize the controller flip-flops and counters. (See fig. 4.) At the end of the initialization process, the mode line is cleared and the input controller awaits a data clock pulse to begin loading data into memory.

The input process begins when the rising edge of the first data clock pulse is detected by the input controller. The system timing fo this process is illustrated in figure 5. The D flip-flop (① of fig. 4) uses the rising clock pulse to set the input status signal high. This low-to-high transition of the status pulse is received by a toggle flip-flop (② of fig. 4), which controls the system bus switches. The switches configure RAM A as the input buffer and RAM B as an output buffer for this first input cycle. Another D flip-flop (③ of fig. 4) holds the address counter in the cleared state until the negative clock transition is encountered. This insures that the first data byte of the process will be loaded into memory address location zero. (See fig. 5.) One-shot circuit A uses the rising edge of each data clock pulse to generate the write functions for the input buffer. The width of these functions ($t_w$ in fig. 5) must be greater than the memory access time $t_a$ by an amount equal to the time required to switch the input buffer to the controller outputs $t_{sw}$. This requirement is necessary to prevent the loss of the first data byte at the beginning of each input cycle. It also establishes the maximum data rate for the system as

$$f_{max} = \frac{1}{t_w} = \frac{1}{t_a + t_{sw}}$$

where $t_w$ is the width of the write command, $t_a$ is the access time of memory employed in the system, $t_{sw}$ is the switching time of bus switches, and $f_{max}$ is the maximum data clock pulse rate in hertz. The data are loaded into the specified memory location when the write command makes the transition from a low level to a high level. Memory select functions for RAM A and RAM B are generated by address decoders which reside on the individual memory circuit boards.

When the address associated with the last byte of the data block (address M of fig. 5) appears at the A input of comparator A, the comparator issues a command (A = B) to clear the input status signal (fig. 5). One-shot circuit B narrows this command pulse width to prevent the D flip-flop from missing the subsequent data clock. This data clock resets the status, signifying that the input buffer is full. The T flip-flop (fig. 4) then toggles to the opposite state causing the bus switches to reconfigure RAM A as an output buffer and RAM B for input. RAM B is sequentially loaded with input data while data are output from RAM A. This ping-pong process continues until the last data block is formatted.

## Output Controller

The actual task of formatting the input image data is performed by the output controller (fig. 6). The key element to the formatting technique employed by the system is the lookup table (LUT) memory. The output buffer addresses are stored in the appropriate sequence based upon the desired output-data format. A counter, incremented by the data clock, selects the LUT locations from which the addresses are read and placed on the address lines of the output RAM (fig. 6).

During initialization, the mode command causes the LUT address bus switches to attach the LUT address and control lines to the microprocessor system. Also, the bus transceiver is enabled, connecting the system data bus to the lookup table data lines. The processor software uses the selected format algorithm to generate buffer address parameters and then loads these parameters into the LUT memory. The mode signal is used to initialize the hardware components of the controller. Magnitude comparator C receives the address of the final byte of the data blocks from the input controller programmable I/O. This comparator clears the address counter after the final address of the block is encountered. Upon completion of the initialization process, the controller remains in a wait mode until the first block of data is loaded into the input buffer.

When the mode command goes low, the LUT address bus switches connect the counter output to the LUT address lines. During the wait mode, a D flip-flop (④ of fig. 6) holds this address counter clear, and flip-flop ⑤ holds the system output latch (fig. 2) and the output data clock buffer in their high impedance state. The output buffer address residing in location zero of the LUT memory is latched to the LUT output to provide the first address for RAM A when the output cycle begins.

The timing diagram of figure 7 illustrates the reformatted output process. This process begins when RAM A is switched to the output controller and output bus. Switching occurs as a result of a low-to-high transition of the input status signal (transition B of fig. 7), which signifies that the input buffer is loaded. Prior to this event the high-to-low transition of the input status (A) causes flip-flop ④ to remove the clear command from the address counter. The counter output is then incremented to a one count by the following clock pulse whose rising edge (C) is coincident with the memory switch transition (B). This clock pulse simultaneously latches the buffer address in LUT location zero to the RAM A address lines. The high-to-low transition of that same clock in conjunction with a one on the counter's least significant bit causes flip-flop ⑤ in figure 6 to release the system output latch and data clock buffer from their tri-state condition (fig. 7). The rising edge of the output data clock pulse is used to enable the output latch and notify the processing element downstream from the formatter when new data are available on the system output bus. When the next rising input clock pulse (E) appears at the output controller, both the data from RAM A and the address from LUT location one are valid at their respective outputs. This low-to-high clock transition latches the data to the system output and latches the LUT output to the RAM A address lines while it increments the address counter to a count of two. The subsequent positive transition of the clock (F) increments the counter to a count of three, latches the buffer address contained in LUT location two, and latches the output data from the buffer address which resides in LUT location one. This pipeline technique for buffer address generation, i.e., counter to LUT to buffer, is continuous throughout the formatting process.

Magnitude comparator D and flip-flop ⑥ in figure 6 are used to generate an output status signal. This signal is monitored by the input controller (fig. 4) to terminate the formatting process when the final block is output. The comparator places a one on the flip-flop input whenever the counter output is not equal to zero. This one is clocked through to the flip-flop output on the positive transition of the clock pulse. Clock transition E initially sets the status high to denote the presence of data at the system output. The status remains high until another situation with the address equal to zero is encountered. At that time clock transition G clears the status signal. The next low-to-high transition (H) returns the output status signal to a high level, which coincides with the completion of that output cycle (fig. 7).

While the last byte of RAM A is held to the system output, the first address for the RAM B output cycle is latched from the LUT output to the RAM B address lines (fig. 7). The first data byte from RAM B is latched to the output when transition H of the clock pulse appears at the data latch enable.

## CONCLUDING REMARKS

An approach to onboard data formatting has been presented. The necessity for data formatting results from the sensor-specific organization of data produced by multispectral imagers. The formatter reorganizes the image data into a format suitable to the subsequent processing function for efficient processing. The high data rates exhibited by image data prohibit the use of conventional computer software approaches to data formatting. Therefore, an approach has been developed which employs a high speed alternating buffer-memory configuration to accommodate the high processing rates. In addition, this dedicated special-purpose hardware includes both an input and an output controller which generate the proper sequence of addresses and commands to the input and output buffers. The output controller contains a lookup table memory which generates the desired sequence of output buffer addresses by mapping each output of a counter to a specific buffer address. Each controller is initialized by a microprocessor to establish the desired output format and the length and quantity of the data blocks to be processed.

A hypothetical formatting application was applied to the alternating buffer-memory approach, and specifications for memory speed, system power, and size were presented. The application involved formatting image data at a rate of 100 megabits per second (MBPS), with each block consisting of 65 536 bytes. Eight-bit sensor data were assumed for this process; however, the approach could be applied to other data word lengths by reconfiguring the memory organization. The selected data rate could be accommodated by using commercially available memory devices with 55-nanosecond access times. Total system power dissipation was computed and found to be approximately 78 watts for these particular memory devices. This system could be packaged in a 0.0114 cubic meter volume.

The data formatter design approach offers several advantages for future onboard applications. The programmability of the microprocessor-based formatter provides the system flexibility to accommodate a variety of formatting tasks. Also, a continuous stream of high speed data can be processed in real time with no data loss. The maximum data rate of this data stream is limited by the access time of the memories employed in the formatter plus the time required to switch the buffers to their appropriate controllers. Implementation of the formatter approach will become more cost effective and more practical in terms of power and size as the development of high speed memory devices continues as a major technology thrust.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
April 30, 1981

# APPENDIX

## SYSTEM SPECIFICATIONS FOR A HYPOTHETICAL FORMATTING APPLICATION

The following data formatting task is presented to illustrate the system power and size requirements and the performance of the formatting approach described in this paper. The multispectral image data to be processed is in the form of a continuous 8-bit parallel stream. The data will be formatted in real time at a data rate of 12.5 megabytes per second or 100 megabits per second. This high rate is consistent with that of projected imaging systems. Each block of data is composed of 65 536 data bytes. Since the actual output format to be produced is programmable, it is irrelevant to the hardware specifications considered in this exercise.

The formatter employs commercially available bus switches whose switching time $t_{sw}$ is specified as 25 nanoseconds. Having determined this parameter, the following equation can be used to compute the required access time $t_a$ of the memories employed in the two RAM buffers and the lookup table:

$$f_{max} = \frac{1}{t_a + t_{sw}}$$

where

$$f_{max} = 12.5 \text{ mHz}$$

and

$$t_{sw} = 25 \text{ nanseconds}$$

Computation reveals that the data rate proposed for this application can be accommodated with memories whose access time is 55 nanoseconds. Such devices are available in NMOS technology arranged in a 4K × 1 bit organization.* The two data buffers must be 64K × 8 bits each to accommodate the 64K byte data blocks. The lookup table memory size is 64K × 16 bits since 16 output lines are required to address the full range of buffer address locations. Therefore, 512 4K × 1 bit memory circuits are necessary to implement the formatter memories. The total 5-volt power dissipation of the 512 memory devices is typically 63 watts. An additional 15 watts is allotted for the central processing unit (CPU), computer memory, I/O circuits, and all other circuitry. This brings the total 5-volt-system power requirement to 78 watts. All the formatter circuitry can reside on 10 circuit boards whose dimensions are 24.13 cm by 15.24 cm.

*$1K = 2^{10} = 1024$.

## APPENDIX

The system can be enclosed in a 27.94 cm by 17.78 cm by 22.86 cm package, which translates to a volume of 0.0114 cubic meter.

## ABBREVIATIONS AND SYMBOLS

| | |
|---|---|
| ABS | address bus switch |
| DBS | data bus switch |
| LSB | least significant bit |
| LUT | lookup table |
| R/W | read/write |
| T.S. | tri-state |
| $t_w$ | width of the memory write command pulse |

## REFERENCES

1. Ewalt, D.: On-Board Multispectral Classification Study. NASA CR-3134, 1979.

2. Kelly, W. Lane, IV; Meredith, Barry D.; and Howle, William M.: High Speed Lookup Table Approach to Radiometric Calibration of Multispectral Image Data. NASA TP-1741, 1980.

Data source - multispectral scanner
Spectral bands (B) - 4
Pixels (P) per line - n
Lines (L) per frame - x

BIP

$L_1 B_1 P_1$
$L_1 B_2 P_1$
$L_1 B_3 P_1$
$L_1 B_4 P_1$
$L_1 B_1 P_2$
$L_1 B_2 P_2$
$L_1 B_3 P_2$
$L_1 B_4 P_2$
●
●
$L_1 B_1 P_n$
●
$L_1 B_4 P_n$
$L_2 B_1 P_1$
●
$L_2 B_4 P_1$
●
●
$L_2 B_1 P_n$
●
$L_2 B_4 P_n$
●
●
$L_x B_1 P_1$
●
● ●
$L_x B_4 P_n$

BIL

$L_1 B_1 P_1$
$L_1 B_1 P_2$
$L_1 B_1 P_3$
●
●
$L_1 B_1 P_n$
$L_1 B_2 P_1$
$L_1 B_2 P_2$
●
●
$L_1 B_2 P_n$
$L_1 B_3 P_1$
●
●
$L_1 B_3 P_n$
$L_1 B_4 P_1$
●
●
$L_1 B_4 P_n$
$L_2 B_1 P_1$
●
●
$L_2 B_4 P_n$
●
●
$L_x B_1 P_1$
●
●
$L_x B_4 P_n$

BSQ

$L_1 B_1 P_1$
$L_1 B_1 P_2$
$L_1 B_1 P_3$
●
●
$L_1 B_1 P_n$
$L_2 B_1 P_1$
●
●
$L_2 B_1 P_n$
●
●
$L_x B_1 P_1$
●
● ●
$L_x B_1 P_n$
$L_1 B_2 P_1$
●
●
$L_x B_2 P_n$
$L_1 B_3 P_1$
●
●
$L_x B_3 P_n$
●
●
$L_1 B_4 P_1$
●
●
$L_x B_4 P_n$

Figure 1.- Typical image-data formats.

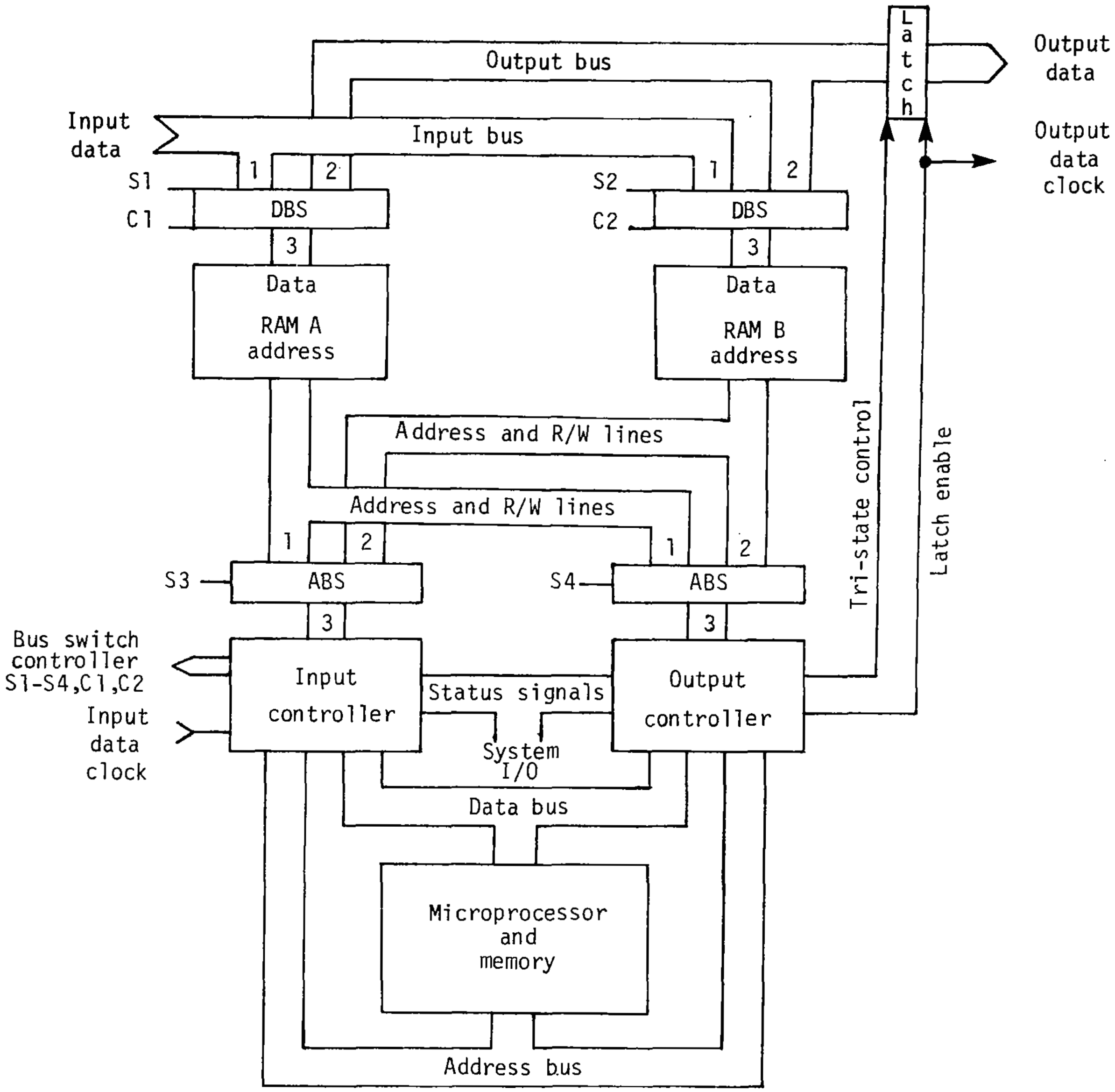


Figure 2.- Block diagram of alternating buffer-memory system for data formatting.

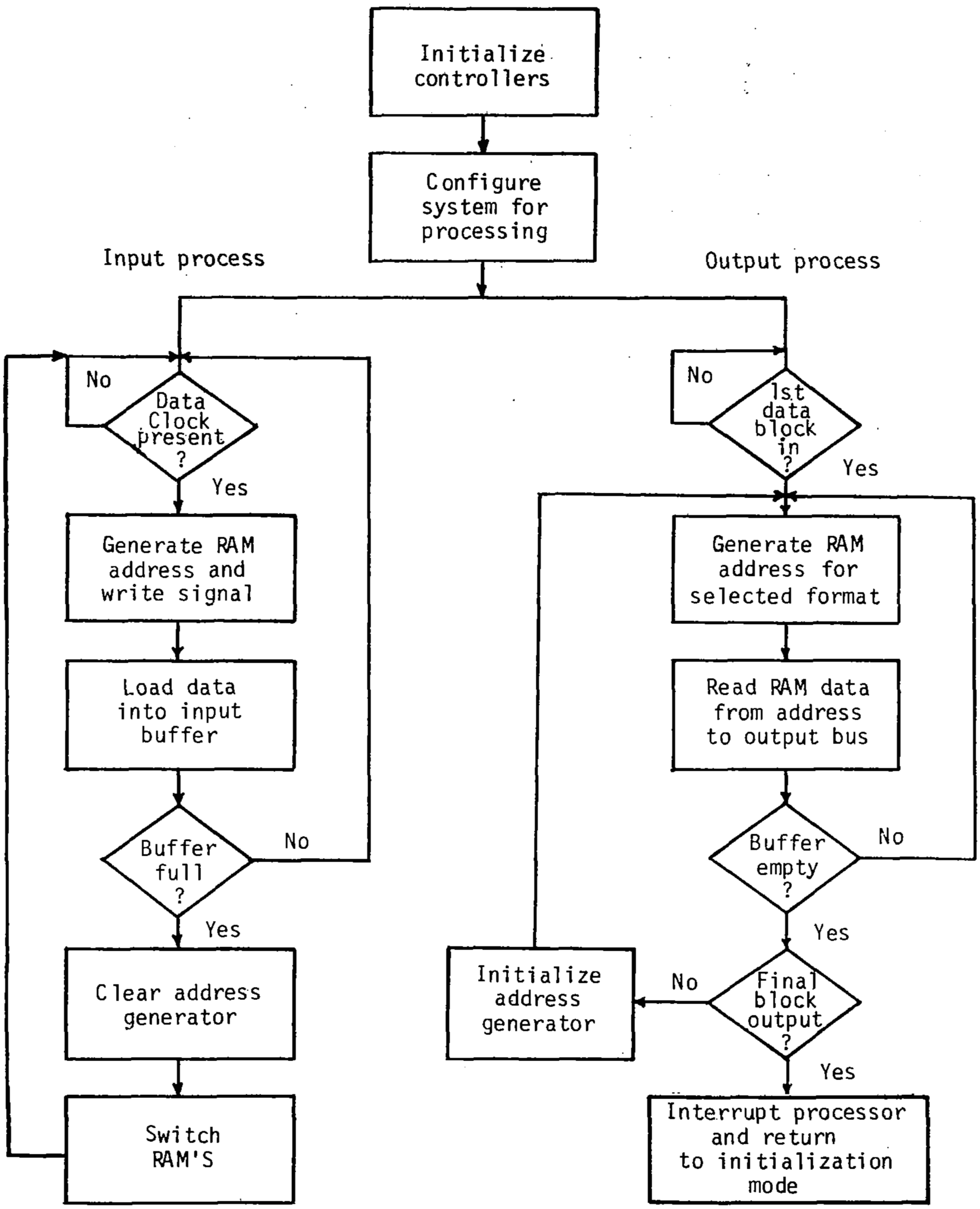

Initialize controllers
Configure system for processing
Input process
Output process
No
Data Clock present ?
Yes
Generate RAM address and write signal
Load data into input buffer
Buffer full ?
No
Yes
Clear address generator
Switch RAM'S
No
1st data block in ?
Yes
Generate RAM address for selected format
Read RAM data from address to output bus
Buffer empty ?
No
Yes
Final block output ?
No
Initialize address generator
Yes
Interrupt processor and return to initialization mode

**Figure 3.- Flow chart of data formatter operation.**

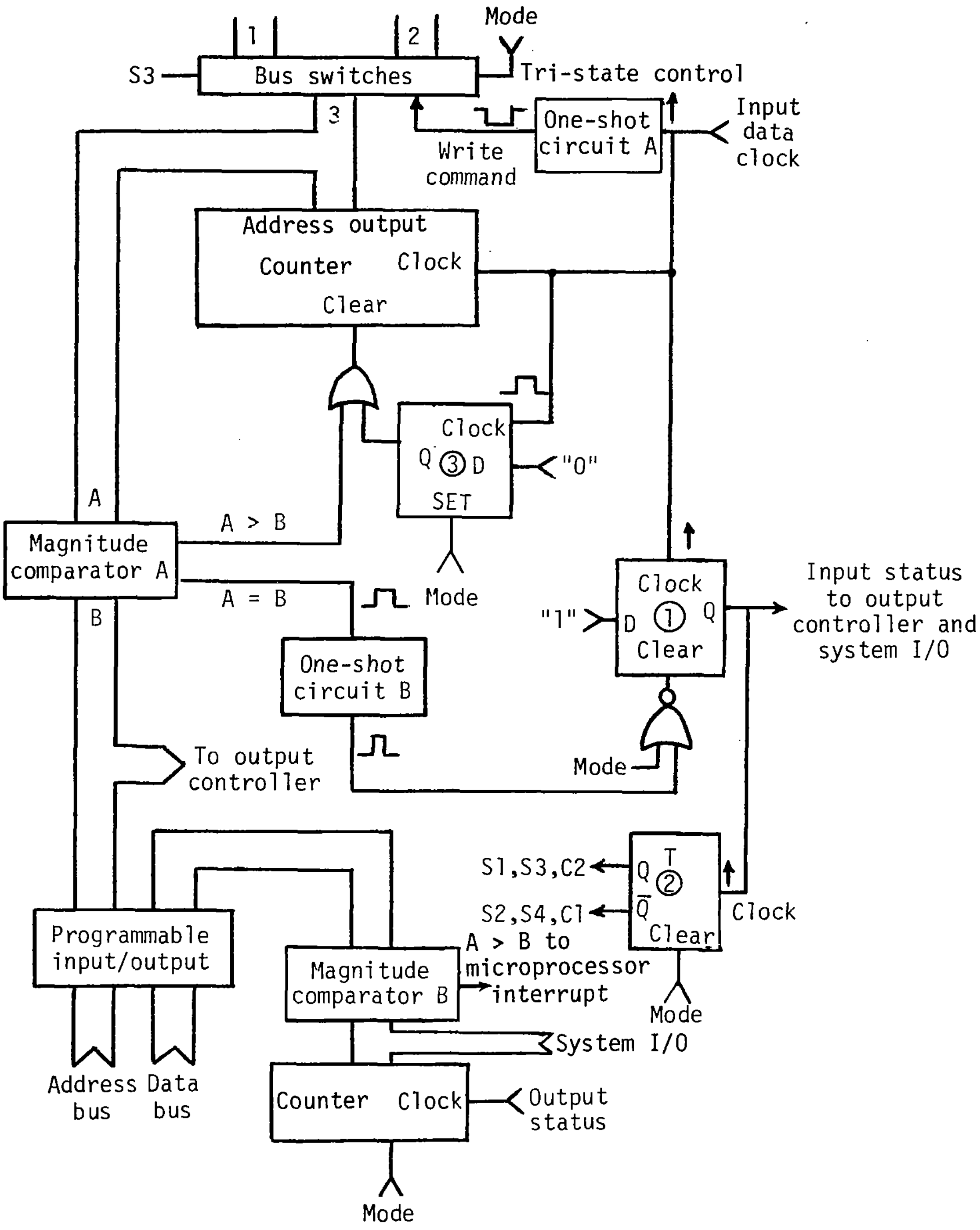


**Figure 4.- Schematic of input controller.**

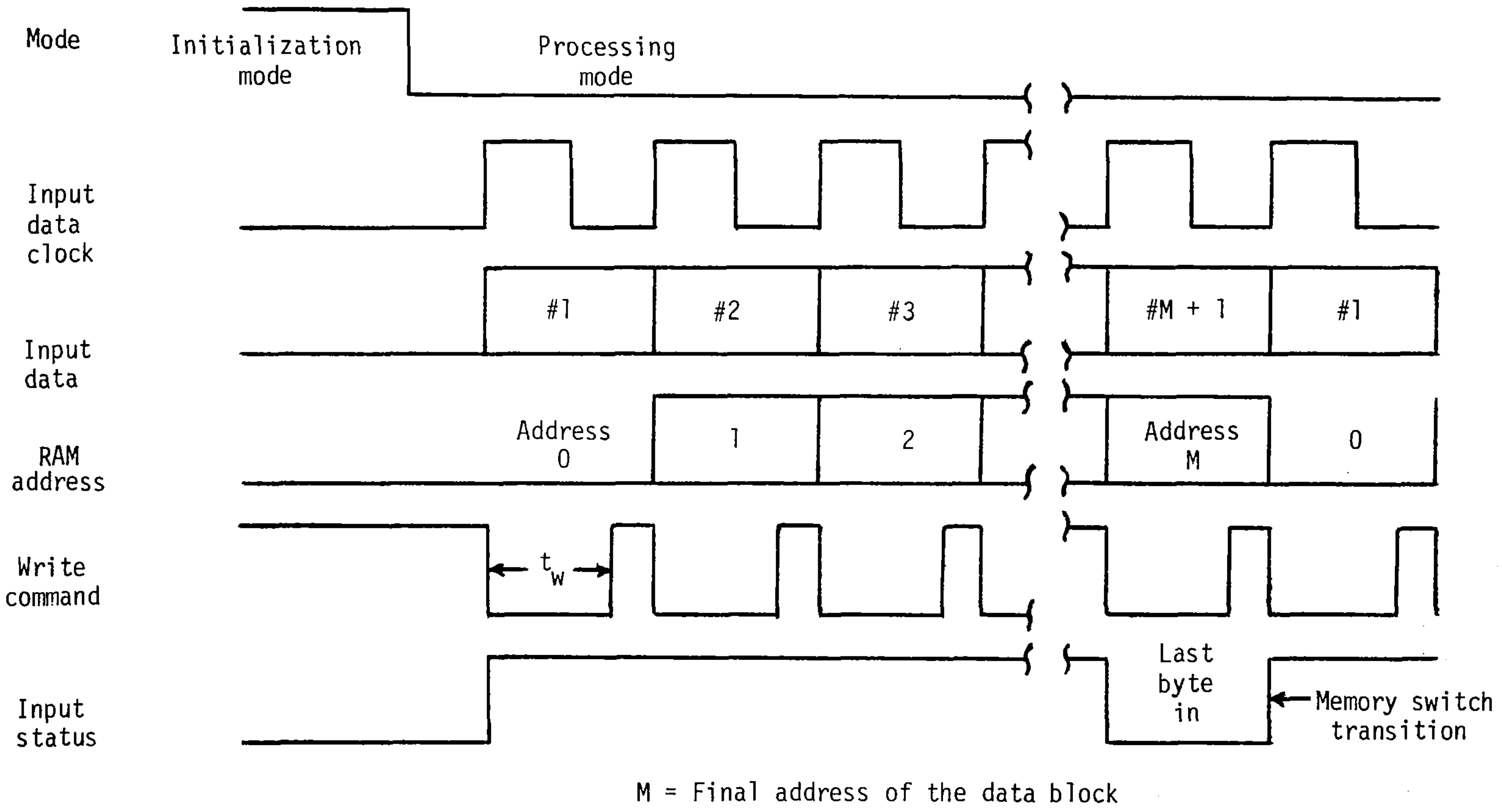


Figure 5.- Timing diagram for data input process.

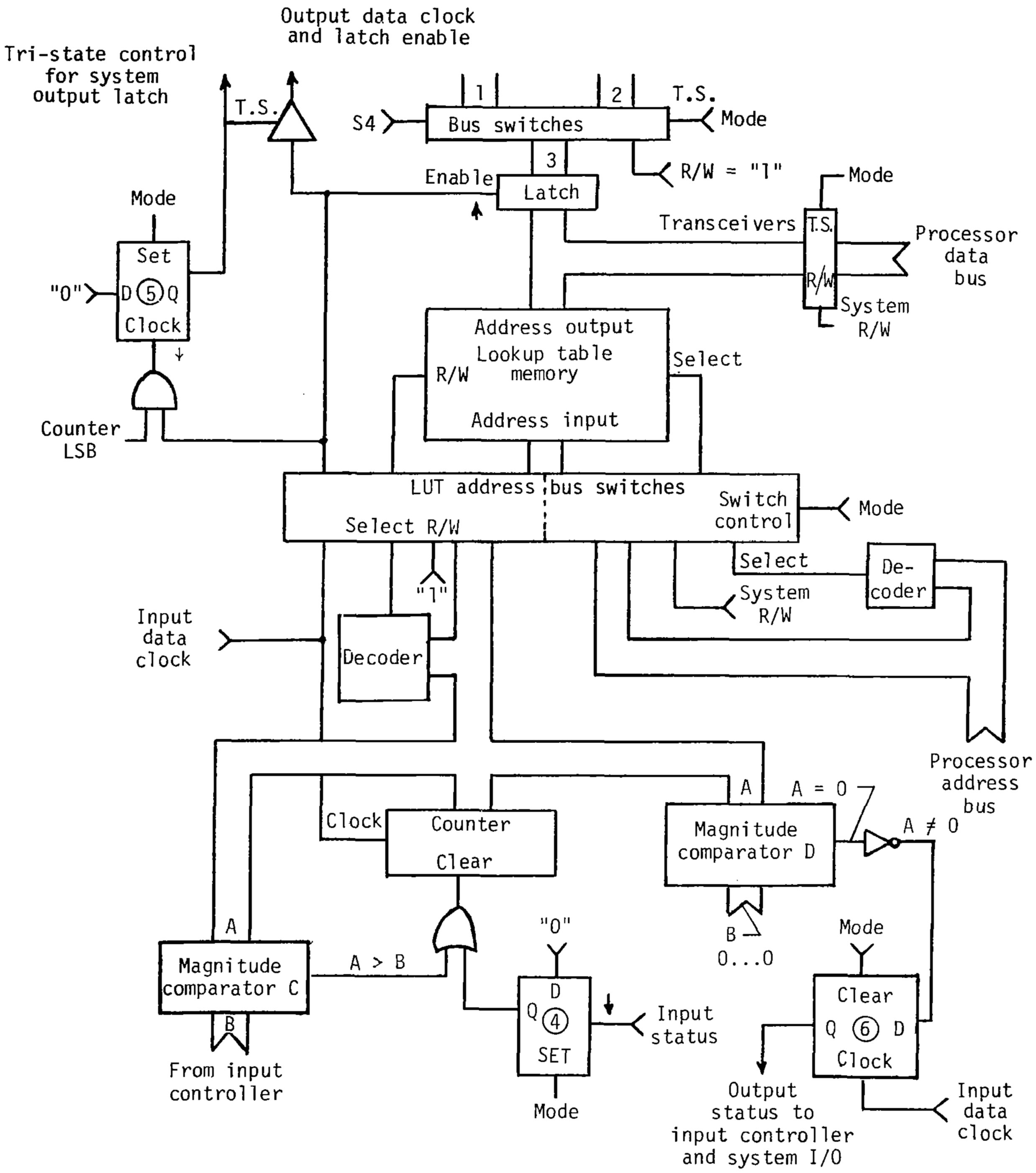
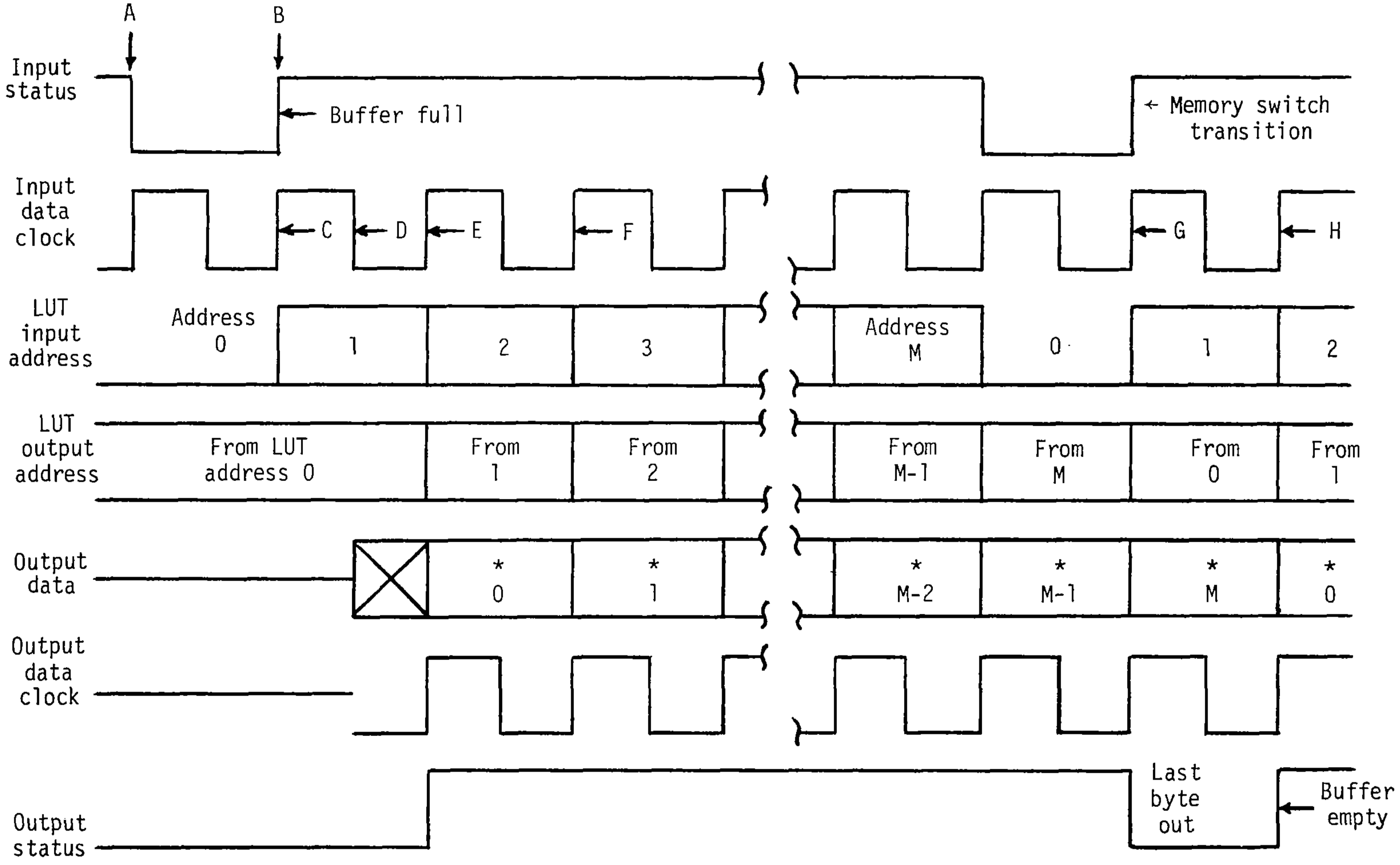


Figure 6.- Schematic of output controller.



* Lookup table address which contains the buffer address of the data byte

Figure 7.- Timing diagram for data output process.

1. Report No.
NASA TP-1870

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle

A DESIGN APPROACH TO REAL-TIME FORMATTING OF HIGH SPEED MULTISPECTRAL IMAGE DATA

5. Report Date
May 1981

6. Performing Organization Code
506-61-53-06

7. Author(s)

Barry D. Meredith and W. Lane Kelly IV

8. Performing Organization Report No.
L-14454

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665

10. Work Unit No.

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546

13. Type of Report and Period Covered
Technical Paper

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A design approach to formatting multispectral image data in real time at very high data rates is presented for future onboard processing applications. The approach employs a microprocessor-based alternating buffer-memory configuration whose formatting function is completely programmable. Data are read from an output buffer in the desired format by applying the proper sequence of addresses to the buffer via a lookup table memory. Sensor data can be processed using this approach at rates limited by the buffer-memory access time and the buffer-switching-process delay time. This design offers flexible high speed data processing and will benefit from continuing increases in the performance of digital memories.

17. Key Words (Suggested by Author(s))

Lookup table
Alternating buffer memory
Multispectral image data
Real-time formatting

18. Distribution Statement

Unclassified - Unlimited

Subject Category 60

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of Pages
17

22. Price
A02

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1981

National Aeronautics and
Space Administration

Washington, D.C.
20546

**Official Business**
**Penalty for Private Use, $300**

THIRD-CLASS BULK RATE

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

U.S.MAIL

4 1 1U,G, 051881 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LABORATORY
ATTN: TECHNICAL LIBRARY (SUL)
KIRTLAND AFB NM 87117

NASA

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return