2

NAS9-11655- 025
MA-129TB

CR-128553

FINAL REPORT

FAULT TOLERANT DATA MANAGEMENT SYSTEM

(NASA-CR-128553) FAULT TOLERANT DATA MANAGEMENT SYSTEM Final Report W.M. Gustin, et al (General Electric Co.) 17 Oct. 1972 98 p (G40P) CSCL 09B G3/08 N72-32211 Unclas 43510

CONTRACT NAS9-11655

This document was prepared for the National Aeronautics and Space Administration Manned Spacecraft Center per Contract NAS9-11655 Data Requirements List, Line Item 8, DRD MA-129TB, date 10/27/70.

Prepared By:

WMGustin

W. M. Gustin
Circuit Development Engineering

Prepared By:

M. A. Smither
Circuit Development Engineering

Approved By:

C. D. Warnick, NASA
Technical Monitor

Spencer Wolfe

S. E. Wolfe, Manager
Circuit Development Engineering

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151

Apollo Systems
Space Division
General Electric Company
Houston, Texas

NOTICE

THIS DOCUMENT HAS BEEN REPRODUCED FROM THE BEST COPY FURNISHED US BY THE SPONSORING AGENCY. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE.

SECTION I

CONTENTS

Illustrations

Summary

The report is divided into two sections.

Section I describes:

a) Results obtained in modifying the On-Board Data Management System software to a multiprocessor fault tolerant system;

b) A functional description of the prototype buffer I/O units;

c) Description of modification to the ACADC and stimuli generating unit of the DTS;

d) Summaries and conclusions on techniques implemented in the rack and prototype buffers.

Section II documents the work done in investigating techniques of high speed (5 Mbps) digital data transmission in the data bus environment. The application considered is a multiport data bus operating with the following constraints:

a) No preferred stations

b) Random bus access by all stations

c) All stations equally likely to source or sink data

d) No limit to the number of stations along the bus

e) No branching of the bus

f) No restriction on station placement along the bus

A. - FAULT TOLERANT PERFORMANCE

1. Multiprocessor Performance

The system is an expansion of the multi-programmed On-Board Checkout System. Supervisior and executive functions were modified by addition of status words and tables in common external memory available to both computers. These status words and tables provide reference information related to the allocation of peripherals, task sequences, current activity level of each task and common data storage areas.

Internal memories of the two computers are identical containing the operating system programs and the active tasks. When tasks are deactivated or activated both memories are identically updated (figure 2).

Each computer attempts to step through the task sequence as if it were operating independently. Prior to executing an element it checks the status in common memory. If the other computer has already accepted that element, it steps to the next element and processes it. This creates an interlacing type of operation with both computers time-sharing the work load.

This system philosophy provides maximum fault tolerant capability. During normal operation, processing is fully shared. If one computer stops for any reason, the other assumes the full load with no interruption.

The SOLVE element was broadly expanded for this system. Trigonometic functions, logical and shift operations were included with the capability for on-line operation.

The COMMAND element was modified to provide proper formatting for the Buffer and to accommodate the various requirements for simplex, dual and triple redundant operations.

A SCAN element provides interface between the system tasks and the buffer operation. This executive interrogates each task as it becomes active for SCAN elements. The measurement ID's associated with each SCAN are placed in a table with their associated buffer memory locations. Process ID's in the memory map are modified where desirable. Each pass through the executive, new data in the change buffer is read-out and routed to the appropriate task for whatever processing is required.

A Load and Execute subroutine was written to utilize the graphic capability of the plasma display. Ordinate and abscissa limit parameters may be entered into a standard subroutine which will then draw the coordinates on the display. Successive inputs of X-Y values into data registers D0 and D1 of a task and execution will result in point to point line graphing.

To determine system response, time tests were run to determine throughput time. One computer was loaded with the fault tolerant system and the second computer monitored time to complete an executive pass. The executive pass is the time to complete one element in each of the fourteen tasks. The sequence is from the executive to task 1, to the executive, to task 2 and continuing through task 14 and back to 1.

Worst case loading is when the buffer is being scanned on each executive pass. With each buffer task contained four SCAN elements, the buffer was loaded at about 1000 samples per second except where otherwise noted.

Average results are listed below for one complete executive pass.

Item	Time	Description
1	4.2 MS	No buffer scan and no active tests

Item	Time	Description
2	4.5 MS	Scan buffer with no active tests and no data entering buffer
3	6.0 MS	Scan buffer with 1000 S/S entering change buffer--no active tasks
4	7.2 MS	Scan buffer with data-one active buffer task
5	7.6 MS	Scan buffer with data--2 active buffer tasks
6	8.4 MS	Scan buffer with data--8 active buffer tasks
7	8.7 MS	Scan buffer with data--8 active buffer tasks and 4 non-buffer monitor tasks

Some other characteristics of the system were noted during these loading tests. The concept of the change buffer is to provide only significant data for computer processing. Limit testing is performed by hardware in the buffer to eliminate redundant data and eliminate this overhead processing by the computer.

During system testing it was observed that high volume input to the change buffer had a tendency to overpower the system. Input to the acquisition devices was a function generator sine wave producing a steady load. More normal operation would assume bursts of data allowing more rapid recovery of the system.

Two things are suggested to minimize this condition. Floating tolerance utilization in the memory map provides only significant data and is recommended as the primary means of obtaining significant data. Use of fixed limit testing should be minimal and used only where a specific requirement exists. A memory built into the hardware to inhibit multiple entries of out of limits data would be desirable. For example, when an out of fixed limits condition is recorded into the change buffer a flag would be set in the instruction word. Subsequent out of limits data would be inhibited if the flag was set. The next time the data value was within limits, the flag would be reset resuming normal operation.

2. Independent Simplex Mode

The design goals for the simplex mode were, (a) to allow each computer system to operate totally independently of the other during concurrent operations, (b) allow reconfiguration to other modes of operation without physical recabling or other extensive reconfiguration of the system.

Full independence of the two systems was not possible because of the requirement to share common peripherals on the I/O bus. Specifically, these were the card reader, typewriter, printer and magnetic tapes. These restraints had to be worked around by the operators while using the system.

Another operating problem occurs when using the computer system control panels. As the external memory and I/O channels are shared, a system reset from either control panel likely disrupts the other system. By coordinating the operators activities at the control panels, this restraint is negated.

The external memory, ACADC and buffer are all accessible from either computer on a non-interference basis. No recabling or physical reconfiguration of any type is required to change modes. Except for the relatively minor operating restraints noted above, mode to mode reconfiguration is an exclusive software function and concurrent simplex operation by both computers was achieved.

3. Multiprocessing Mode

When the fault tolerant system is entered into the computer, all pertinent status information and buffer registers are loaded into external memory. To enter a multiprocessing mode, computer memory from the initially loading system is transferred over the I/O channel to the idle computer and operation is initiated.

Based on the status indicators in external memory, both computers share task execution. Static information for the system is retained as identical copies in the internal memories of each machine. This information includes task elements for all tasks, supervisor, executive and resource control.

The external memory contains status words indicating availability of I/O devices, pointers defining the next task elements to be performed, task data, buffer registers and scan words from the buffer. Basically, all the active information necessary for both computers to share task assignments.

Sharing of tasks is done by each computer performing a task element. When an element is complete, the computer tests external memory to determine which element is next in line and flags that it is doing this element. The two computers then alternately sequence through tasks performing elements concurrently.

An element like SOLVE may require extended time to complete if it is complex. In this case, the other computer may complete two or three elements during the time frame. If one computer performs more than three successive elements from the task table, it assumes malfunction of the other system and shuts it off. The operating computer still operates in the multiprocessing mode. Thus, as soon as the other computer is restarted it immediately has all the information it needs to re-enter the processing cycle.

4. Triple Redundant Mode

Under software control, three separate buffer areas may be reserved for command words. Each of these areas is reserved for a specific computer and locks out write access from any other source. Establishing the lock-out area enables the Hamming encoder so any word written into the area is encoded for transmission.

When a COMMAND element is encountered by a computer sequencing through the task list, it flags that it is writing its copy of the command into the buffer. The second and third computers must complete identical sequences before the COMMAND element is satisfied. After notification that all three entries in the buffer have been completed, the third computer sends an execute word to initiate the command. The sequence of the computers is not significant and in fact may be different on subsequent redundant operations.

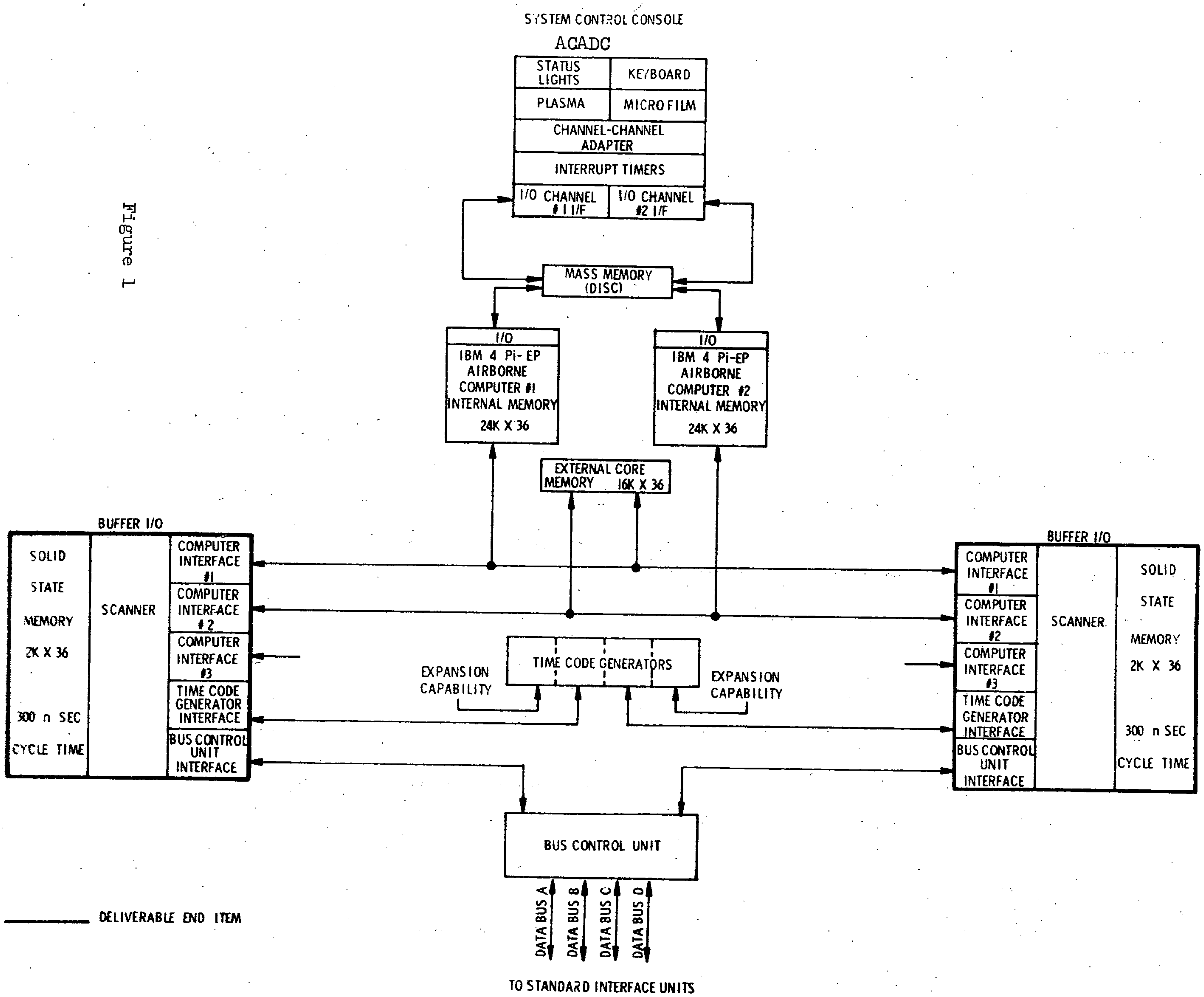
Common tables for three computer operations are contained in external memory. As only two computers are physically part of the system, the third computer is simulated by the second computer.

Upon receipt of the execute word, the three copies of the command in the buffer memory are accessed, majority voted and transmitted to the user. Transmission includes the command word stored by the computers and the encoded parity. Any voting error causes a status word to be written into the change buffer indicating which word was in error.

5. Dual Redundant Mode

Entry of the system into dual redundant mode causes entry of two copies of commands into the buffer by two different computers. Specific blocks of words are dedicated to each computer and attempts to write into the area by any other user is locked-out.

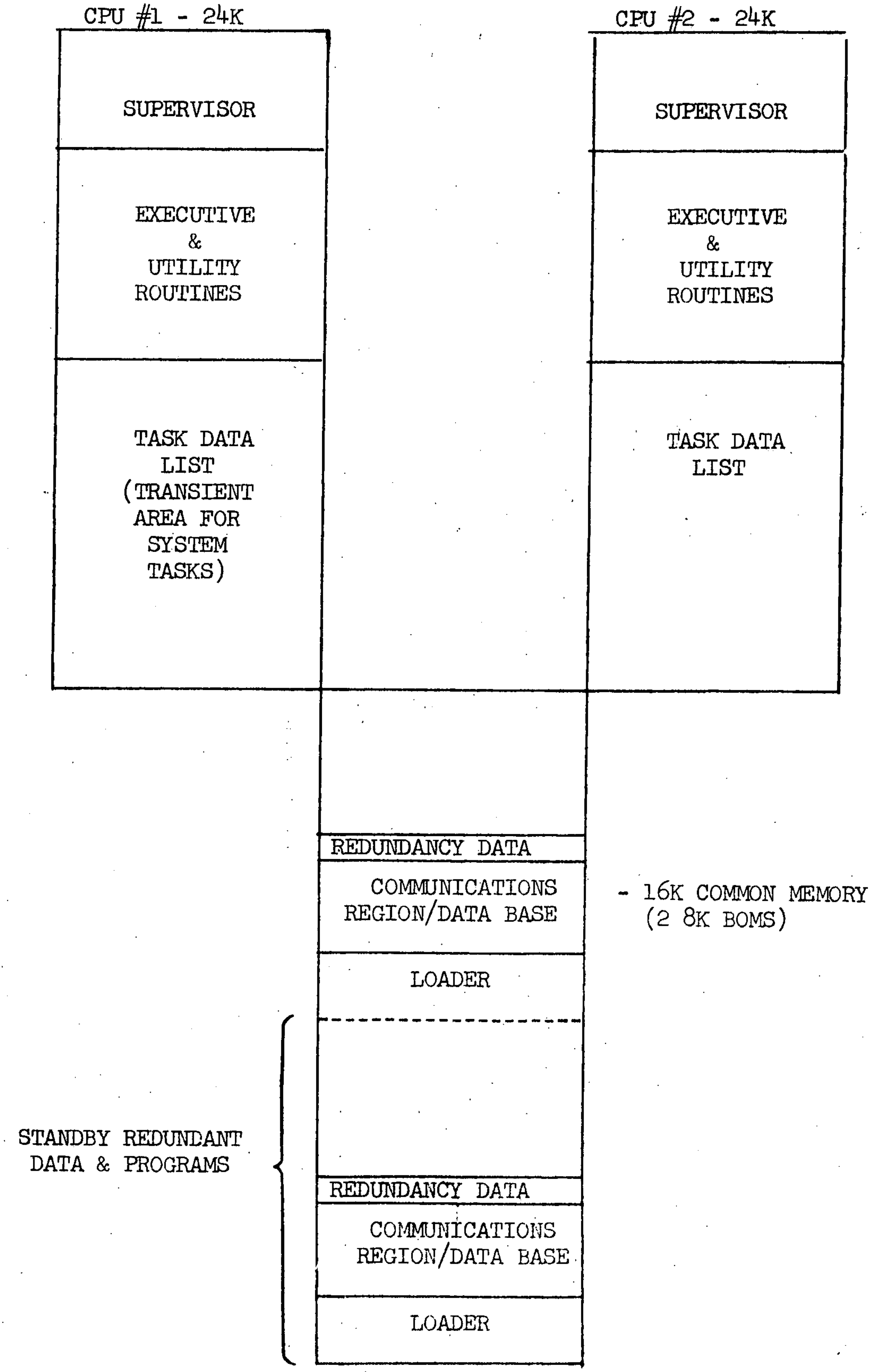
The first computer to encounter a COMMAND element writes its copy of the command word into the buffer. The second computer writes its copy into the buffer and executes the instruction. The sequence of the computers may be different on subsequent elements.



FAULT TOLERANT MULTICOMPUTER CONFIGURATION

Figure 1

FIGURE 2
MAIN STORAGE ALLOCATION FOR
THE DMS OPERATING SYSTEM


CPU #1 - 24K
SUPERVISOR
EXECUTIVE & UTILITY ROUTINES
TASK DATA LIST (TRANSIENT AREA FOR SYSTEM TASKS)
CPU #2 - 24K
SUPERVISOR
EXECUTIVE & UTILITY ROUTINES
TASK DATA LIST
REDUNDANCY DATA
COMMUNICATIONS REGION/DATA BASE
LOADER
- 16K COMMON MEMORY (2 8K BOMS)
STANDBY REDUNDANT DATA & PROGRAMS
REDUNDANCY DATA
COMMUNICATIONS REGION/DATA BASE
LOADER

B. - HARDWARE DEVELOPMENT

1. Stimuli Generation

The Stimuli Generator Unit (SGU) of the Digital Test Set was modified to accept encoded command words from the command controller and generate programmed stimuli. The Hamming decoder and correction circuitry identical to that developed for the Stimuli Switching Unit was incorporated.

Common command and response buses go to both units from the controller. Upon receipt both units perform error detection and correction on the received word. The addressed unit scans back the command with a flag bit set if error correction has taken place. If an address error or dual bit error is detected, no response is returned. If no response is received, the controller retransmits. After two failures, an error word is written into the change buffer and the operation terminated.

2. Advanced Control and Display Console (ACADC)

Modification of the ACADC provides control for the fault tolerant system. Major hardware components include:

a) System status display of 20 indicator lamps indicating general operating status and system response to operator inputs;

b) Plasma display capable of operating in either graphic or character modes used primarily to display operator input and task display information;

c) Multiparameter display providing detail information to the operator in the form of rear projection microfilm frames;

d) An alphanumeric keyboard providing operator entry capability;

e) Dual I/O channel interface allowing access by either computer to any device;

f) A channel to channel interface allowing computer to computer data transfer on the I/O channel;

g) Interrupt timers capable of generating real-time or time interval interrupts.

The first four items were part of the original equipment and remained essentially the same. Computer interfaces, channel to channel capability, interrupt timer and redesigned control units were added to fulfill requirements of the fault tolerant system.

3. Prototype Buffer I/O

Two prototype Buffer I/O's were fabricated, each having five basic elements as shown by the block diagram in figure 3. There are three IBM 4 pi EP computer memory bus interface units, a Time Code Generator interface, and a Bus Control Unit Interface. All interface units have access to the solid state memory through the memory scanner.

The function of each unit is described in subsequent paragraphs.

Memory Scanner

The memory scanner provides four basic functions:

a) Common busing from eight ports to memory.

b) Port selection control to provide time-shared access to memory.

c) Data input control to a significant data change buffer.

d) Enables lockout for specific memory blocks.

Data buses are illustrated on figure 4.

a) Memory Data In Bus - 36 lines including 32 data and 4 parity bits shared by all ports to write data into memory.

b) Memory Data Out Bus - 36 lines including 32 data and 4 parity bits shared by all ports to read data from memory.

c) Address Bus - 16 lines including four mark bits for byte selection and 12 address bits for word selection.

d) Read/Write Bus - A single line indicating a read or write operation from the interface to the memory.

e) Change Buffer Write - A single line that may be used by any interface to access the change buffer on a write cycle.

In addition to the common buses available to all units, each interface has two unique control lines. These are a Request and Recognition of Request. When an interface requires memory access, a Request is issued to the Memory Scanner. The scanner, continually monitoring the request lines, acknowledges the request by returning a Recognition of Request (ROR). This signal can occur as soon as 30 nanoseconds after the request was raised or as long as 2.8 usec later depending on the activity of the scanner. Receipt of the ROR tells the interface that all buses are dedicated to it and its memory request is being processed. Either a read or write request requires about 400 nanoseconds to process. On a read cycle, termination of the ROR indicates valid data is on the bus for the requestor. On a write cycle, termination of ROR indicates data has been stored in memory. When the ROR becomes inactive, the interface must disconnect and is allowed only passive access to the data and control buses.

Port assignment to the scanner are listed below:

Port	Description
0	Computer Interface Unit #1
1	Computer Interface #2
2	Computer Interface #3
3	Computer Interface Unit (unassigned)
7	Time Code Generator Interface
4	Bus Control Unit Interface
5	Unassigned
6	Unassigned

Change Buffer Control

The change buffer provides a block of words in memory into which significant data may be written. Location and size of the buffer are under software control. A block diagram for the change buffer control is shown in figure 5.

The starting address and block length are preset into the address counter by addressing byte #1 of memory location 004 and initiating a write cycle. The change buffer control logic continuously monitors the memory bus lines and when it detects this condition the decoder output presets the six most significant bits of the counter and a two bit latch which determines the size of the change buffer block. The latch outputs are decoded to provide a signal which causes the counter to recycle when the buffer block has been filled. The block size and the corresponding code are as follows:

00	16 words
10	32 words
01	256 words
11	512 words

The outputs of the address counter (Q_0 thru Q_{11}) are routed to multiplexers where either they or the input address bus are selected for addressing the memory. The selection is determined by the level of the "Change Buffer Write" signal. When the "Change Buffer Write" signal is activated the address in the counter will be transmitted to the memory. When the "Change Buffer Write" line returns to the quiescent state the trailing edge increments the address counter.

Operation of the change buffer was simplified from the rack buffer for the express purpose of circuit and hardware reduction. This simplification provides some operating restraints. If a buffer size of 256 words is used, the first address must be address 256 or direct multiples of 256. If a word length of 512 is used, the first address must be 512 or multiples of 512.

The above restraints are accommodated by presetting only the most significant four bits to define the first address of 256 word buffer or the most significant three bits for a 512 word buffer.

The buffer must be located such that the first word address plus the buffer length does not exceed 4095. This eliminates any attempt of the change buffer to write into the low storage dedicated locations.

Lockout Control (Figure 6)

The lockout control logic provides the means whereby selected blocks of the memory can be protected by inhibiting a write access to this block by any user other than a specific computer. This prevents inadvertent destruction of critical control and command words stored in this area.

A computer may lock-out any block of 512 words. The only exception will be the first 16 locations used for system control functions.

Solid State Memory

The memory unit uses Monolithic Memories, Inc. MM6510 Bipolar RAM dual-in-line packages with the following characteristics:

a)	Word capacity	4096 words
b)	Word length	36 bits per word
c)	Addressing	Random access
d)	Cycle time	300 nanoseconds max. per word

The word format is identical to the memory word of the IBM 4 pi EP computer.

Memory packages are electrically arranged in four arrays of 16 x 9. The address matrix will select a 36 bit parallel word on a read cycle. On a write cycle, one word in each array will be selectively accessed depending on the mark bits as follows:

1 2 3 4	
1 0 0 0	write byte 0
0 1 0 0	write byte 1
0 0 1 0	write byte 2
0 0 0 1	write byte 3

Only the byte(s) selected will be written with the other bytes retaining their previous data. Writing a full word requires that all four mark bits be set to 1.

Computer Interface

The computer interface transmits and receives data in parallel on the IBM 4 pi EP memory bus with the Buffer I/O appearing to the computer as an external memory bank, three interface units have been implemented.

If the computer requests a read operation, the interface requests memory. As it may have to wait for its turn through the scanner, response to the computer is inhibited until it has completed access of the word from memory. At this time it allows the sequence to continue by responding to the computer. The interface is immediately available for another computer request.

For a write operation, the interface responds immediately to the computer request and loads the data into an input register. Memory access is then requested through the scanner and the word written into buffer memory. Until this write cycle is complete, the interface will not respond to a subsequent computer request.

Although provisions have been made to accommodate possible delays in data interchange between the computer and memory, the actual total effect on delaying the memory bus operation is negligible.

The computer memory access time is a minimum of 2.5 microseconds per word. Worst case access to the memory in the buffer will be if all ports request access simultaneously and the scanner has just passed the computer interface. Each port requires 400 nanoseconds for access, thus a maximum delay to service the other seven ports is 2.8 microseconds.

In an operation with only the computer interface requesting memory access, the worst case time delay is 700 nanoseconds (100 per port) and memory access time of 400 nanoseconds or a 1.1 mocrosecond response.

Time Code Generator Interface

This interface, illustrated on figure 7,

a) accepts time data from a Time Code Generator and stores it into dedicated memory locations,

b) accepts preset or adjust data and transfer it to the Time Code Generator, and

c) writes time into the change buffer with each data entry.

Three memory words are dedicated to the Time Code Generator.

0	Most significant time from TCG
1	Least significant time from TCG
2	Preset and adjust time input

The one millisecond level of the TCG is monitored by the interface. When a change is detected in this value, the current time is read into the interface and stored in memory locations 0 and 1.

For a preset operation, the Address Bus is monitored for write requests into location 2. When recognized, the data on the Data In Bus is stored in a holding register and transferred in parallel to the Time Code Generator.

A fine adjustment operation is implemented in the same manner as preset and may be used to decrement or increment the value in the holding register as opposed to a preset loading operation.

The Write Change Buffer bus is continuously monitored by the TCG interface. When a word is written into the change buffer by another user, the TCG interface will request memory and write the current time into the next address of the change buffer.

Data Acquisition and Command Controller

The Data Acquisition and Command Controller combines the basic functions of the Demand DAU and Command subsystems of the rack buffer. The concept of a stored program memory map is retained for both command and data acquisition sequences. Execution of any sequence is controlled by the computers.

Data Acquisition

For acquisition sequences, the memory map consists of two instruction words plus the number of data words appropriate for the instructions (1, 2, 4 or 32 words). Instruction words contain control information to be transmitted to the SIU and control information for the DACC instructing it as to the type of word transmitted and the response anticipated.

Received data is subjected to floating and fixed limit tests. Limit parameters are retained in the memory map data word. Failure of either or both limit tests will cause a word to be written into the change buffer identifying the measurement point and value. Failure of floating limits will also update the value in the data field of the memory map word. This provides the capability for the computer to access directly from memory the last significant value of any measurement point.

Implementation of limit test controls is modified from the techniques used in the rack buffer. Control bits were used in the rack buffer to active or deactivate specific limit checks. For the prototype buffer, fixed limit tests are performed on all data. If the user does not want these tests the limits are set to the maximum values thus nullifying the effect.

Floating limit tests were allowed to range from zero to full value in the rack buffer. Such a broad range is not considered realistic. In the prototype buffer, limits range from 0 to about 21% of full scale with 3% increments. For a 5 volt value, this corresponds to limits up to 1 volt in increments of 150 millivolts. Inhibiting floating limit tests is programmable.

These modifications simplified the hardware and software by eliminating unnecessary redundancy of control to achieve the same results.

Data Update

Update sequences require two instruction words and a data word containing 1, 2 or 4 bytes of update information dependent on the word type to be transmitted. The instruction words contain control information for the SIU and for the DACC. In addition to the various word formats that may be transmitted, the DACC may operate in either a simplex or triple redundant mode.

In a simplex mode, single copies of the update word are accessed from memory and transmitted. In a triple redundant (triplex) mode, three identical copies are accessed from memory and majority voted prior to transmission. The mode of operation is selected by the execute word. For testing and checkout purpose, all update sequences are loaded triple redundantly in the memory map. The sequence may then be executed in either mode.

Block Data Transfer

During the conceptual phase of development, a need for block data, transfers on the bus appeared a distinct possibility. Potential users might be display systems or remote processing stations. To accommodate this possibility, the four byte update word was selected because of its efficiency and compatability with the computer memory word. In a block transfer mode, up to 255 memory words may be transmitted to an SIU with one instruction set.

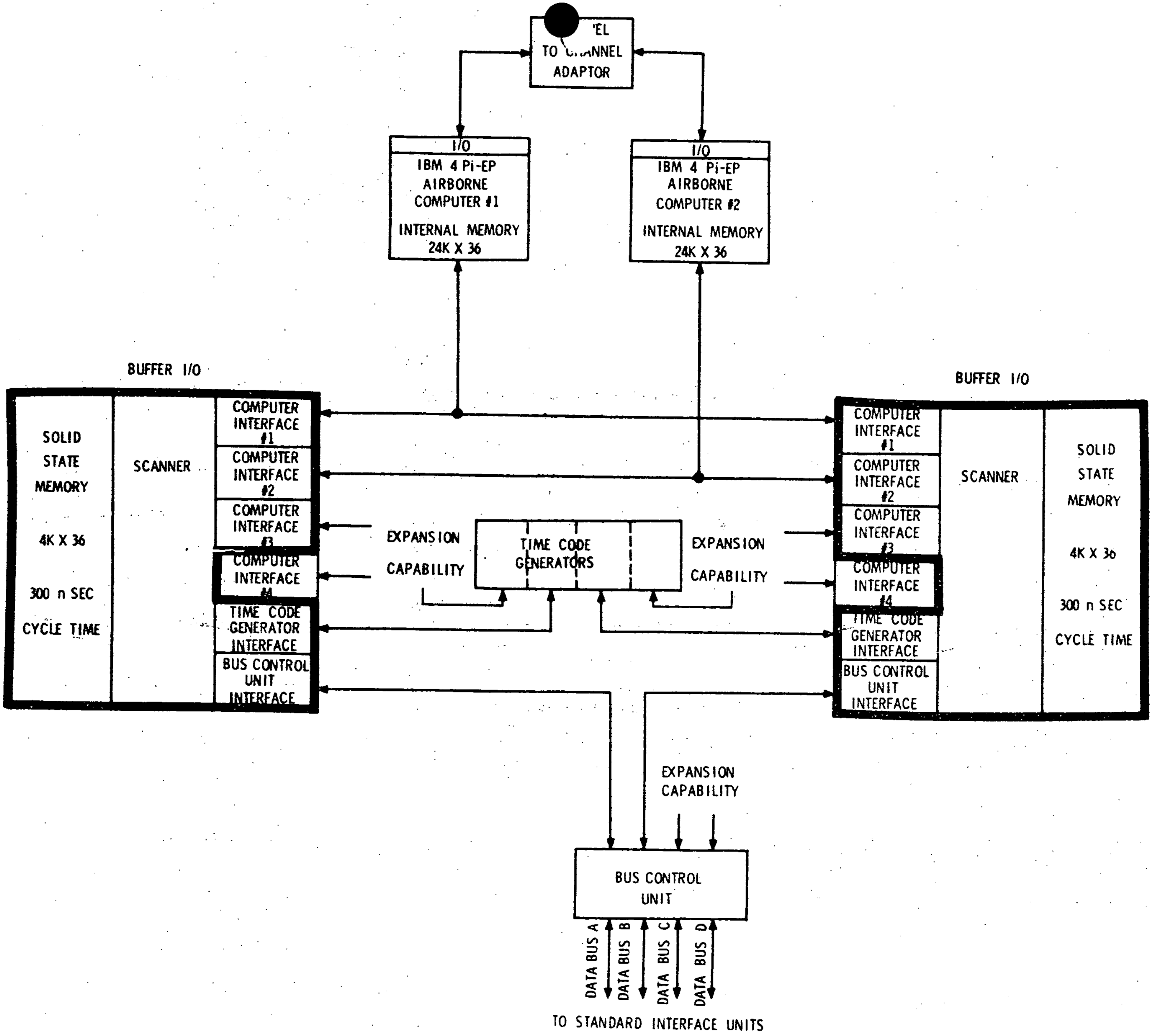
Error Detection and Fault Tolerant Capability

Provisions were incorporated to detect gross equipment malfunctions and report these in the form of status words to the computer. When possible, an attempt is made to complete the sequence even though an error has been detected.

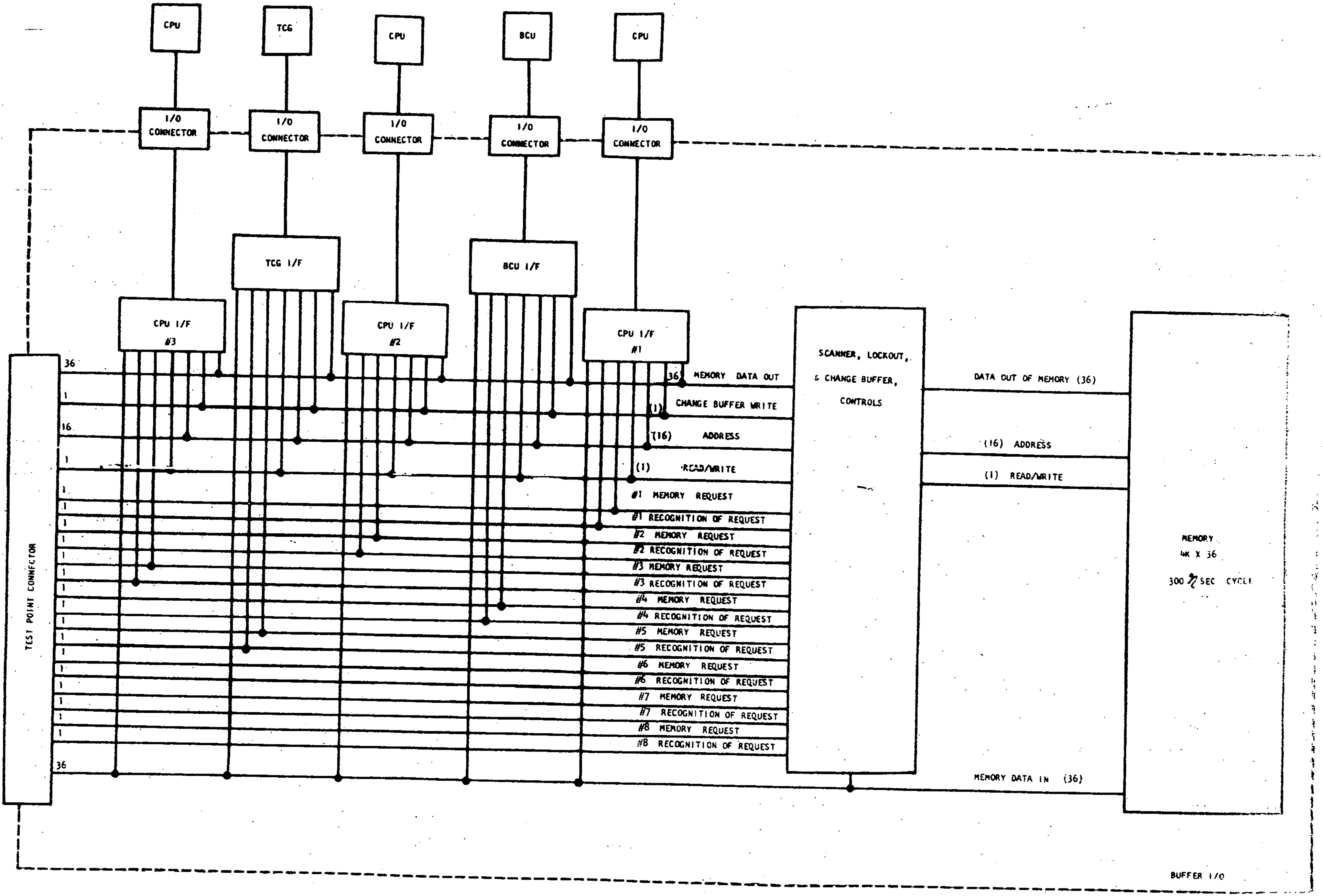
Parity errors. Each 8 bit byte in memory has an associated parity bit. When a word is read, parity is checked. If an error is detected, an error word written into memory indicates the failed word. If the error is detected in an acquisition instruction word, simplex update or data word the operation is terminated. If the error occurs in a triplex update, the operation proceeds allowing the voter to correct the error. On acquisition data words, operations on that word is terminated but the sequence continues to the next word.

Polynomial code error detection. Response words on the bus are encoded as received and checked for transmission error. If an error is detected a word is written into the change buffer indicating the failure and the operation is terminated.

Time out. When no bus response occurs within an allocated time period a error word is written into the change buffer.

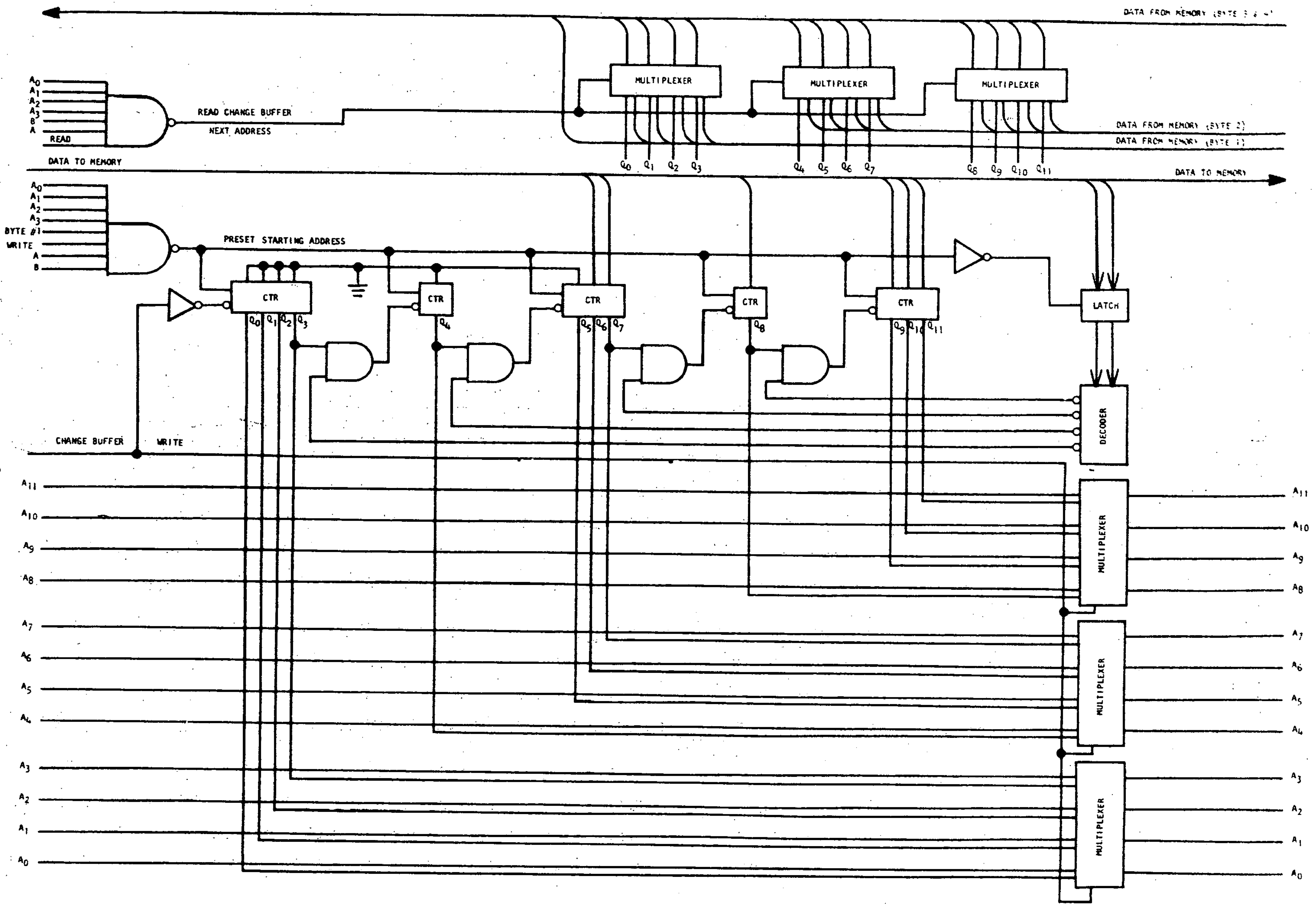


BUFFER I/O BLOCK DIAGRAM
Figure 3



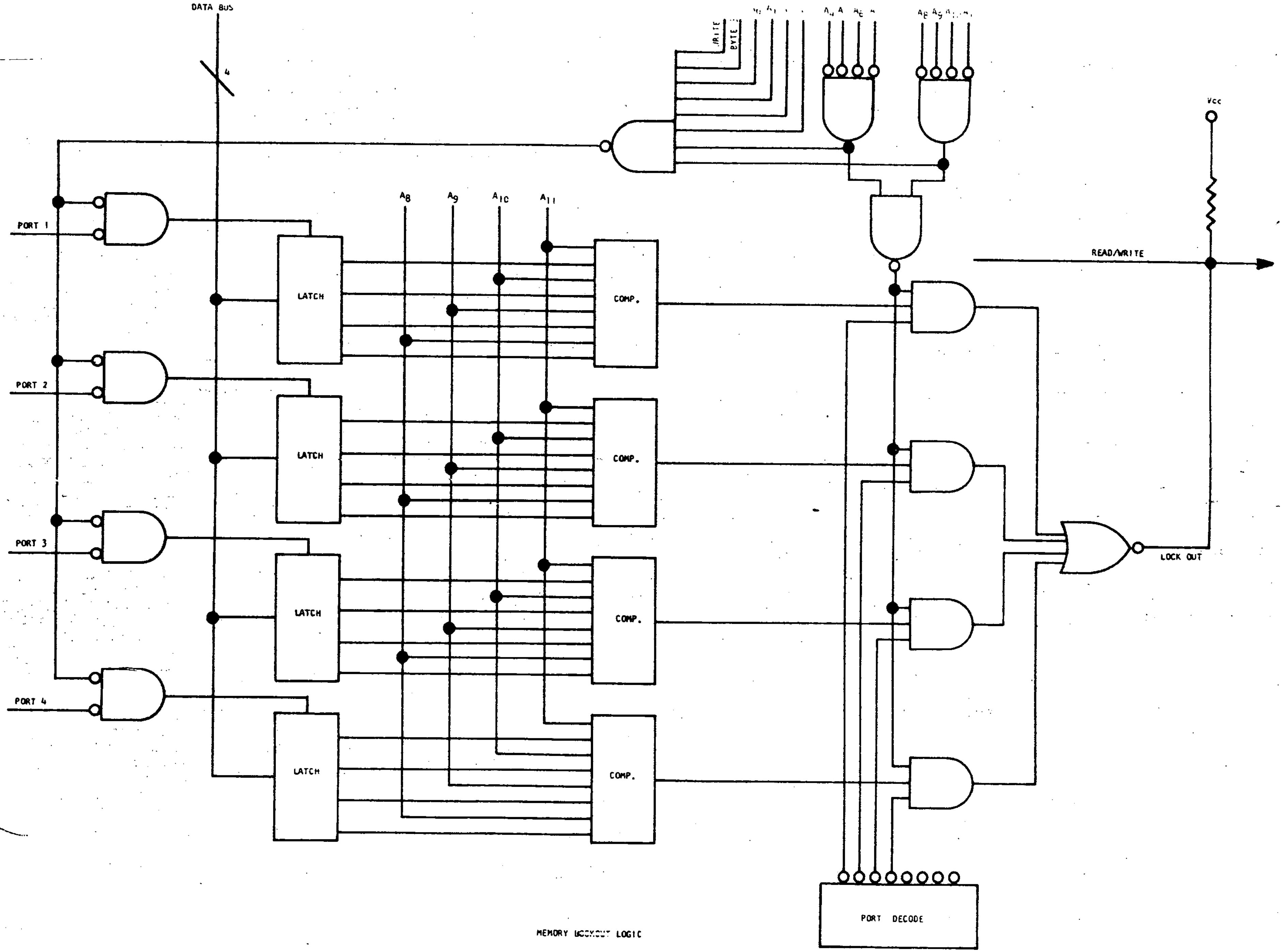
MEMORY SCANNER

Figure 4



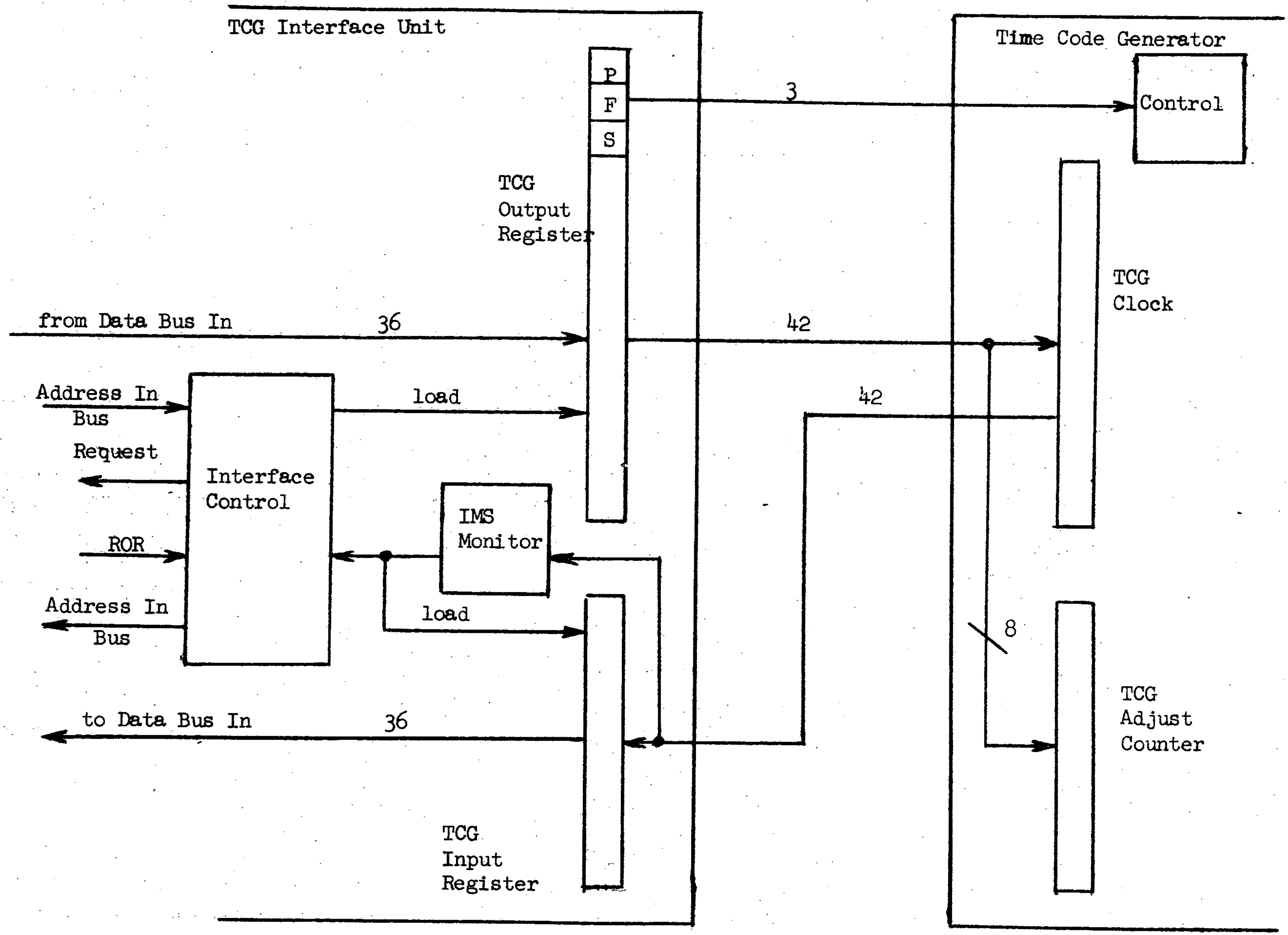
CHANGE BUFFER CONTROL

Figure 5



MEMORY LOCKOUT LOGIC

Figure 6



Time Code Generator Interface

Figure 7

Packaging

One of the primary objectives of the prototype buffers was to miniaturize the design and obtain high density packaging. Significant steps to achieve this were reduction of power supply requirements to a single voltage and time shared busing to eliminate cables. The other major factor is use of stitch-welding techniques developed by the Micro Technology Division of Sterling Electronics.

All logic consists of TTL dual in-line packages mounted on daughter boards fabricated by Micro Technology. Each board provides for 153 packages in a 17x9 matrix. Heat sinks are mounted on the DIP side of the board so the inserted package is in contact and provides a thermal dissipation path through the bottom of the package. The heat sink extends out to two edges of the board and contacts ~~the~~ cold plates when inserted.

The enclosure is divided into a power supply compartment on the bottom, a mother board serving as a divider and a common busing plane for power and signal routing to the logic boards and external connectors. The upper compartment contains capacity for 13 logic panels (daughter boards) that may be inserted into the mother board.

Two sides of the enclosure are cold plates against which the heat sinks of the daughter boards and the power supplies are flush mounted for heat dissipation.

C. - RACK BUFFER EVALUATION

1. PCM Decommutator

The decommutator was built with the capability of accepting two input channels simultaneously. Test and evaluation were accomplished with an Apollo PCM commutating unit input to both channels. Each channel had a unique memory map with switch options for either channel to work on either memory map or both on the same memory map. Other options include acceptance of up to 256 words per frame, 8 or 10 bit data words and time slot selection of the frame word count.

For this development, a dedicated memory was assigned to the PCM unit which was time-shared by the two channels. For an operational configuration, a non-dedicated memory configuration may be desirable. Time-sharing is an option which may or may not be desirable for a specific system configuration. Time-sharing has the advantage of using only one port of the memory scanner for both channels of PCM. If multiple scanner channels are available, independent decommutators have the advantage of full system redundancy.

The most useful innovation was utilization of indirect addressing techniques with super-commutated data and for structuring the memory map to locate the next time slot instruction word.

For super commutated data, a flag bit is set and the address refers back to the first time slot data. A 200 S/S data point has the first instruction set containing all processing instructions plus the last significant value. The second, third and fourth time-slot instruction words refer back to the first. This function eliminates the necessity for computer correlation of the four data words. Additionally, any supercommutated rate may be accommodated by the basic frame rate block couter. For this system, with a basic rate of 50 S/S, sample rates of 50 x 2^n S/S may be accommodated.

Because of its serial nature, bit failures in the next address or block counter fields cause a catastrophic out-of-sync condition. This was particularly noticeable as the decommutator development was in conjunction with the solid state memory development. With this combination, intermittent bit drop-out or pick-up was relatively frequent during the debugging stage. These subtle memory map changes are very difficult to isolate as initially it appears the decom circuits have failed and all evidence of the actual failure is frequently obliterated. To achieve much greater reliability, a single bit error correction, double bit error detection code protecting the next address and block counter fields would be most desirable. The code bits would be encoded during assembly of the memory map, retained in the memory and decoded each time they are read out by the decommutator.

The decom may be implemented using shared or dedicated memory. A dedicated memory requires an estimate of the memory size requirement. A survey of current Apollo and projected future PCM requirements indicates a relationship of about 10 samples/second per measurement provides a reasonable rule of thumb figure. Based on Apollo PCM providing 6400 time slots per second, 640 measurements can be accommodated. The implemented decom requires 2 memory words per measurement or a total of 1280 memory words per channel.

2. Adaptive Subsystem

The adaptive system was implemented to demonstrate that a free-running data compression scheme could be implemented into a system and to determine the operating system restraints. A more thorough treatment of the adaptive system operation and the expected results of the remote analog zero-order predictor (AZOP) is contained in the Final Report, Contract NAS9-9681, document NAS9-9681-153, and the OCS Development Data Compression Report, NAS9-9681-106.

The remote adaptive system incorporating both analog and discrete zero-order predictors performed as predicted by the breadboard and advanced study and were readily implemented. Transmission on the bus is reduced to only significant change data with no polling requirement.

The system was intentionally designed to operate as rapidly as possible and fully implemented achieves throughput rates of about 85,000 measurement values per second. As might be assumed, and was anticipated the potential volume of data tends to overpower both the hardware and software systems. The Fault Tolerant software systems scans the change buffer at 1 millisecond intervals retaining only one sample from a given measurement point even though several may appear in the change buffer.

To achieve better compatability with the overall system, the scan rate could be reduced about one order of magnitude to achieve an output measurement rate of about 10K values per second and still provide high measurement resolution.

3. Command Subsystem

Several different techniques were implemented to assure integrity of the command words to be transmitted. These included:

a) Hardware lockout of designated memory blocks that allow write access only by a specific computer;

b) Triple redundant storage of commands with majority voting to form a simplex transmission word;

c) A Hamming block code with single bit error correction and dual bit error detection capability.

Each of these controls was readily impelemented and operated trouble free during the evaluation period. Lockout, triple-redundant command storage and voting were implemented into the prototype buffer. The error correction code was not implemented into the prototype buffer because of the data bus requirements of that system.

Encoding is accomplished in the memory scanner as a hardware function thus protecting the word in residence and during buffer operation. The approach may be expanded back through the system and have the encoding a software function performed early in the generation cycle. For example, the data could be encoded prior to buffer transfer or even as a function of an off-line compiler if appropriate for the system. This approach is recommended to protect specific address fields in a PCM decommutator and the reader should refer to that section.

The hardware or software implementation would appear to be primarily based on frequency of use and system concept. Hardware implementation was done in this instance to make the buffer a totally independent unit. If the real-time frequency of encoding is high, the system operation is more efficient with a hardware encoder. If the frequency is low, software encoding provides more protection through the system.

In any event, for critical data, the code provides a high level of confidence and allows the using device the capability to determine the validity of data received.

D. - CONCLUSIONS

1. Power Distribution

Problems of line drop, distribution of various supply voltages, and incompatability of voltage levels between logically connected drawers were encountered in the rack buffer. To minimize these problems in the prototype buffers, cabling requirements were virtually eliminated, power plane distribution for logic and single voltage levels were utilized. Spare pins on the logic modules were connected to V_{cc} or ground to facilitate any required installation of decoupling capacitors.

2. Multiport Scanning

On the prototype buffer, a time shared busing scheme is used as opposed to the individual port gating employed in the rack buffer. This shifts the responsibility of controlling activation of data lines from the scanner to the user interface.

The actual implementation provided one common bus on the mother board with full access by any subsystem located on daughter boards. Gating and decoding are done on the subsystem boards and all intra-chassis cabling was eliminated.

3. Solid State Memories

Two different approaches were made in development of solid state memory systems. For the rack buffer, memory chips were purchased and mounted on thick-film substrates. The single unit development precluded any extensive test device. Consequently, off-line testing was limited and extensive testing required on line testing in the buffer using the computer. Diagnosis, repair and test turn-around were very slow and it was concluded that this fabrication was not appropriate for a development memory.

In the prototype buffer, dual in-line memory packages were used. Replacement of malfunctioning devices and general maintainability was greatly enhanced. Cycle time and reliability characteristics were superior on the DIP devices. It should also be pointed out that the state of the art for solid state memory devices substantially improved over the time interval between the two developments.

4. Error Detection and Correction

Block coding, majority voting, and simple parity techniques were implemented for error detection and correction. Each of these methods is readily implemented and may be effectively employed.

Simple parity provides effective protection of memory as these are generally high reliability equipments with minimal transmission line problems. When malfunctions do occur, they characteristically are single bit failures.

Majority voting provides the capacity to correct errors occuring prior to the voting operation. Multiple copies of the information is its primary disadvantage. Voting in the buffers prior to transmission provides correction for resident memory failure and protection against an improper command copy from one of the computers.

Block codes are very powerful methods for detecting data loss and consequently find application in transmission systems. Another advantage, often overlooked, is they provide capability for the using system to determine the usefulness of the information thus minimizing feed-back loop requirements.

In the rack buffer, a Hamming block code served the dual functions of protecting command words in memory as well as the transmission. This type of application has merit in a system like PCM to help protect against out-of-sync conditions by data loss in the address fields.

Implementation of any detection system requires decisions on how to handle the error once detected. The general philosophy on both buffer developments was to continue the operation and indicate the nature of the error by writing status into the change buffer.

The primary purpose of these developments was to determine the effectiveness and difficulty of implementation. In both buffers, multiple methods of error detection were implemented. Because of the difficulty of determining and making a decision on what to do when an error is detected, it is recommended that when possible only a single technique be selected that satisfies the nature and reliability requirement of the system.

SECTION II

CONTENTS

ILLUSTRATIONS

40

SUMMARY

This report discusses digital data communication techniques for a random access multi-station data bus system (see Figure A1-1). Several signal formats are considered in terms of the bandwidth required for proper transmission and the hardware required for implementation (see Part 3). The bandwidth required is an important consideration due to the low-pass characteristic of the channel used (124 ohm Balanced Line). The hardware must be kept simple due to the potentially large number of stations required.

The signal format selected for use on a data bus is based on the Modified Duobinary technique developed by Lender (11). The implementation developed uses standard T^2L logic elements operating off a single +5 volt power supply. The recommended transceiver design (Transmitter-Figure A7-1, Receiver-Figure A5-5) has been satisfactorily demonstrated at a 5Mbps data rate over 2000 feet of BL-1244 cable.

Operating systems are discussed in Part 2 and Part 7 and error control coding is discussed in Part 6.

Part 1 - Transmission Line Characteristics

The necessary care and feeding of transmission lines imposes several important constraints on the design of a data bus. The parameters which influence system design include the characteristic impedance (Z_o), the velocity of propagation (V_p), the maximum distance between stations (L), and the attenuation coefficient (a).

The attenuation coefficient is defined as follows. On a properly terminated transmission line, a sinusoid of peak amplitude V and frequency f is impressed at one end of the line, resulting in a sinusoid of peak amplitude V' and frequency f at the far end of the line. For a line x meters long,

$$V' = Ve^{-ax}$$

or

$$a = \ln (V/V')/x \quad \text{nepers/meter.}$$

The transmission line must be terminated in its characteristic impedance Z_o. Proper termination implies that any signal source which impresses a signal onto the line will "see" a load of Z_o ohms down all line segments to which the source is connected.

The system configuration dictated by constraint e) (see the Introduction) is shown in Figure A1-1. There are several methods of properly terminating the line.

If a matching pad (see Figure A1-2) were placed in the line at every station location, proper termination with respect to all stations would result. Note that a signal on the transmission line is attenuated by a factor of $(2 + \sqrt{3})^{-1}$ as it passes each pad (or each station). With no limit on the number of stations, this method of terminating is clearly unacceptable.

An alternate method of maintaining proper line termination is depicted in Figure A1-3. If the input impedance for the station receivers is large, they can continually monitor the line without affecting the line termination. If only one station transmitter is switched onto the line at any time, the line will appear as an impedance of $Z_o/2$ (two properly terminated line segments in parallel) to the transmitter. This method is more complex (involves transmitter switching) than the previous method, but does not involve additional signal attenuation at each station. The maximum signal attentuation would occur in a communication from station 1 to station N and is given by e^{-aL}.

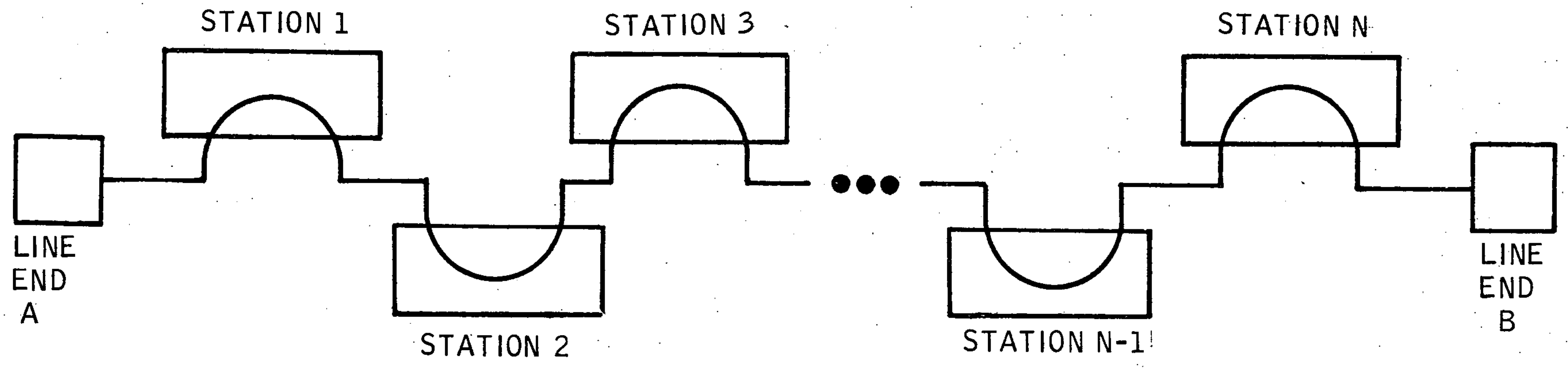
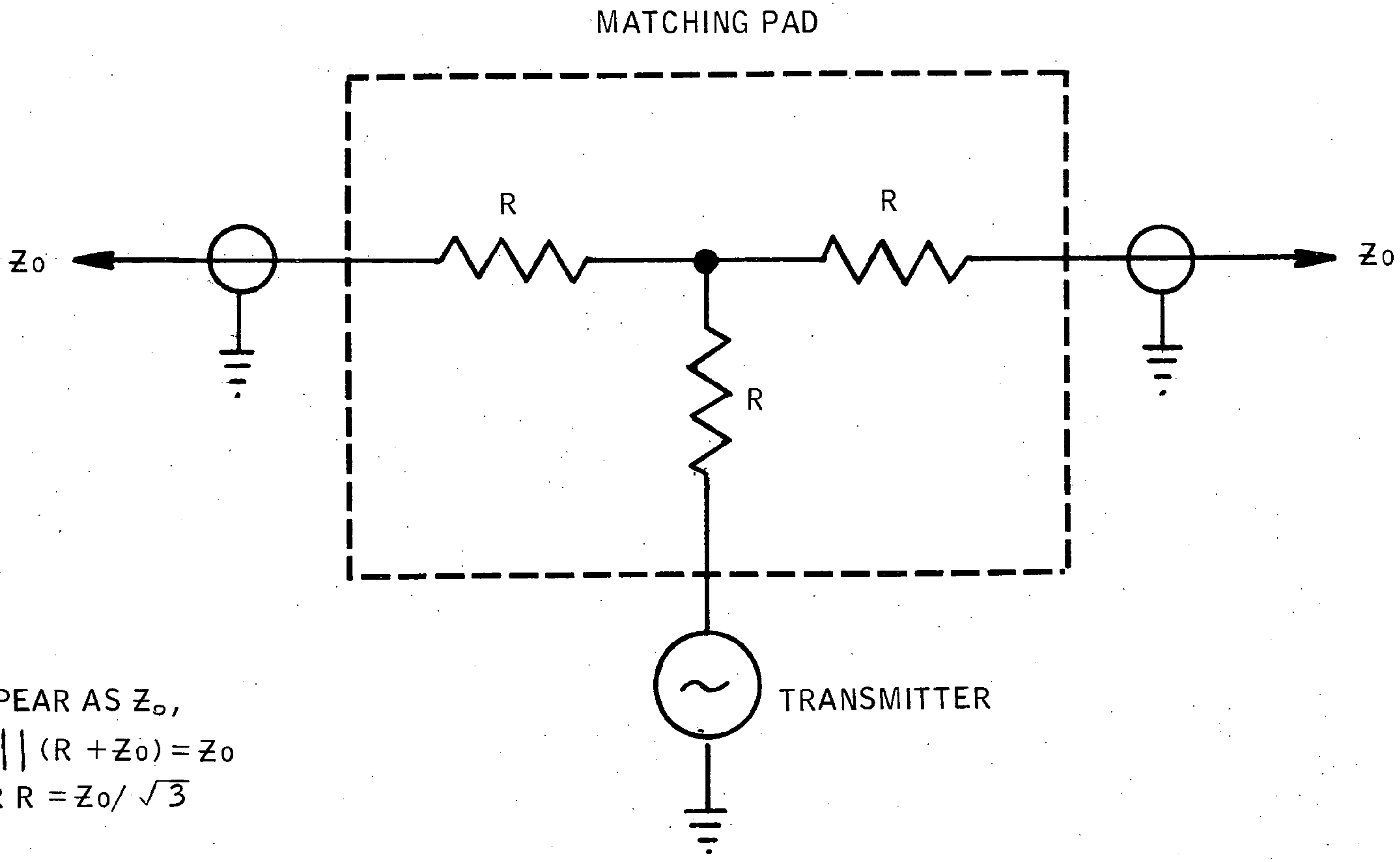


FIGURE A1-1

SYSTEM CONFIGURATION



TO APPEAR AS Z_o,
$R + R \,||\, (R + Z_o) = Z_o$
OR $R = Z_o/\sqrt{3}$

FIGURE A1-2

MATCHING NETWORK

The general expression for signal attenuation between two stations is

$$\frac{Vi}{Vj} = e^{-a\left|Xi - Xj\right|} \qquad (1)$$

where station i is receiving a transmission from station j, Vi and Vj are signal amplitudes, and Xi and Xj are defined in Figure A1-3.

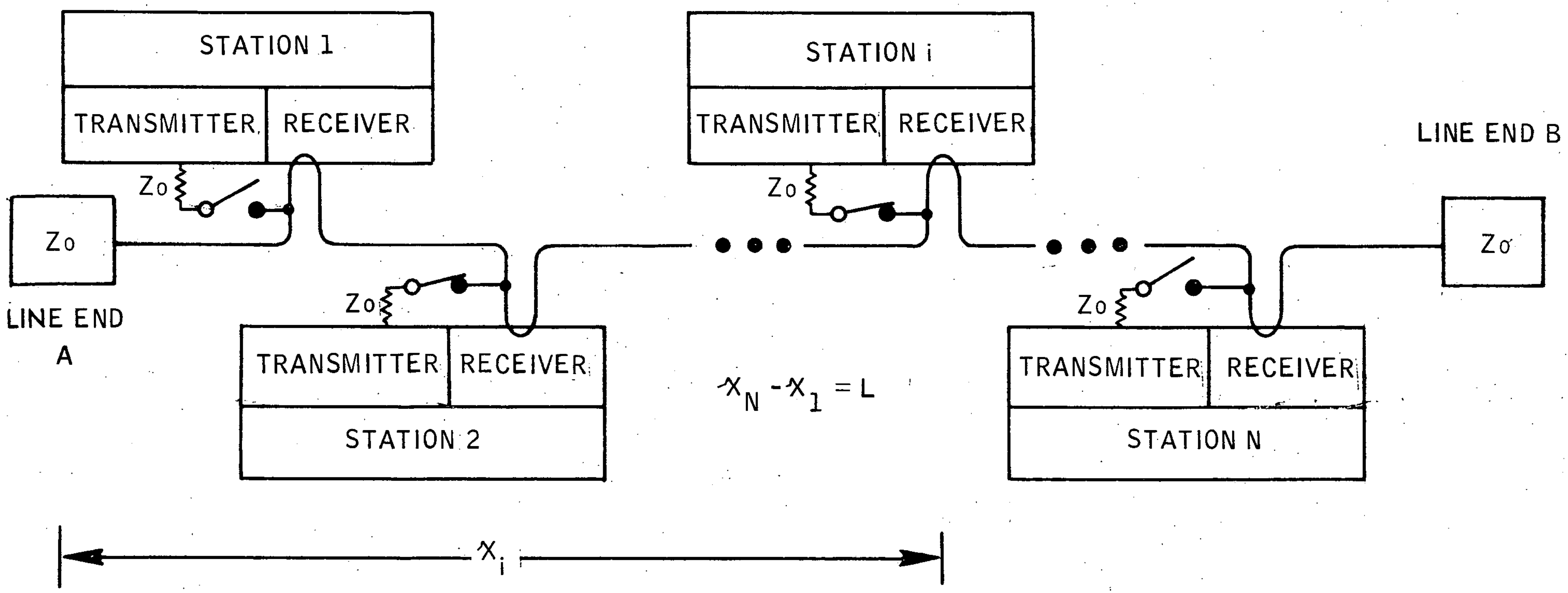
The method of line termination indicated in Figure A1-3 is compatible with the system constraints and will be used in the system design.

The velocity of propagation (Vp) implies a finite time delay for signal transmission between stations. The maximum delay is given by:

$$T_D = L/V_p$$

The general expression is given by:

$$t_d\ (i,j) = \left|Xi - Xj\right| /V_p. \qquad (2)$$



TRANSMITTER Z = Zo

RECEIVER Z ≫ Zo

FIGURE A1-3
SWITCHED TRANSMITTER SYSTEM

Part 2 - Random Bus Access

The requirement that no station be unique in terms of line access priority leads to some general restrictions on the characteristics of inter-station messages. The key problem is to allow any station to access the line in such a way that all other stations are prohibited from accessing the line concurrently. Additionally, the possibility of simultaneous line accession by several stations must be considered.

There are several alternate system designs which allow random bus access. Two approaches will be discussed here.

Approach A initiates a data transfer by transmitting the entire word, including transmitter and receiver addresses and data. The receiving unit, upon recognition of its address in the received word, immediately returns an "OK" to the transmitting unit. This can be nothing more than a short signal burst. The signal occurances are indicated in Figure A2-1 for a transmission over the entire bus length (L).

All non-participating stations automatically lock out their transmitters when a signal is present on the bus. This inhibit is maintained for $2T_D$ seconds after the bus is quiet to prohibit a station from talking while a data transfer is under way.

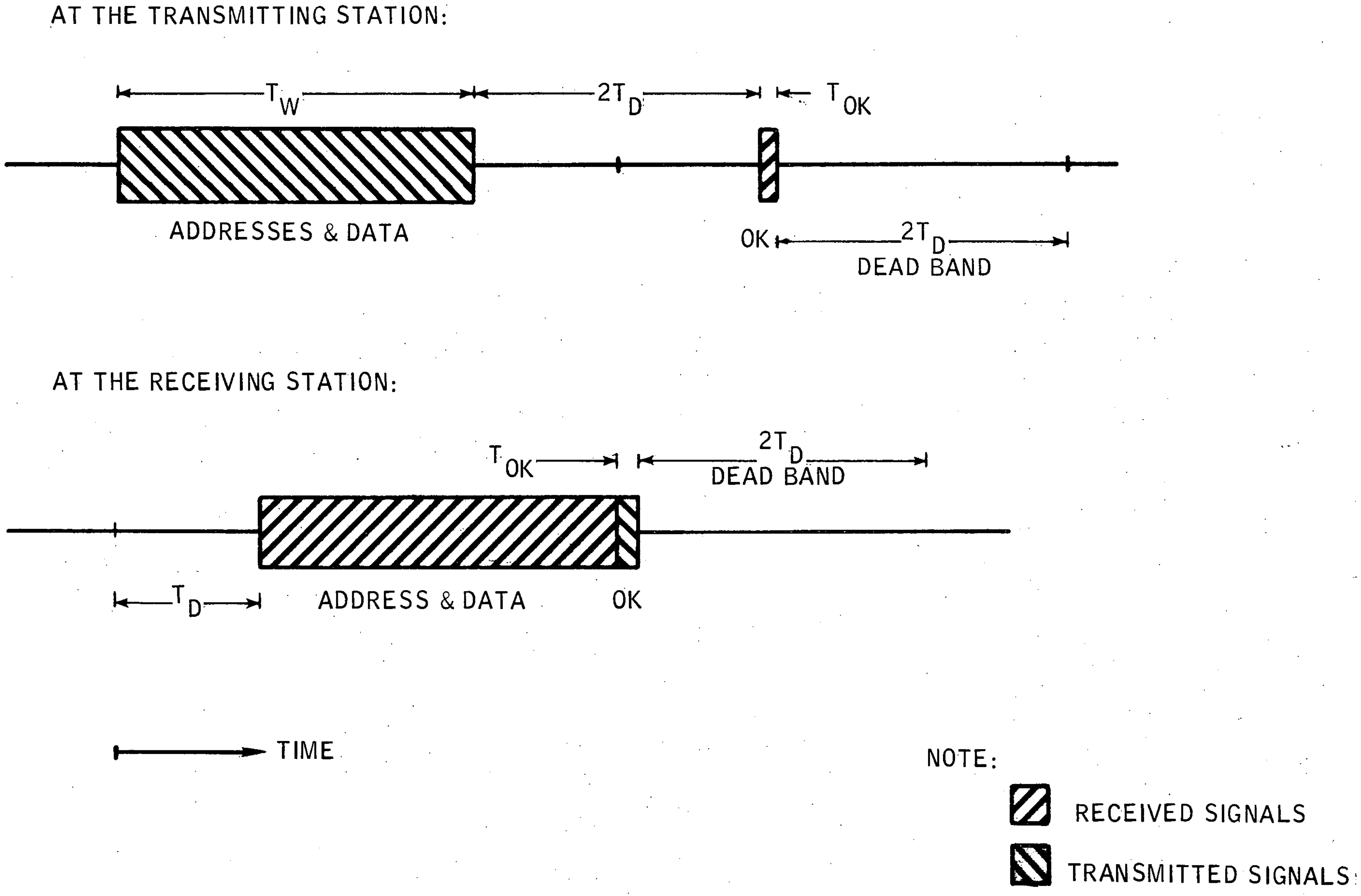


FIGURE A2-1 APPROACH A

All stations are to be electrically identical except for their addresses. This requirement implies that all built-in time delays are the same for all stations, so that data transfer over L meters must be allowed for in all units.

With the foregoing in mind, the worst case maximum word rate possible is:

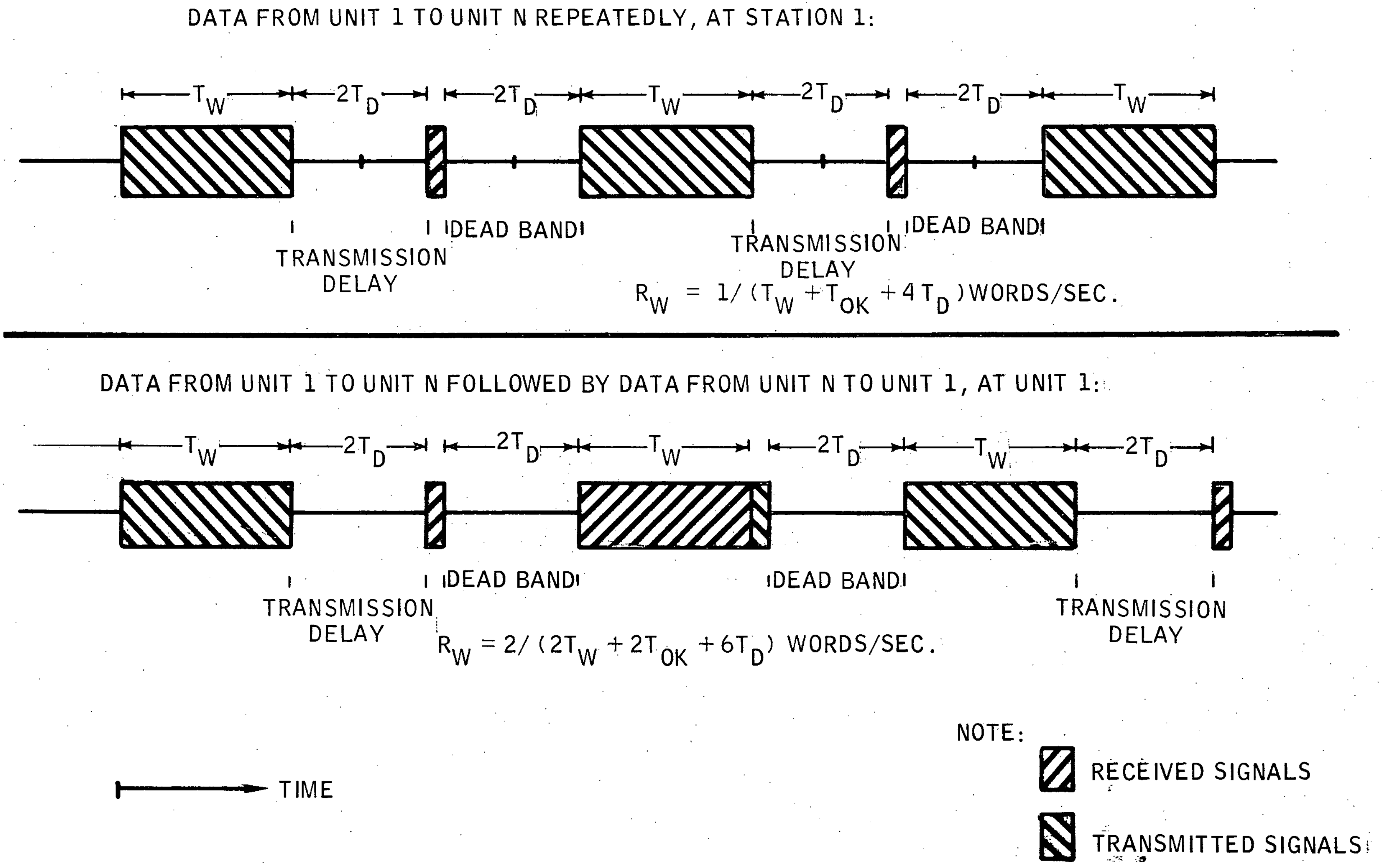
$$R_W = 1/(T_W + T_{OK} + 4T_D).$$

This rate would result when station 1 communicated with station N repeatedly. If station 1 communicated with station N, followed by station N sending data to station 1 etc., the rate would be:

$$R'_W = 1/(T_W + T_{OK} + 3T_D).$$

These two circumstances are indicated in Figure A2-2. (Note that in the second case, the dead band between a transmitted 'OK' and a transmitted word serves no purpose and could be eliminated. This is true because the last unit to transmit an 'OK' will also be the first unit to be enabled for transmission. This would result in a word rate $R''_W = 1/(T_W + T_{OK} + 2T_D)$ words/ sec.)

The dead band shows the effect of transmitter lock out as previously mentioned. All transmitters are locked out for a period of $2T_D$ seconds after the line goes quiet, regardless of the origin of the last signal on the bus. The exception to this is the transmitter used to return an


DATA FROM UNIT 1 TO UNIT N REPEATEDLY, AT STATION 1:

DATA FROM UNIT 1 TO UNIT N FOLLOWED BY DATA FROM UNIT N TO UNIT 1, AT UNIT 1:

FIGURE A2-2

APPROACH A TIMING

'OK' to an originating station. The reason for the dead band can be inferred from Figure A2-2. For the case depicted at the top of the figure, unit 1 is not allowed to transmit until the dead band following the reception of the 'OK' has elapsed. Other units, since they are located between unit N and unit 1, will be enabled before unit 1. This prevents unit 1 from monopolizing the data bus. The situation shown results only if all other units have no need for the data bus.

Transmissions between units closer than L meters would result in higher word rates since the transmission delay would be shortened. The dead band could not be shortened as it is designed into each unit. For units located adjacent to each other, the corresponding word rates are:

$$R_W = 1/(T_W + T_{OK} + 2T_D),$$

$$R'_W = 1/(T_W + T_{OK} + 2T_D),$$

and

$$R''_W = 1/T_W + T_{OK} + T_D).$$

Simultaneous line accession by two or more units is possible with this approach. To prevent erroneous data from being accepted when simultaneous accession has occurred, it is necessary to code the addresses and data in such a way that a receiver will ignore the bad data. Parity checks over the data coupled with an m out of n code ($n > m$) for the addresses should enable a receiver to detect the presence of erroneous transmissions. Such transmissions would be ignored and would not generate 'OK' bursts so that the transmitting stations could take appropriate action.

An alternate solution is as follows. During a transmission, the transmitting station continuously monitors the bus for the presence of additional signals. If an additional signal is detected, the transmitter immediately terminates the transmission. In this manner, if

$$T_W > 2T_D$$

the transmitted word is shortened. The intended receiver, failing to receive an entire word, would not respond.

This technique requires that the modulation used allow for line monitoring during a transmission.

For a 5Mbps data rate on 2000 feet of cable, a word length in excess of 30 bits satisfies the word length requirement.

The second technique (Approach B) for allowing random bus access initiates a data transfer by transmitting the address of the intended receiving station. This transmission serves to lock out the transmitters of all stations, except the originating station, for a predetermined interval. This interval (T_M) allows the originating unit to complete its transmission.

Following the transmission of the receiving unit's address, the originating

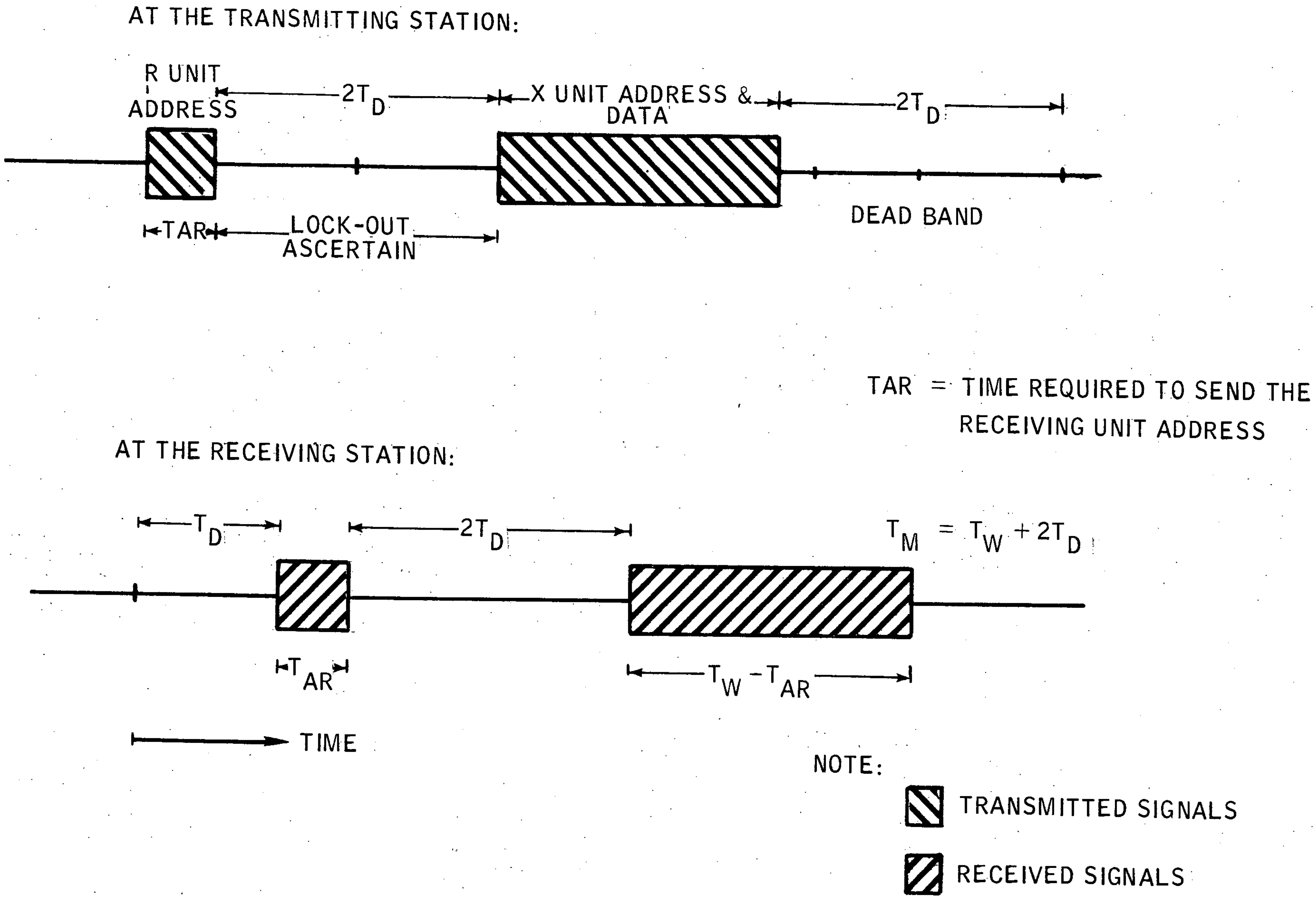


FIGURE A2-3
APPROACH B

station monitors the line for $2T_D$ seconds. Assuming no signal is detected on the bus during this interval (which assures that all remaining units are locked out), the originating unit address and the data are transmitted. This approach is shown in Figure A2-3 for a transmission over L meters.

The dead band shown is necessary only at the transmitting station. This prevents one unit from monopolizing the bus.

The worst case maximum word rate occurs when one unit repeatedly transmits and is given by:

$$R_W = 1/(T_W + 4T_D)$$

Figure A2-4 shows the bus signals which result when station 1 transmits to station N followed by station N responding to station 1 repeatedly. The word rate on the bus for this condition is

$$R'_W = 1/(T_W + 3T_D) \quad \text{words/sec}$$

For units adjacent to each other, the transmission delay indicated in Figure A2-4 would be negligible. In this case, the word rate for two alternately communicating units is

$$R'_W \doteq 1/(T_W + 2T_D) \quad \text{words/sec.}$$

The problem of multiple line access is solved as follows. If the originating unit detects an additional signal on the line at any time during the transmission of the receiving unit's address or the lock out ascertain interval, transmission would cease for a predetermined interval. The wait period would need to exceed $2T_d$ seconds.

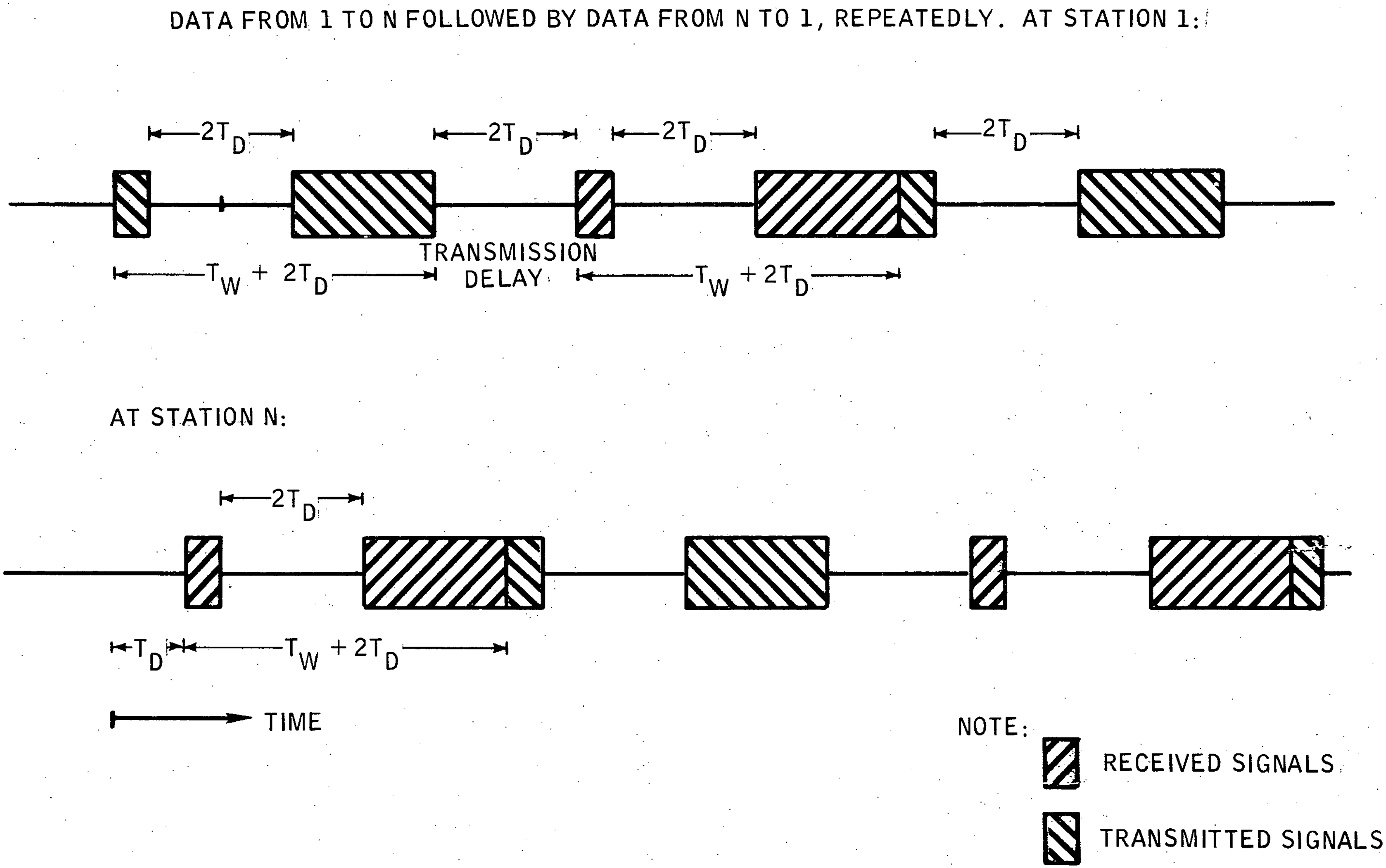


FIGURE A2-4
APPROACH B TIMING

The unit causing the interference would react identically. Due to the tolerances associated with the two delay circuits, one of the units would be enabled first and would proceed with its message.

The unit designated as the receiver would also detect the additional signal. This signal could occur before, during, or following the time when the designated receiver was receiving its address. The following data would then be ignored.

Part 3 - Signal Design

This section will consider the form of the signal to be used to convey data on the bus. The choice of a signal is influenced by the following considerations:

a) The channel to be used

b) Ease of implementation

c) The data rate

The channel in this case is the bus cable itself. A representative cable to use is Times Wire & Cable BL-1244. This is a balanced line designed for high isolation at moderate frequencies. The characteristics are as follows:

Times Wire & Cable BL-1244

Characteristic Impedance:	124 Ω
Outside Diameter :	.420"
Velocity of Propagation :	66.5% of C
Conductor Resistance :	9.4 Ω /Kft
Nominal Capacitance :	12.3 pfd/ft

Attenuation Characteristics:

db/100ft	Frequency (MHz)
.1	.1
.34	1.0
1.2	10.0
4.0	100.0

A reasonable approximation to this is

$$\frac{db}{100\,ft} = .37 \times 10^{-3} \sqrt{f}$$

The attenuation at higher frequencies is the cable characteristic which has the most effect on a data bus signal selection. This cable characteristic dictates the choice of signals having as little energy as possible at high frequencies. Signals based on modulation of a high frequency carrier are effectively ruled out. Carrier signals, with carriers close to the data rate will be considered however.

Before discussing various signals, it is useful to obtain an expression for the spectrum of the data. As a model for the data, a semirandom binary transmission system (corresponding to random NRZ data) is used.

The model is defined as shown in Figure A3-1 where $P\{\underset{\sim}{x}(t) = A\} = P\{\underset{\sim}{x}(t) = 0\}$ $= 1/2$ and transitions are only allowed at

$$t = mT, \quad m = \cdots, -2, -1, 0, +1, +2, \cdots$$

For this model,

$$\begin{aligned} E\{\underset{\sim}{x}(t)\} &= A \cdot P\{\underset{\sim}{x}(t) = 1\} + 0 \cdot P\{\underset{\sim}{x}(t) = 0\} \\ &= A/2 \end{aligned}$$

and

$$\begin{aligned} E\{\underset{\sim}{x}^2(t)\} &= A^2 \cdot P\{\underset{\sim}{x}(t) = 1\} + 0 \cdot 0 \cdot P\{\underset{\sim}{x}(t) = 0\} \\ &= A^2/2 \end{aligned}$$

The autocorrelation is

$$\begin{aligned} R(\tau) &= E\{\underset{\sim}{x}(t+\tau)\,\underset{\sim}{x}^*(t)\} \\ &= A^2 \cdot P\{\underset{\sim}{x}(t+\tau) = A, \underset{\sim}{x}(t) = A\} + 0 \end{aligned}$$

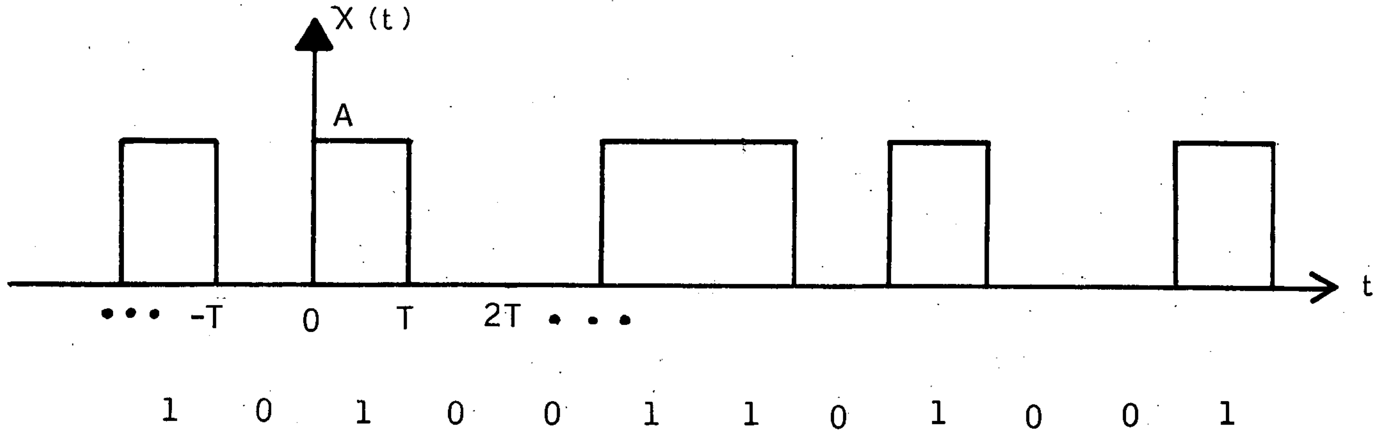


FIGURE A3-1
NRZ DATA

For

$$\tau = 0,$$

$$R(0) = E\{\underset{\sim}{x}^2(t)\} = A^2/2$$

For

$$|\tau| \geq T$$

$$R(\tau) = A^2 P\{\underset{\sim}{x}(t+\tau)=1, \underset{\sim}{x}(t)=1\}$$

$$= (1/2)(1/2)A^2 = A^2/4$$

For

$$0 \leq |\tau| \leq T$$

$$R(\tau) = \{1/2 \quad - (1/4)|\tau|/T\}A^2$$

The power spectrum of the random process $\underset{\sim}{x}(t)$ can now be derived as follows:

$$S(\omega) = \int_{-\infty}^{\infty} R(\tau)e^{-j\omega\tau}\,d\tau$$

$$= \int_{-\infty}^{\infty} \frac{A^2}{4} e^{-j\omega\tau}\,d\tau + \frac{A^2}{4}\int_{-\infty}^{\infty} (1-|\tau|/T)e^{-j\omega\tau}\,d\tau$$

Or

$$S(f) = \frac{A^2\pi}{2}\,\delta(f) + (A^2/4)T\,\frac{\sin^2 \pi f T}{(\pi f T)^2}.$$

Two important facts can be determined from S(W):

1) NRZ data contains energy down to and including zero frequency (DC).

2) Most of the energy is contained in the frequency band below 1/T Hz .

The spectrum for NRZ data is shown in Figure A3-2.

The question arises as to why not simply transmit the NRZ data on the bus? With the above facts, an answer is possible:

> The bus will probably be transformer coupled to provide a maximum of noise immunity for the stations tied to the bus. Even if bipolar NRZ were used (which removes the δ(f) from S(f)), the NRZ signal still requires a transmission medium with response down to and including DC. Since transformers exclude DC, the bus waveform would suffer from "DC wander" or "Baseline gallop". DC wander makes the signal difficult to reconstruct at the receiver.

The power spectrum for amplitude modulation is easily derived with the knowledge of S(W) for NRZ data. The spectrum of a sinusoidal carrier at fc is simply $\pi\,\delta(f - fc)$. The resultant spectrum is therefore simply S(W) shifted by fc Hz. The situation for fc = 1/T is shown in Figure A3-3. The spectrum contains a carrier component at fc(=1/T) and extends from 0 Hz to 2/T Hz.

The line signal for this modulation method is shown in Figure A3-4. The DC component has been removed by modulation and the bandwidth has been doubled with respect to NRZ.

If bipolar (± B volts) NRZ is used to amplitude modulate a sinusoidal carrier

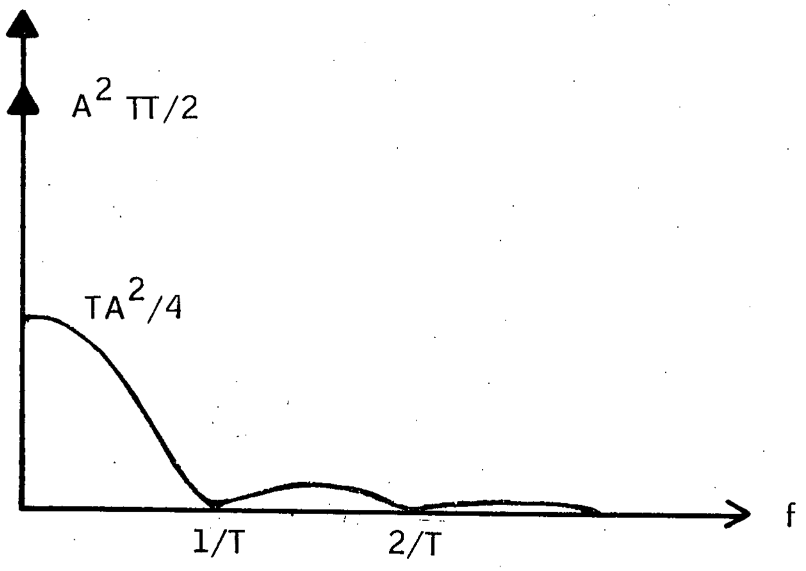


FIGURE A3-2
NRZ DATA SPECTRUM

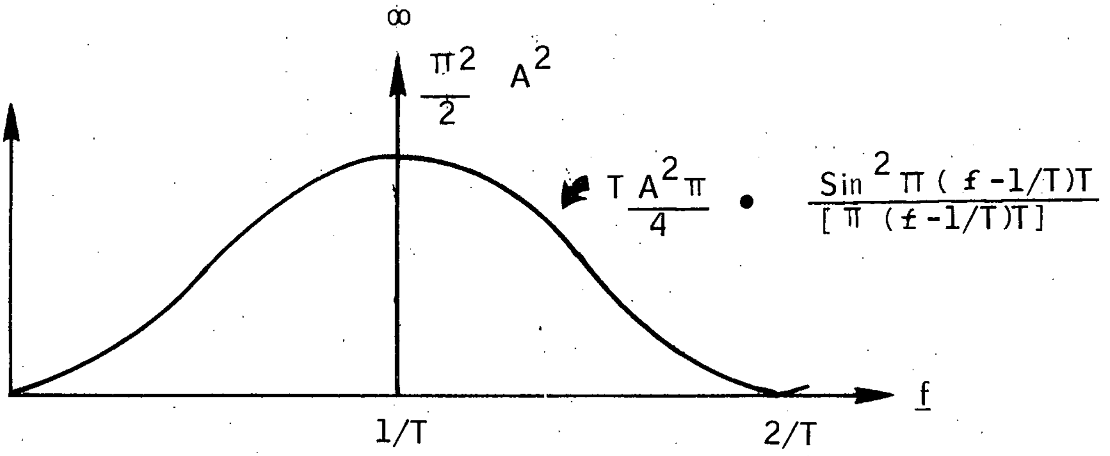


FIGURE A3-3
SPECTRUM OF ASK SIGNAL

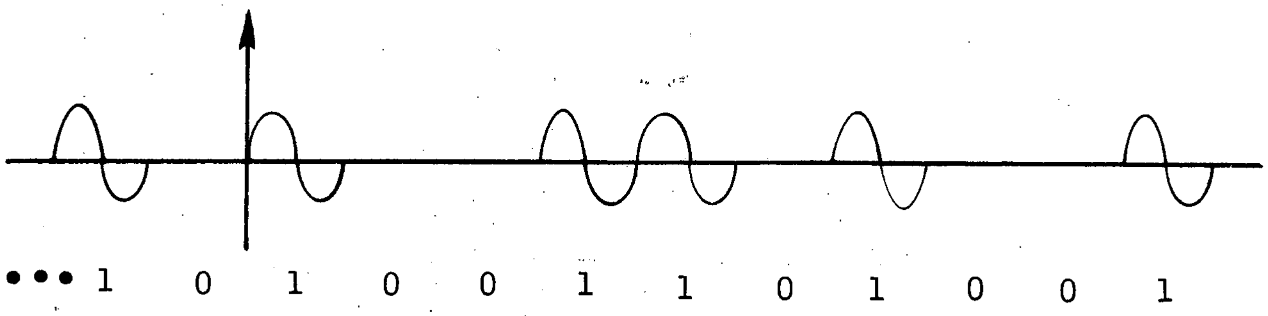


FIGURE A3-4
ASK LINE SIGNAL

the spectrum shown in Figure A3-5 results. The line signal is shown in Figure A3-6. Again the DC component is removed and the bandwidth (with respect to NRZ) is doubled. Note that this is exactly the same signal as split-phase C or binary PSK.

Frequency shift keying can be viewed as a combination of two amplitude modulated signals with different carrier frequencies. The situation for $fc_1 = 1/T$ and $fc_0 = 2/T$ is depicted in Figure A3-7. Each of the two component waveforms has a spectrum similar to Figure A3-3 with center frequencies of 1/T and 2/T. This follows from 1's and 0's being equally likely (recall the model used for NRZ data). The fact that the Fourier transform is a linear operation, i.e.

$$F\{x(t) + y(t)\} = F\{x(t)\} + F\{y(t)\}$$

allows us to immediately arrive at a power spectrum for the FSK signal as shown in Figure A3-8. For clarity, this figure only shows the major lobes of each individual spectrum.

The three preceeding examples cover the three well-known carrier modulation methods. Of the three, on the basis of spectrum occupation of the signals alone, FSK is seen to be the least desir able. On-off keying and split-phase C occupy the same range of frequencies and so offer no reason to select one over the other on this basis. It is well known that split-phase C has several implementation advantages (such as ease of clock recovery) and will therefore be selected as the "best" of the above three to hold up as a model for further comparisons.

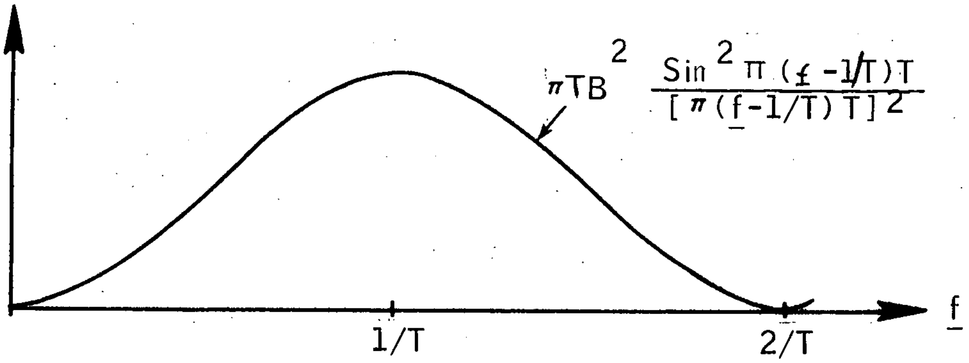


FIGURE A3-5
BIPOLAR NRZ ASK SPECTRUM

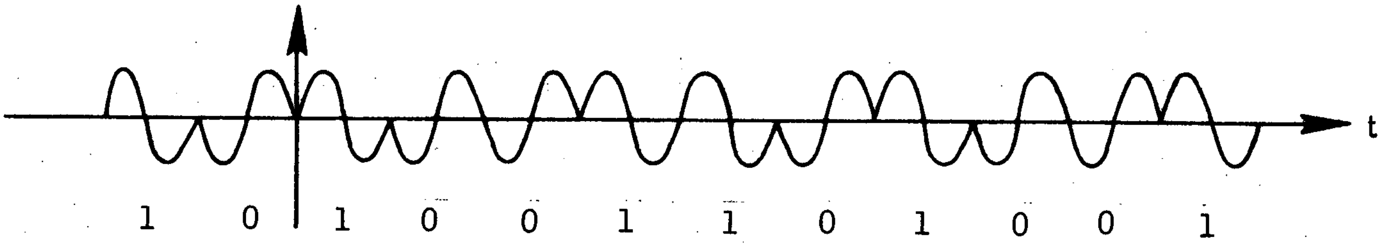


FIGURE A3-6
BIPOLAR NRZ ASK LINE SIGNAL

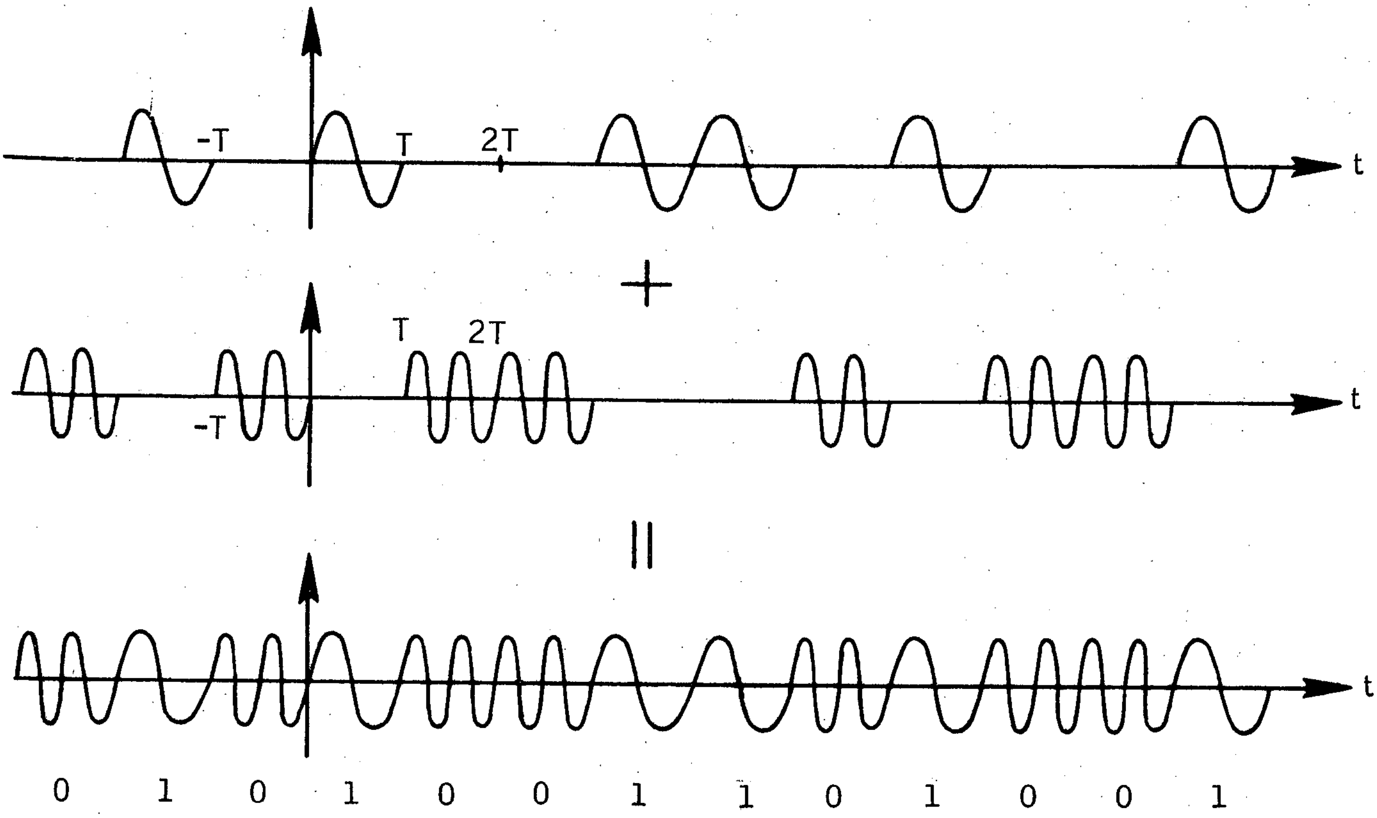


FIGURE A3-7
FSK LINE SIGNAL

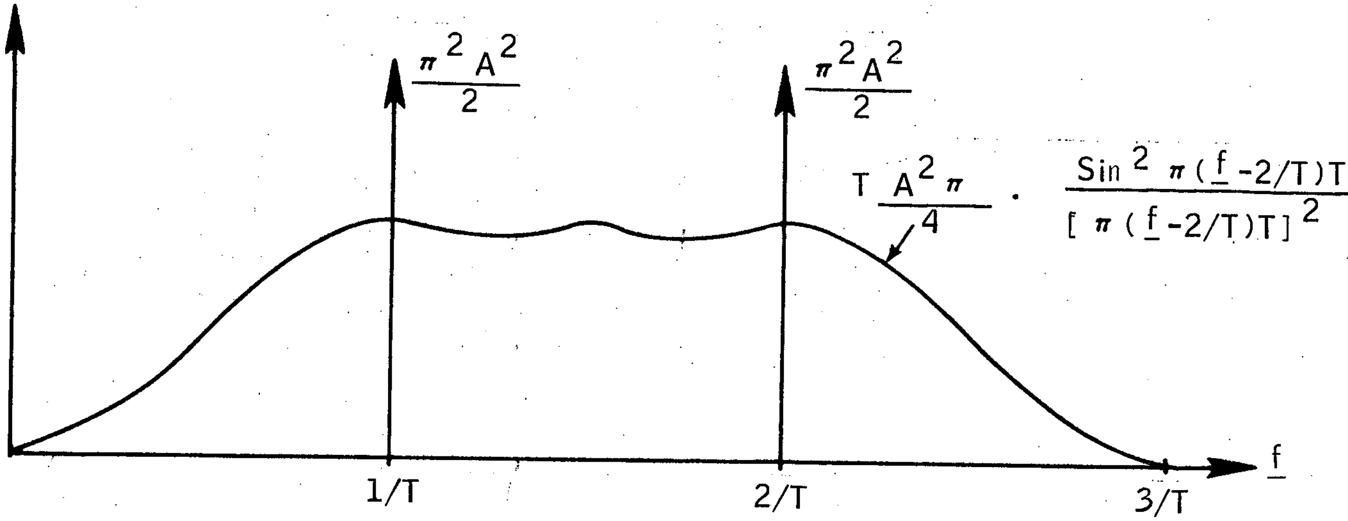


FIGURE A3-8
FSK SPECTRUM

The above results can be easily generalized to the case of multi-level signalling. Assume that there are $2^m \ (m = 1, 2, 3, \cdots)$ equiprobable levels. Each of the 2^m signals represents m binary bits. If 1/T is the bit rate, the signalling rate for multi-level signals is 1/mT levels/second. The data model has transitions at

$$t = n(mT), \quad n = \cdots, -2, -1, 0, +1, +2, \cdots$$

If C volts represents the highest level and 0 volts the lowest, the allowed levels are

$$0,\ C/(2^m-1),\ 2C/(2^m-1),\ \cdots,\ (2^m-1)C/(2^m-1).$$

For this model,

$$E\{\underset{\sim}{x}(t)\} = \sum_{i=0}^{2^m-1} \frac{iC}{(2^m-1)} P\{\underset{\sim}{x}(t) = iC/(2^m-1)\}$$

$$= C/2$$

$$\left[P\{\underset{\sim}{x}(t) = iC/(2^m-1)\} = 1/2^m\right]$$

$$E\{\underset{\sim}{x}^2(t)\} = \sum_{i=0}^{2^m-1} \left[\frac{iC}{(2^m-1)}\right]^2 (1/2^m)$$

$$= \frac{C^2}{2^m(2^m-1)^2} \sum_{i=0}^{2^m-1} i^2$$

$$= \frac{C^2(2^{m+1}-1)}{6(2^m-1)}.$$

The autocorrelation at $\tau = 0$ is:

$$R(0) = E\left\{ \underset{\sim}{x}^2(t) \right\} = \frac{c^2\,(2^{m+1}-1)}{6\,(2^m-1)}$$

At $|\tau| = mT$,

$$R(m\tau) = P \left\{ \sum_{i=0}^{2^m-1} \frac{ic}{(2^m-1)} \sum_{j=0}^{2^m-1} \frac{jc}{(2^m-1)} \right\}$$

where

$$P = 1/2^{2m}$$

This reduces as follows:

$$R(m\tau) = \frac{Pc^2}{(2^m-1)^2} \sum_i i \sum_j j$$

$$= \frac{Pc^2}{(2^m-1)^2} \sum_i i \left\{ \frac{2^m}{2}\,(2^m-1) \right\}$$

$$= \frac{Pc^2\,2^{2m}}{4} = c^2/4$$

The autocorrelation is shown in Figure A3-9. The spectrum is obtained as before:

$$S(\omega) = \int_{-\infty}^{\infty} R(\tau)\, e^{-j\omega\tau}\, d\tau$$

$$S(\omega) = \int_{-\infty}^{\infty} \frac{C^2}{4} e^{-j\omega\tau} \, d\tau$$

$$+ \left\{ \frac{C^2(2^{m+1}-1)}{6(2^m-1)} - \frac{C^2}{4} \right\} \int_{-mT}^{mT} \left(1-\frac{|\tau|}{T}\right) e^{-j\omega\tau} \, d\tau$$

So that

$$S(f) = \frac{\pi C^2}{2} \delta(f) + \left\{ \frac{C^2(2^{m+1}-1)}{6(2^m-1)} - \frac{C^2}{4} \right\} mT \, \frac{\sin^2 \pi f mT}{(\pi f mT)^2} \, .$$

This is illustrated in Figure A3-10.

The baseband power spectrum can be used to determine the spectrumof the various modulated signals previosly discussed. An example would be multilevel AM. Consider a carrier at fc = 1/mT modulated with the previously described multilevel data. The resulting spectrum is illustrated in Figure A3-11. This spectrum can be compressed as much as desired by increasing m. The line signal for m = 2 is depicted in Figure A3-12.

Shifting the DC level of the data by -C/2 results in the spectrum and line signal (for m = 2) shown in Figure A3-13 and A3-14. This is an interesting combination of amplitude and phase modulation. For m = 2, the waveform contains no DC component and occupies half the bandwidth of the split-phase C signal.

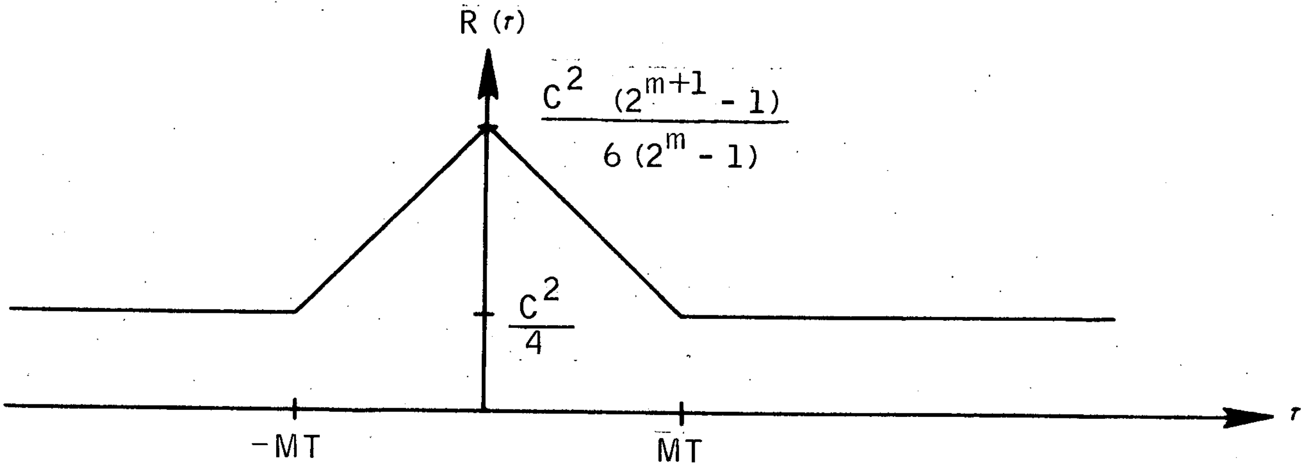


FIGURE A3-9
ANTOCORRELATION FOR MULTI-LEVEL SIGNALLING

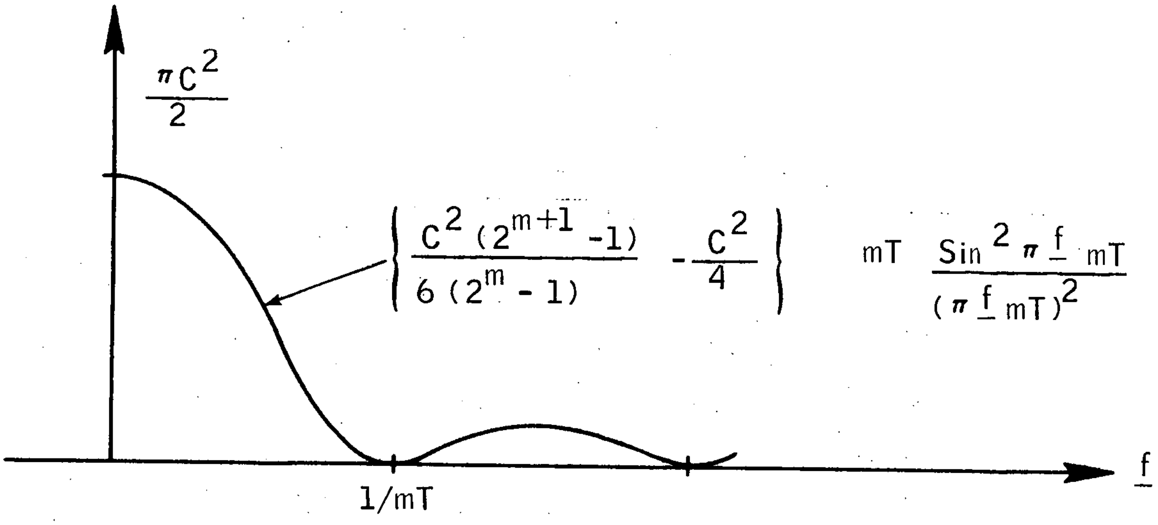


FIGURE A3-10
SPECTRUM OF MULTI-LEVEL SIGNALLING

73

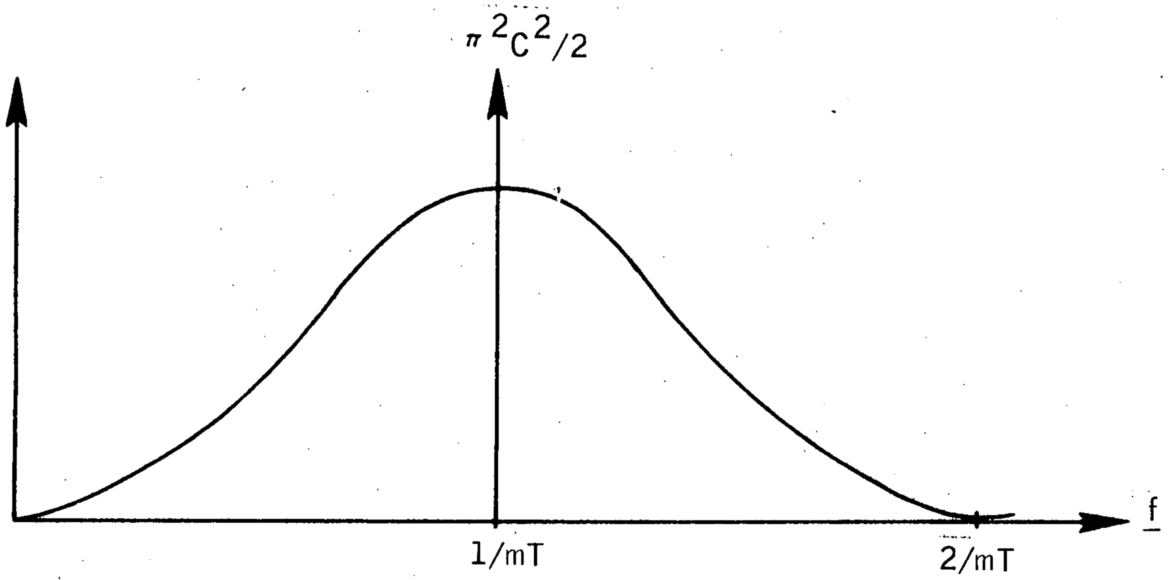


FIGURE A3-11
MULTI-LEVEL ASK SPECTRUM

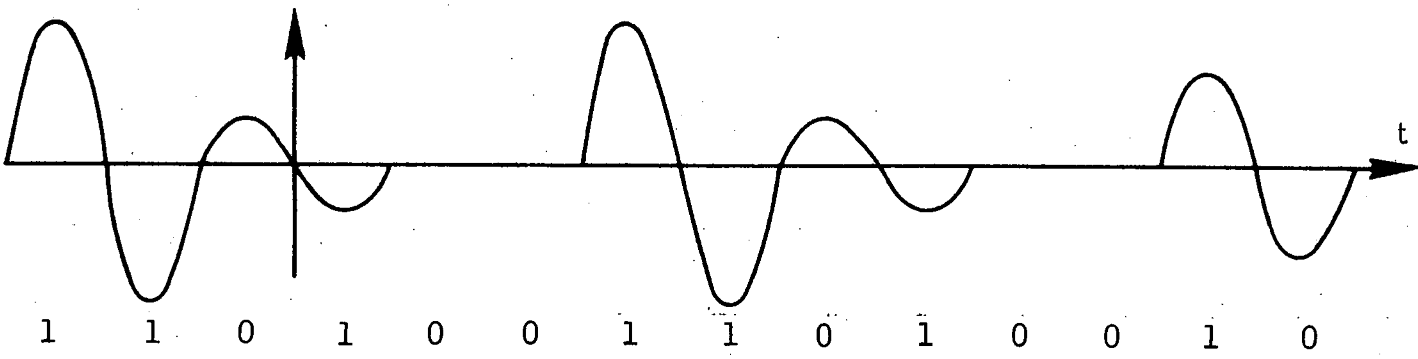


FIGURE A3-12
MULTI-LEVEL ASK LINE SIGNAL FOR m = 2

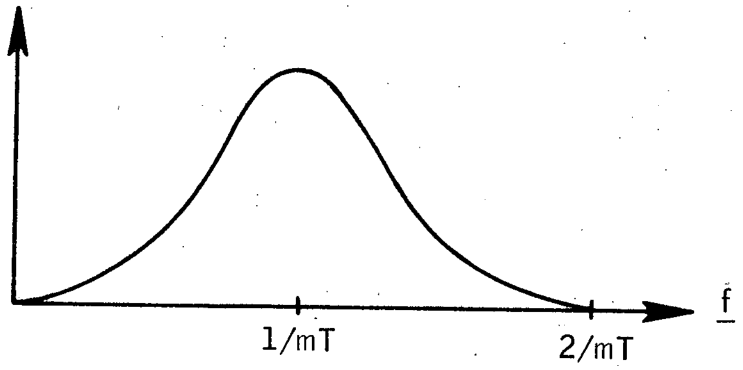


FIGURE A3-13
MULTI-LEVEL ASK SPECTRUM WITH NO CARRIER COMPONENT

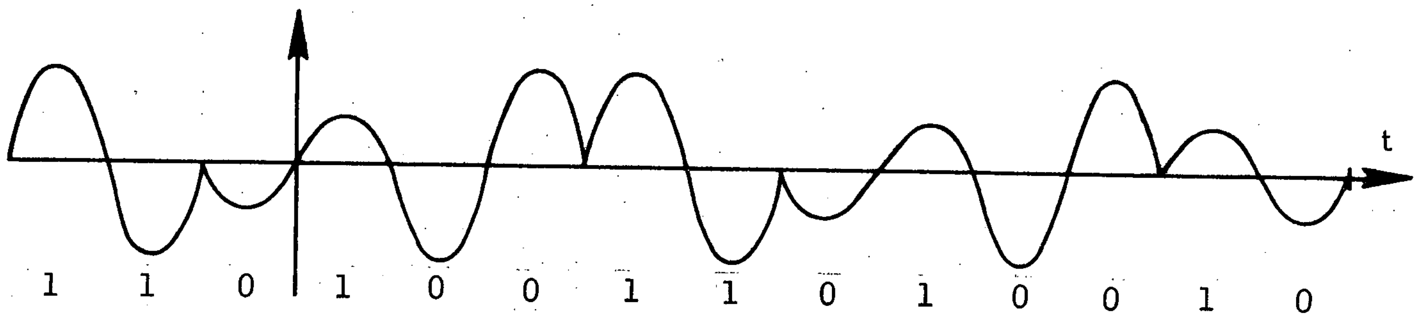


FIGURE A3-14
MULTI-LEVEL ASK LINE SIGNAL WITH NO CARRIER COMPONENT, m = 2

One additional example will serve to complete this discussion. The signal illustrated in Figure A3-15 can be constructed by the steps shown in Figure A3-16. The original data stream (at 1/T bps) is represented in a). $\underset{\sim}{x}^+$ and $\underset{\sim}{x}^-$ result from taking the odd and even bits from a) and halving the data rate to 1/2T bps. It is easy to see that the waveform of Figure A3-15 can be expressed as:

$$f(t) = -\left[1 + \frac{\underset{\sim}{x}^+}{2}(A-1) + \frac{\underset{\sim}{x}^-}{2}(A-1)\right] \cos\frac{2\pi t}{T} + \frac{(A-1)}{2}\left(\underset{\sim}{x}^+ - \underset{\sim}{x}^-\right).$$

(NOTE: a shift in the time axis was made to simplify the expression).

$\underset{\sim}{x}^+$ and $\underset{\sim}{x}^-$ have identical spectra for random NRZ data. The first term of f (t) results in a spectrum as shown in Figure A3-17. The baseband spectra of $\underset{\sim}{x}^+$ and $\underset{\sim}{x}^-$ cancel, <u>if</u> the average value of each is equal. This will occur for very large, randomly selected data words. In general, for data words used on the bus the baseband spectra will <u>not</u> cancel and a DC component results. This is a serious flaw in the signal if it is used on an AC coupled bus.

By accepting a wider bandwidth, a signal with no DC component and with the 100% redundant transmission feature of the previous signal can be constructed as indicated in Figure A3-18. The signal is constructed by simply mulltiplying the DC shifted baseband process (at 1/T bps) as depicted in b) by a carrier at 1/T Hertz.

The preceeding modulation methods have at least one common feature: for a data rate of 1/T bps, the occupied bandwidth is at least 1/T Hz. The only

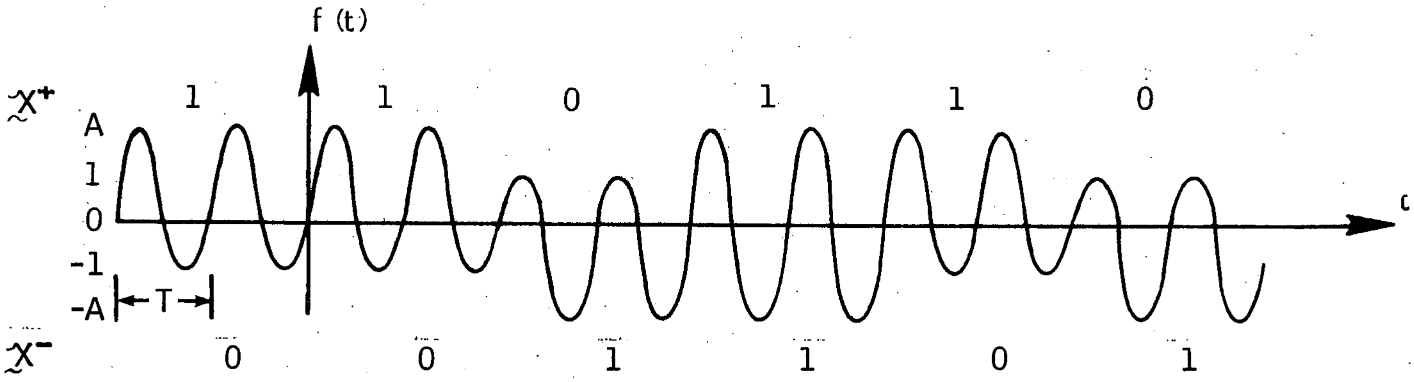


FIGURE A3-15
"MIL-PCM" LINE SIGNAL

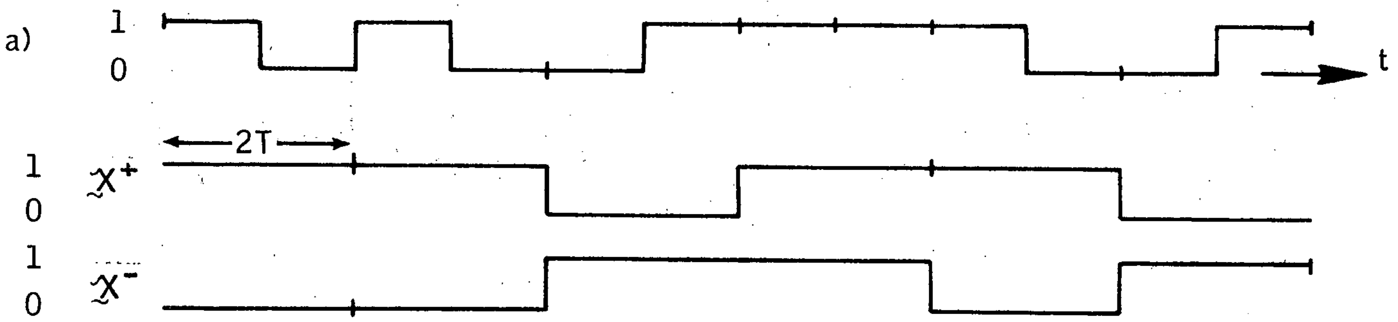


FIGURE A3-16
STEPS IN CONSTRUCTING "MIL-PCM" SIGNALS

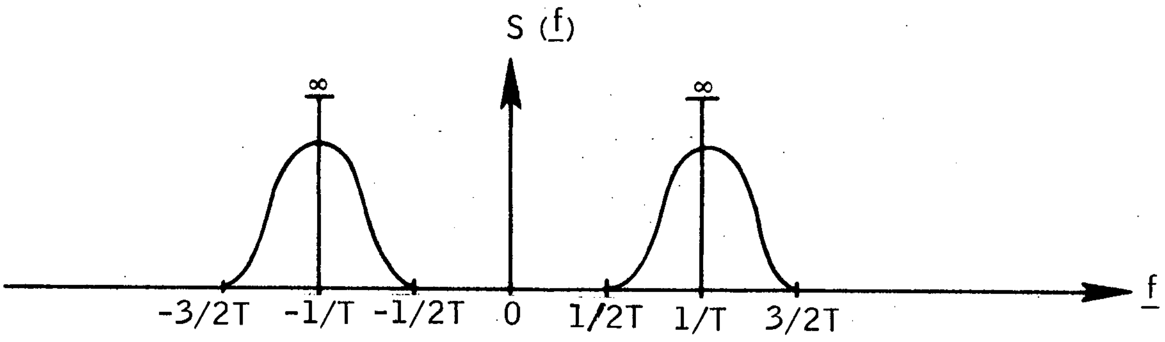


FIGURE A3-17
"MIL-PCM" SPECTRUM

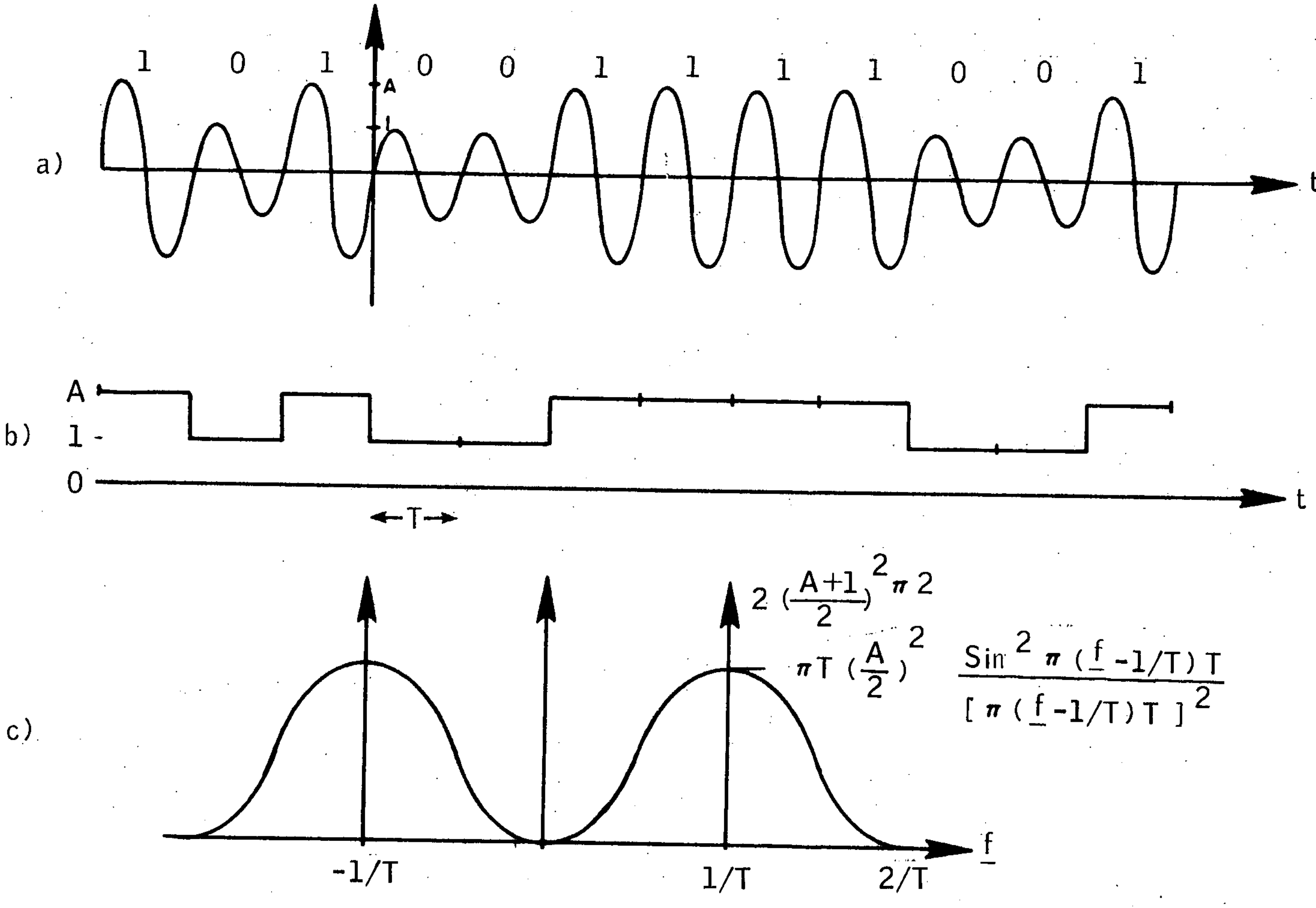

1 0 1 0 0 1 1 1 1 0 0 1
A
1
a)
t
A
b) 1 -
0
t
←T→
$2\left(\frac{A+1}{2}\right)^2 \pi 2$
$\pi T \left(\frac{A}{2}\right)^2$
$\frac{\mathrm{Sin}^2 \pi (\underline{f} - 1/T) T}{[\pi (\underline{f} - 1/T) T]^2}$
c)
-1/T
1/T
2/T
f

FIGURE A3-18
MODIFIED "MIL-PCM" SIGNAL AND SPECTRUM

exception is the case of multilevel signals. This technique leads to extremely complex receiver designs and so will not be considered.

The most important characteristic of the bus is its attenuation at higher frequencies. Because of this, it is worthwhile to investigate any method which promises lower bandwidths if the receiver design does not become unduly complex.

Consider a system in which impulses are used to transmit the data (baseband system). Obviously the bandwidth of such an idealized system would be infinite. If the impulse is first passed through a filter, the filter then determines the signal's bandwidth. The resulting line signal is therefore the filter's impulse response. These considerations are illustrated in Figure A3-19.

As a first example, consider an ideal low pass filter (LPF) with H(jw) as follows:

$$H(j\omega) = \begin{cases} K e^{-j\omega t_0} & |\omega| < \omega_1 \\ 0 & \text{elsewhere} \end{cases}$$

The impulse response to such a system is

$$g(t) = \frac{K\omega_1}{\pi} \frac{\sin \omega_1 (t - t_0)}{\omega_1 (t - t_0)}$$

as shown in Figure A3-20. This response indicates that impulses could be introduced at the rate of one per $\frac{\pi}{\omega_1}$ seconds and the resulting waveform, sampled at $t' = 0, \pm \frac{n\pi}{\omega_1}$, would allow independent determination of the amplitudes of the component impulses. From such considerations it is easily seen that, for an ideal LPF, the signalling rate cannot exceed ω_1/π $(= 2f_1)$ bps unless intersymbol interference can be tolerated.

The limitation is more severe than indicated. The ideal LPF can not be built and even if a suitable approximation could be implemented, the timing of the sampling at the receiver (as well as the original impulse timing) must be unrealistically accurate if severe intersymbol interference is to be avoided.

An example of a realizable filter function is the "raised cosine" filter described by:

$$H(j\omega) = \begin{cases} \frac{1}{2}\left[1 + \cos \pi\omega/\omega_1\right] e^{-j\omega t_0} & |\omega| \leq \omega_1 \\ 0 & \text{elsewhere} \end{cases}$$

for which

$$g(t) = \frac{\omega_1}{2\pi} \cdot \frac{\sin \frac{\omega_1}{2}(t-t_0)}{\frac{\omega_1}{2}(t-t_0)} \cdot \frac{\cos \frac{\omega_1}{2}(t-t_0)}{\left\{1 - \left[\frac{\omega_1}{\pi}(t-t_0)\right]^2\right\}}$$

which is illustrated in Figure A3-21. This impulse response has several interesting features. Most important: the tails fall off rapidly so that timing becomes much less critical than was the case for the ideal LPF. Also

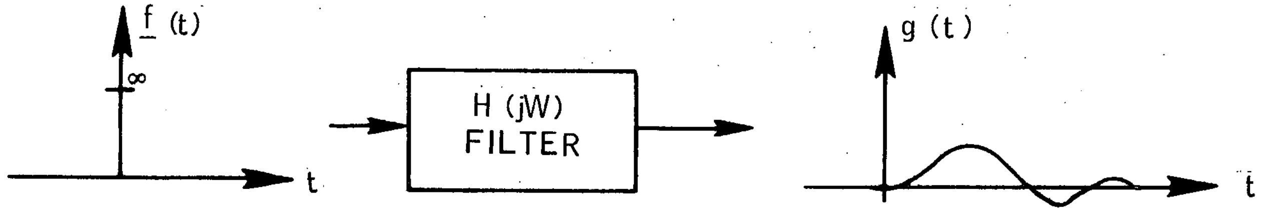


FIGURE A3-19
GENERALIZED BASEBAND SYSTEM

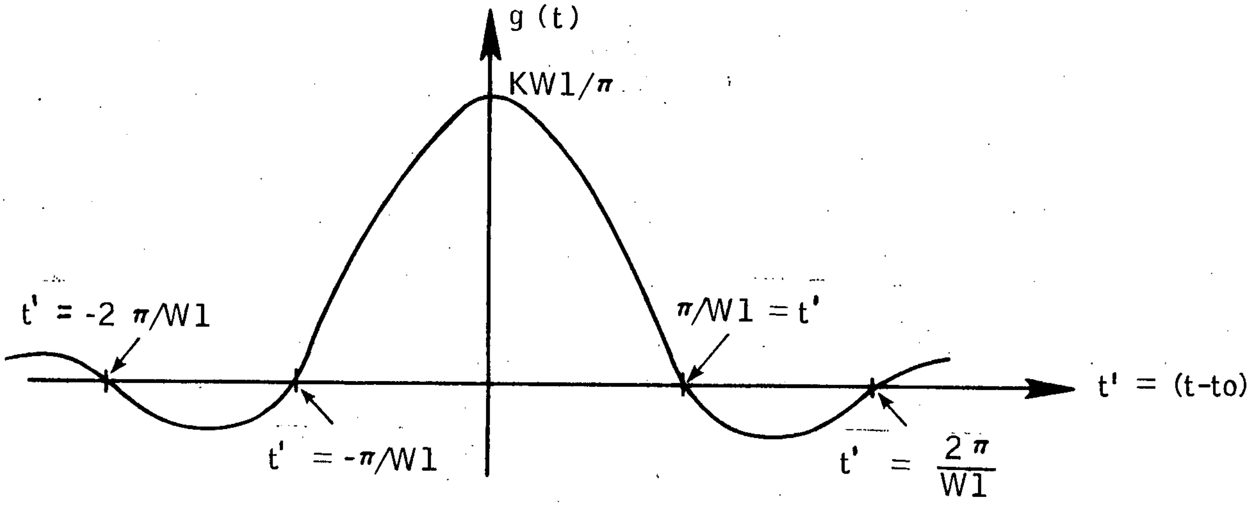


FIGURE A3-20
IMPULSE RESPONSE OF AN IDEAL LOW PASS FILTER

the filter function is easier to implement.

The response of Figure A3-21 indicates that impulses can be sent at a rate of $1/(2\pi/\omega_1)$ bps or f_1 bps. We have therefore sacraficed bandwidth for realizability, probably not a bad trade.

The response of Figure A3-21 is important because of an additional consideration: consider what happens if impulses are sent at the rate of $2f_1$ bps (the Nyquist rate). It can be seen that there will be intersymbol interference, but only among a finite number of digits. That is, if $t' = 0$ is the sampling instant,

$$x_0 = x\left(\frac{n\pi}{\omega_1}\right) = \begin{cases} 2 & n = 0 \\ 1 & n = \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

so that the intersymbol interference can only come from ajacent digits. If a positive impulse is used to represent a '1' and a negative impulse represents a '0', the five level signal illustrated in Figure A3-22 results.

To decode a signal such as that in Figure A3-22, the following equation must be solved:

$$x_{i-1} = a_{i-2} + 2a_{i-1} + a_i$$

or

$$a_i = x_{i-1} - a_{i-2} - 2a_{i-1}$$

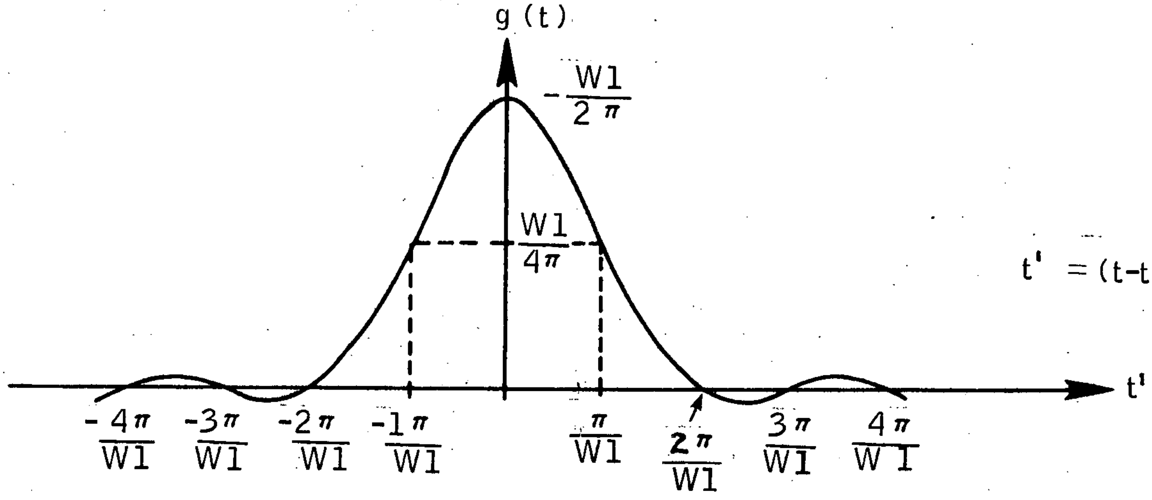


FIGURE A3-21
IMPULSE RESPONSE OF A RAISED COSINE FILTER

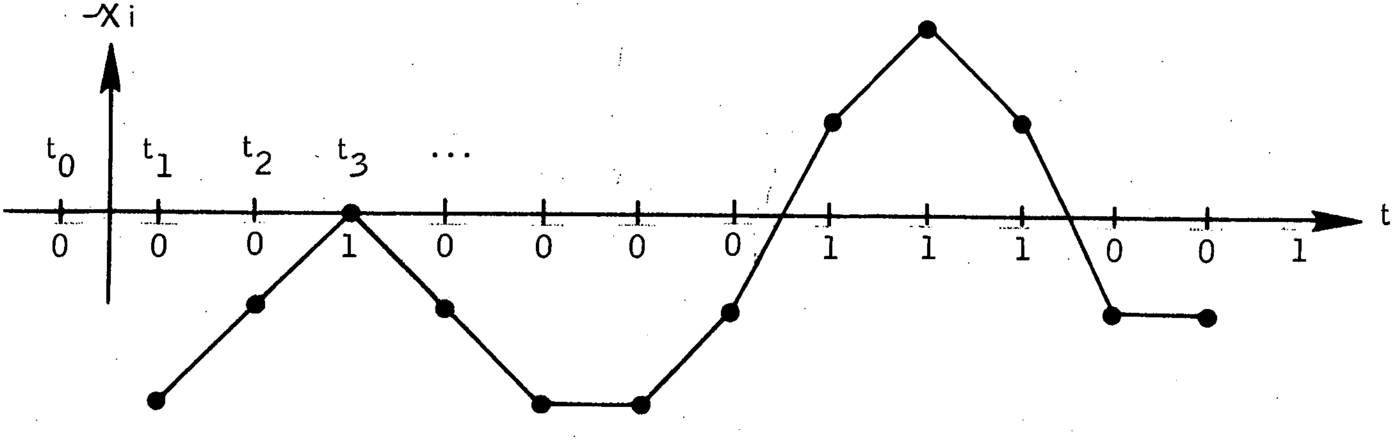


FIGURE A3-22
FIVE LEVEL LINE SIGNAL FROM A RAISED COSINE
FILTER OPERATED AT $2f_1$ bps

where x_i is the signal amplitude at t_i (the sampling instants) and

$$a_i = \begin{cases} +1 & \text{if ith bit} = 1 \\ -1 & \text{if ith bit} = 0. \end{cases}$$

Assuming that a_0 and a_1 are known, the signal of Figure A3-22 is decoded as follows:

$a_2 = (-4) - (-1) - 2(-1) = -1$

$a_3 = (-2) - (-1) - 2(-1) = +1$

$a_4 = (0) - (-1) - 2(+1) = -1$

$a_5 = (-2) - (+1) - 2(-1) = -1$

$a_6 = (-4) - (-1) - 2(-1) = -1$

etc.

The above example is a member of the "partial response" or controlled intersymbol interference class of modulation systems. The particular example above has two drawbacks: 1) it is possible to obtain a DC component in the line signal if the number of "ones" differs from the number of "zeros", and 2) a five level receiver is complex. The system does achieve Nyquist rate transmission however.

A more useful member of the "partial response" class is called modified duobinary. The filter function for MD is shown in Figure A3-23. The resulting impulse response is illustrated in Figure A3-24 and is given by

$$g(t) = \frac{1}{2\pi} \int_{-\pi/T}^{\pi/T} j \sin \omega T \, e^{j\omega t} \, d\omega$$

or

$$g(t) = - \frac{\sin \pi t/T}{2\pi} \left[\frac{2T}{T^2 - t^2} \right] \qquad t^2 \neq T^2$$

and

$$g(t)\Big|_{t = \pm T} = \mp 1/2T$$

The intersymbol interference for impulses sent at the Nyquist rate (for the filter function shown in Figure A3-23 the Nyquist rate is $1/T$ bps) is limited to ajacent digits:

$$x(nT) = \begin{cases} 0 & n = 0 \\ \mp 1/2T & n = \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

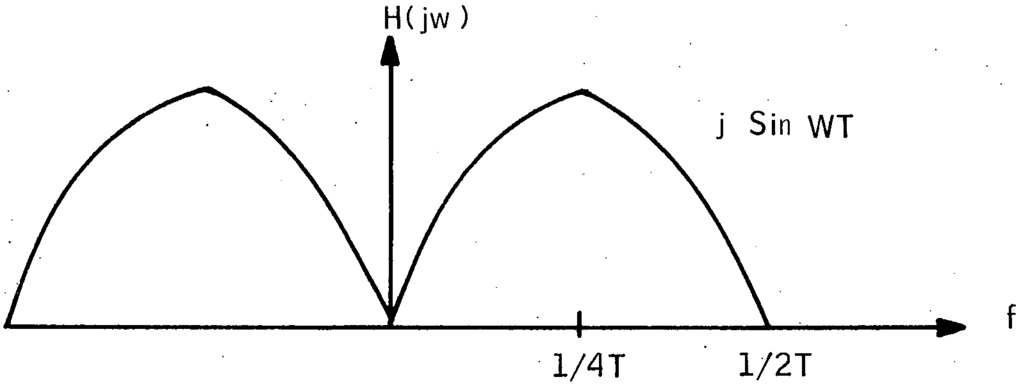


FIGURE A3-23
FILTER FUNCTION FOR GENERATING MODIFIED DUOBINARY

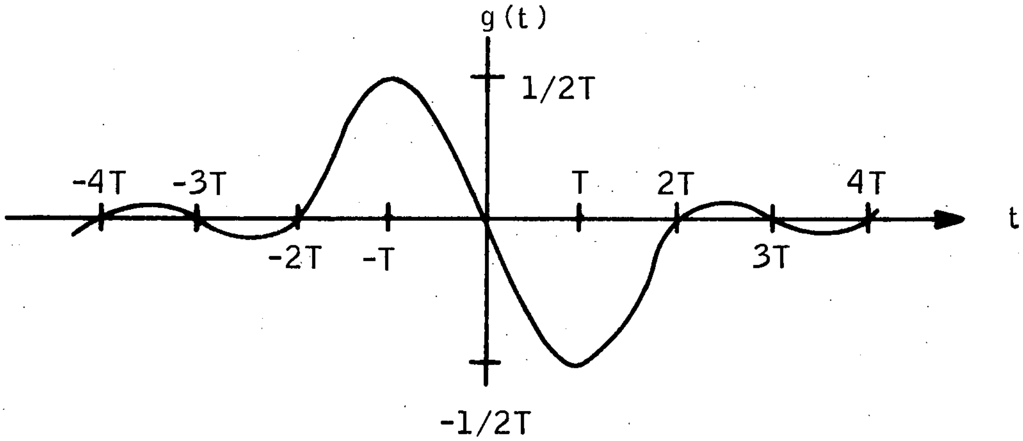


FIGURE A3-24
IMPULSE RESPONSE OF THE MODIFIED DUOBINARY FILTER

The MD line signal is shown in Figure A3-25. Decoding is performed as before using the following decoding equation:

$$a_i = x_{i-1} + a_{i-2}$$

Assuming

$a_0 = x_{-1} + a_{-2} = 0$ (no digit)

$a_1 = x_0 + a_{-1} = -1$ ('0')

$a_2 = x_1 + a_0 = +1$ ('1')

$a_3 = x_2 + a_1 = +1$ ('1')

$a_4 = x_3 + a_2 = +1$ ('1')

$a_5 = x_4 + a_3 = -1$ ('0')

$a_6 = x_5 + a_4 = -1$ ('0')

$a_7 = x_6 + a_5 = +1$ ('1')

$a_8 = x_7 + a_6 = -1$ ('0')

$a_9 = x_8 + a_7 = -1$ ('0')

$a_{10} = x_9 + a_8 = -1$ ('0')

$a_{11} = x_{10} + a_9 +1$ ('1')

This system is capable of Nyquist rate transmission but is unduly complex due to the decoding necessary at the receiver. Also, as a result of the decoding, errors made at the receiver tend to propagate. These two undesirable features can be eliminated by a simple process of "precoding" at the transmitter (Kretzmer (12)). By generating and sending the following sequence, an improved system is realized. Let

$$b_K = \left[a_K + b_{K-2}\right] \bmod 2.$$

That is, interpreting the a_k and b_k as logical variables perform the indicated addition. The b_k are then transmitted as before, a + 1 impulse representing a logical ' '. The resulting sequence is:

k	ak	bk
0	-	-
1	0	0
2	1	1
3	1	1
4	1	0
5	0	1
6	0	0
7	1	0
8	0	0
9	0	0
10	0	0
11	1	1

The line signal resulting from transmitting the b_k is illustrated in Figure A3-26. Except for the first two bits (necessary to "start" the process) there exists a one-to-one correspondence between the non-zero level of the line signal (±2) and the logical 'ones' in the original data sequence. Decoding at the receiver is therefore simplified and is actually a binary decision process. Errors no longer propagate due to the one-to-one relationship mentioned.

Study of the properties of the MD waveform lead to the following rules which can be used to generate the waveform:

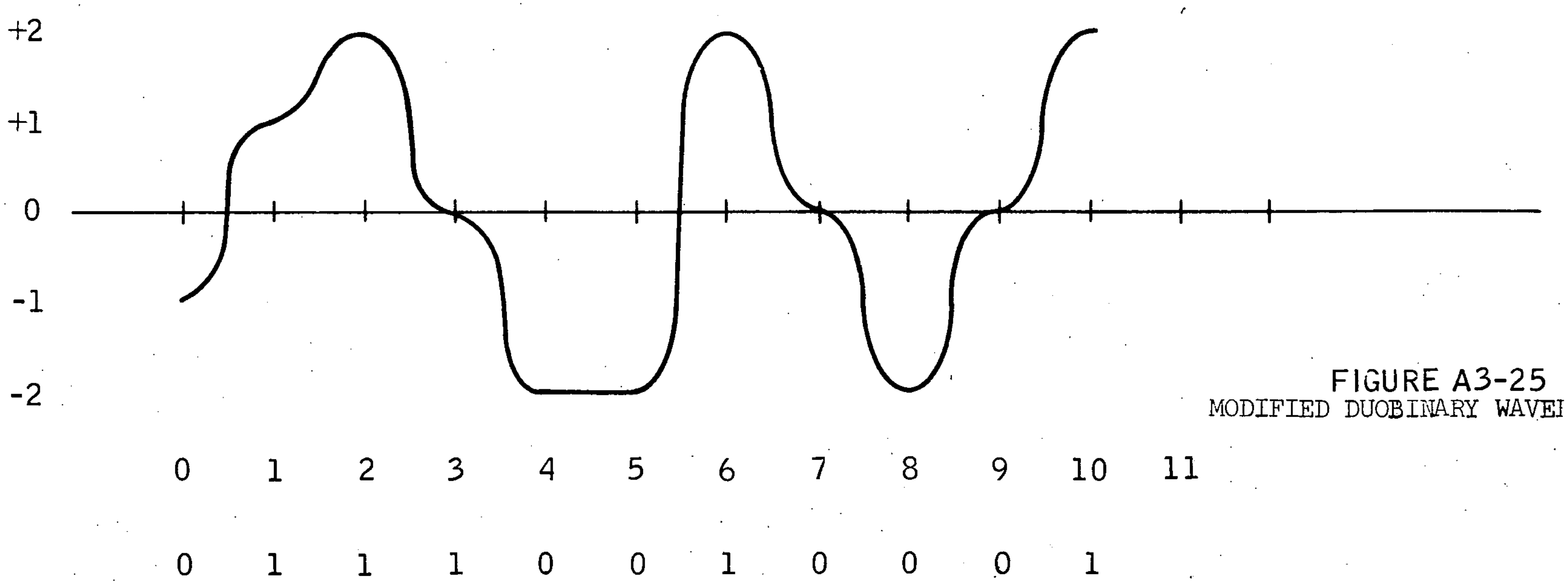


FIGURE A3-25
MODIFIED DUOBINARY WAVE

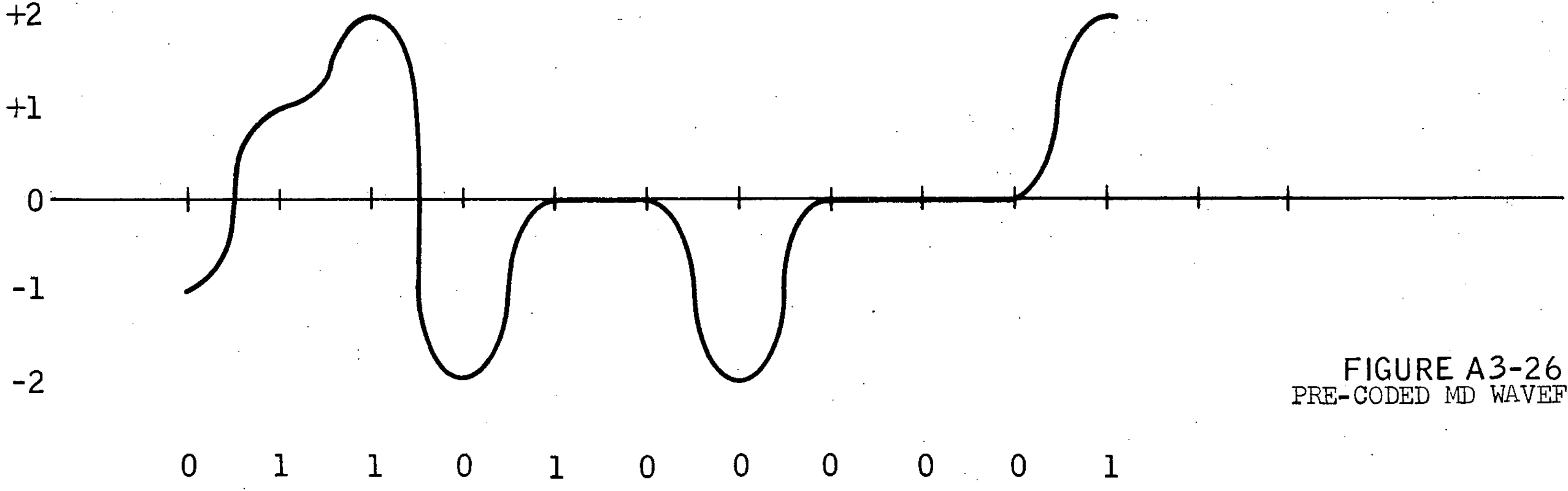


FIGURE A3-26
PRE-CODED MD WAVEFO

1) Data 'zeros' are sent as a zero level

2) Data 'ones' are grouped into consecutive pairs and then sent as follows:

   a) The polarity of the first 'one' in a pair is always different from the polarity of the preceeding 'one'.

   b) The polarity of the second 'one' in a pair is different from the polarity of the first 'one' only if there has been an odd number of intervening zeros.

These rules will be used in the discussion of signal generation circuits. The MD signal has several advantages which include:

1) No DC component

2) Nyquist rate transmission

3) Simple receiver required (binary decision)

4) Relative ease of signal generation (see part 5).

A major problem with the signal is the rel ative difficulty of extracting a clock. This will be discussed in part 4.

The low frequency property of the MD signal is extremely important in data bus applications. Transmitting data at 5 Mbps on the cable described previously results in a signal whose highest component is 2.5 MHz. The maximum attenuation for this signal is approximately .58db/100 feet. By comparision, a split-phase C waveform for 5Mbps data contains energy to 10MHz (see figure A3-5) and is attenuated at approximately 1.2 db/100 feet. The higher attenuation experienced by the split-phase C waveform can be very detrimental over relatively long buses.

Modified Duobinary signalling is very attractive for data bus application for the reasons previously discussed. Parts 4 and 5 will be devoted to the practical problems of implementing a MD data bus system.

One additional feature of MD is of interest here. This is the ability to detect certain transmission error patterns at the receiver without adding redundant digits to the transmission. As an example, consider the following data sequence and corresponding proper line levels:

Data	Level
0	0
1	+1
1	+1
1	-1
0	0
0	0
1	-1
0	0
1	+1
0	0
0	0
0	0
1	-1
1	+1
0	0
1	-1

The line levels are determined from the foregoing rules. If a mistake is made in receiving the 10th bit (as an example), the following sequence might be received:

0
+1
+1
-1
0
0
-1
0
+1
+1   received in error, should be 0
0
0
$\overset{-1}{+1}$
0
-1

The decode algorithm would result in the following data sequence:

0
1
1
1
0
0
1
0
1
1 error
0
0
1
1
0
1

In additional logic is added at the receiver, it can be detected that rule 2b) is violated by the 14th bit. Thus a simple form of error detection capable of detecting all single bit errors is achieved with no additional transmitted bits. The logic required is similar to that used in generating the waveform and will be discussed further in Part 6.

Part 4 - Station Synchronization

The MD signal has several obvious advantages for a data bus application. These include a small bandwidth, ease of decoding, and "free" error detecting capability. These advantages are only available however if a phase coherant reference is available at the receiver. This part of the report will discuss methods of obtaining such a reference.

Figure A4-1 illustrates the considerations involved in initiating a coherant receiver reference. Assuming that a 1-0 sequence initaites all word transmissions, the leading pulse will generate a pulse from the slicing circuit whose width will vary with received signal strength. The desired clock signal has transitions at the center of the received pulses, regardless of signal strength. As indicated in Figure A4-1, this implies a delay of 400 nsec (for 5Mbps transmission) between the center of the received pulse and the initiation of the clock. Figure A4-2 illustrates an approach to pulse-center timing. Three pulses representing different received signal levels are shown at the top of Figure A4-2. The center of these pulses is denoted as t = -400 nsec. corresponding to the initiation of the clock at t = 0. The timing algorithm proceeds as follows: At the leading edge of the first +1 pulse received, a signal with slope equal to m volts/sec is initiated, starting from -V′ volts. At the trailing edge, the slope is increased to 2m volts/sec. The slope (m) is chosen so that

$$2m\,(400\,\text{nsec}) = V' \text{ volts}$$

Figure A4-2 illustrates the result of this algorithm for three pulses of different widths (①, ②, and ③). It can be seen that the waveforms

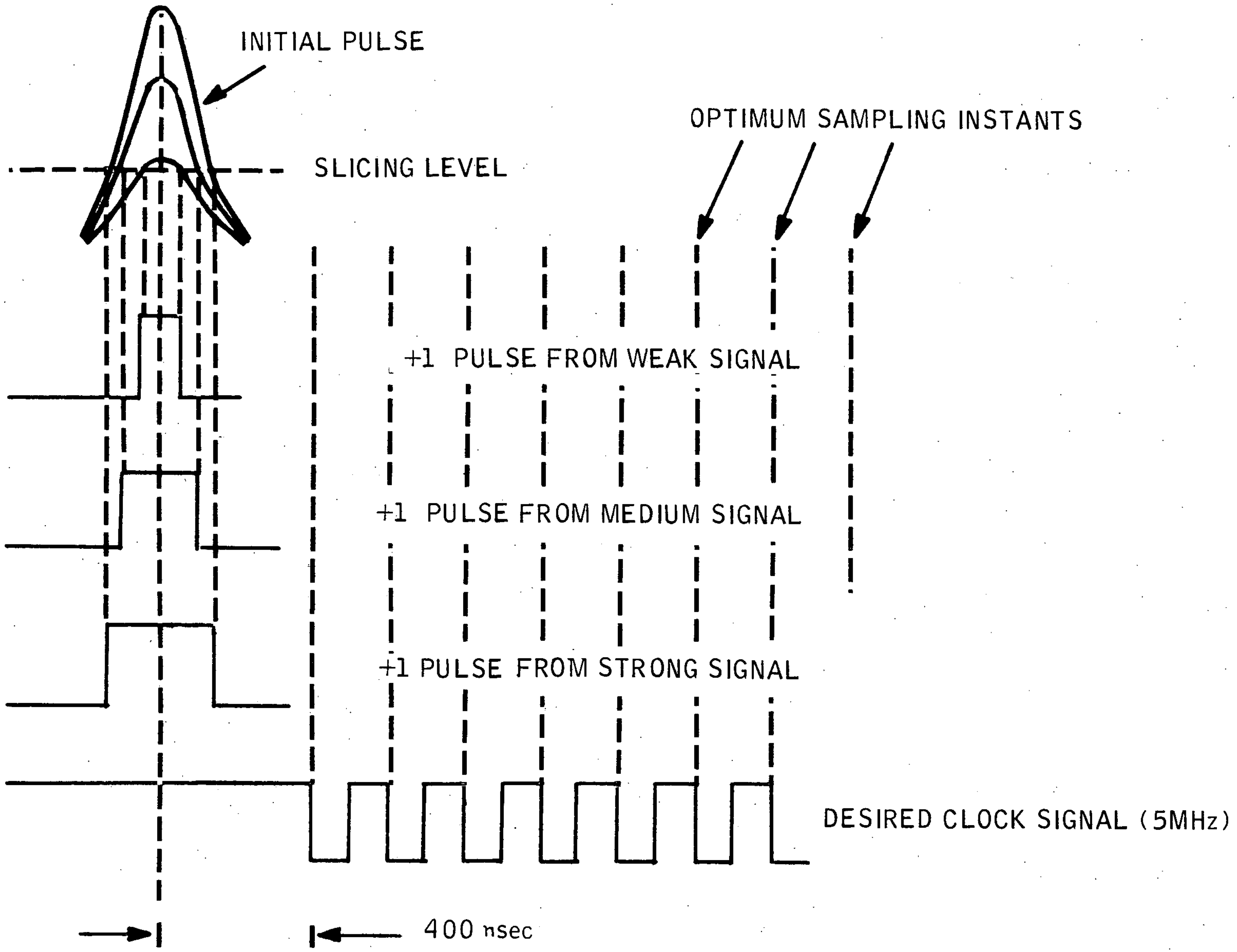

INITIAL PULSE
SLICING LEVEL
OPTIMUM SAMPLING INSTANTS
+1 PULSE FROM WEAK SIGNAL
+1 PULSE FROM MEDIUM SIGNAL
+1 PULSE FROM STRONG SIGNAL
DESIRED CLOCK SIGNAL (5MHz)
400 nsec

FIGURE A4-1

PULSE CENTER TIMING

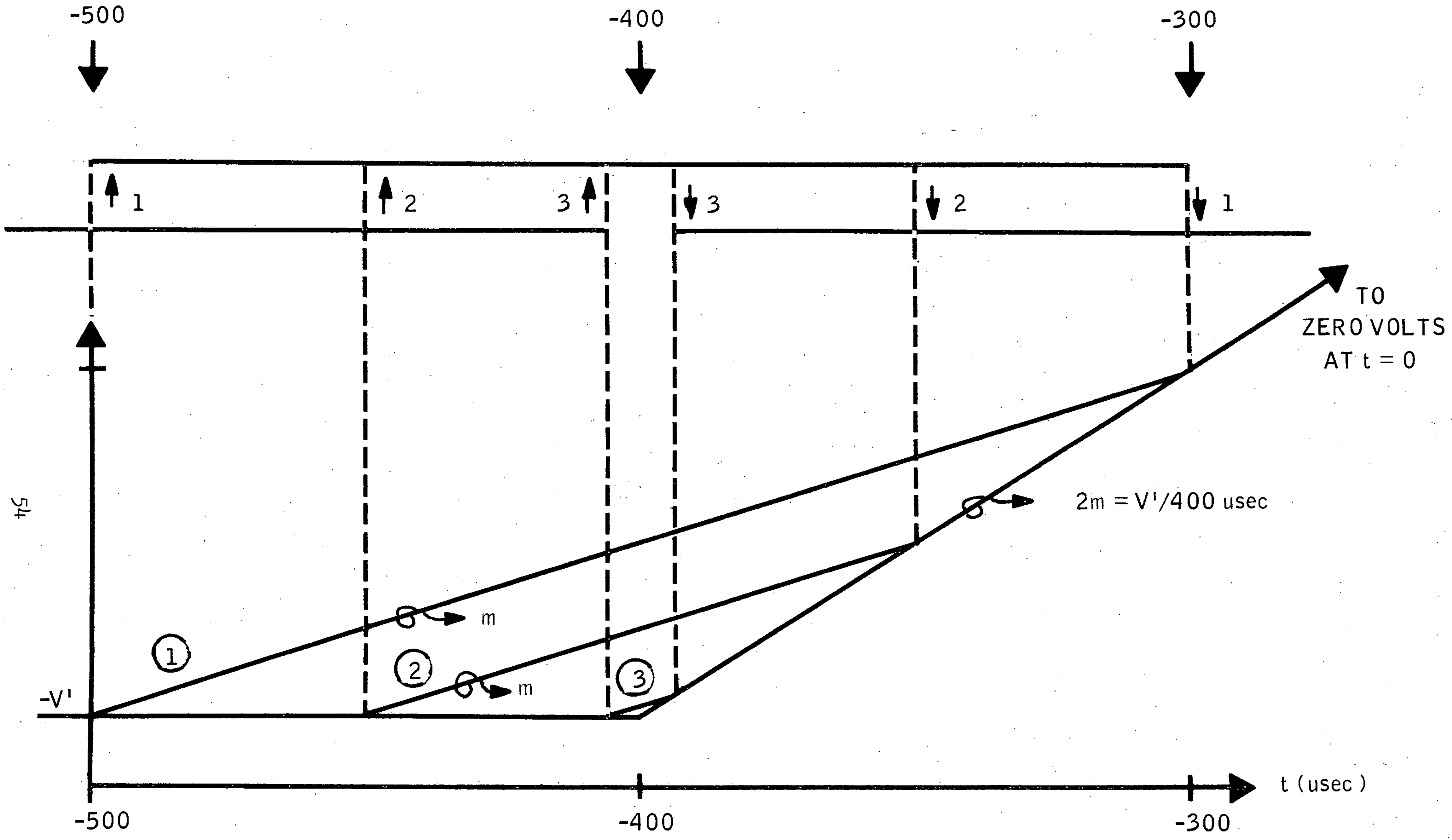


FIGURE A4-2

PULSE CENTER TIMING ALGORITHM

resulting from the three different pulse widths all will cross zero volts at t = 0. This event can then be used to initiate the receiver clock.

A circuit for performing the pulse-center timing algorithm described above is shown in Figure A4-3. A timing diagram is shown in Figure A4-4. The circuit is easily understood. Note that F/F1 and F/F2 are used to activate the two current sources (T1 and T2). The current value (I) is chosen so that

$$m = V'/800 \times 10^{-9} = I/C$$

The algorithm of Figure A4-2 is therefore implemented.

Figure A4-4 illustrates the manner in which e_c is "reset" prior to the next word. When e_c reaches zero volts, A_2 clocks F/F3 so that F/F1 and F/F2 are cleared. This occurs at t = 0. The current sources are therefore switched off and e_c decays toward V'. Note that

$$V' = (15) R_2' / (R_1' + R_2').$$

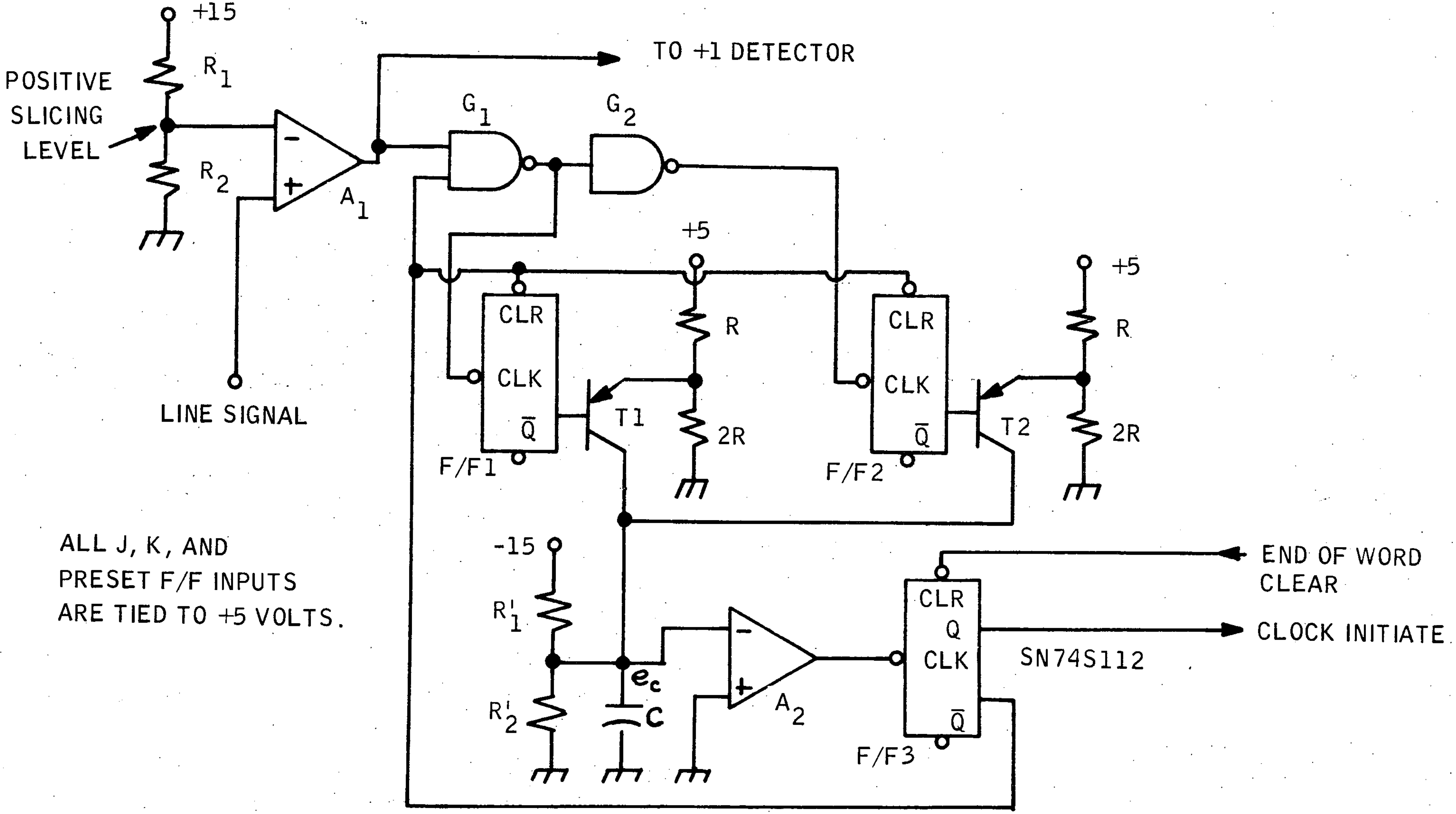
The decay of e_c proceeds as

$$e_c(t)\Big|_{t \geq 0} = -V' \left(1 - \exp\left(-t/R''C\right)\right)$$

where

$$R'' = R_1' R_2' / (R_1' + R_2')$$

The time constant (R"C) is selected so that the decay is completed before the end of the word, thus effecting the "reset" of e_c(t).



ALL J, K, AND PRESET F/F INPUTS ARE TIED TO +5 VOLTS.

FIGURE A4-3
PULSE CENTER TIMING CIRCUIT

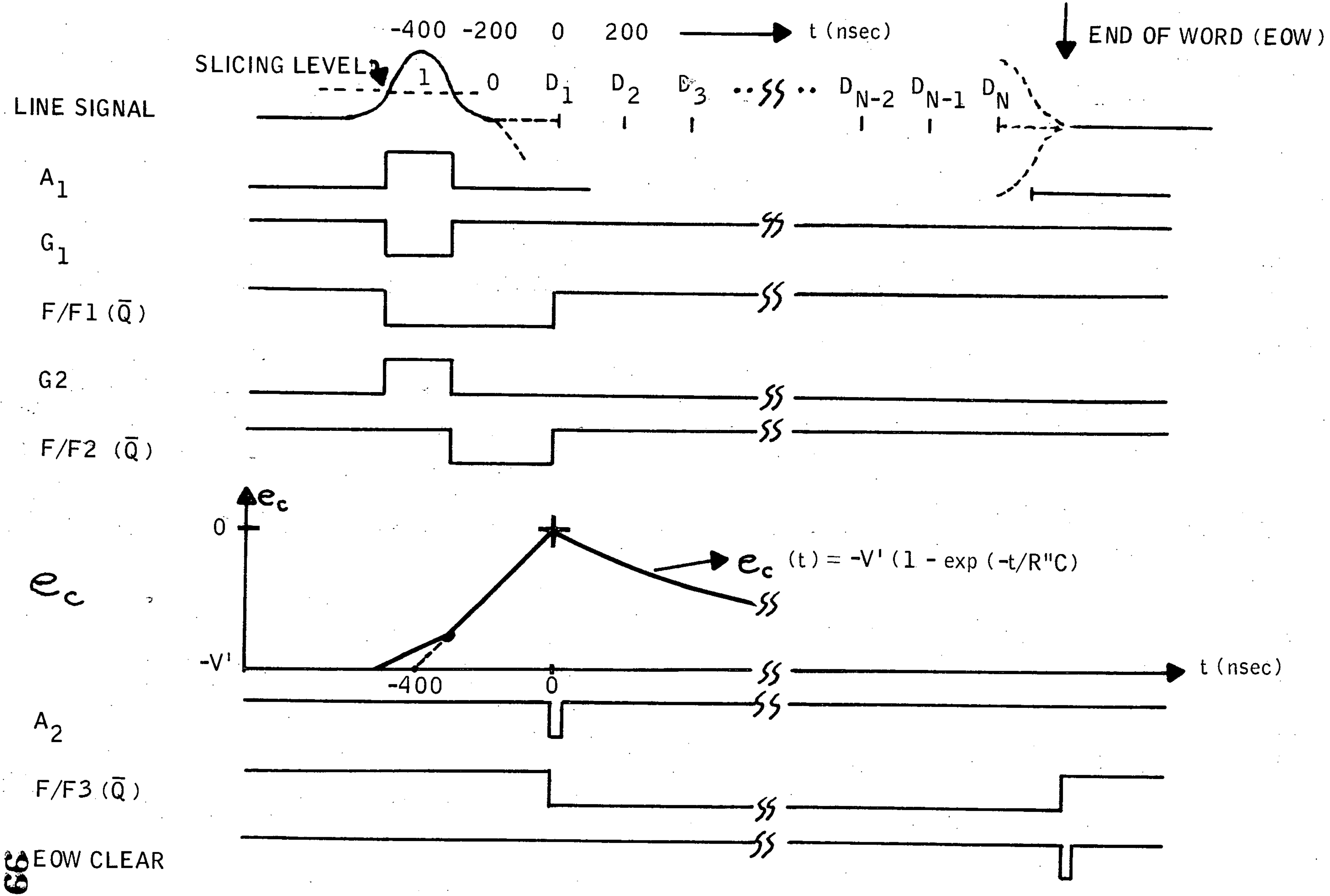


FIGURE A4-4
PULSE CENTER TIMING DIAGRAM

The initiate pulse is used to start a receiver clock at 5MHz. It is assumed that both transmitter and receiver clocks are crystal controlled to minimize phase "slippage" during the reception of a word. Even if this is the case, a problem exists in starting the receiver clock. When power is initially applied to a crystal clock it may take as long as several seconds for the output frequency to stablilize. Obviously this is excessive for the application discussed above. Figure A4-5 illustrates one method of using the initiate pulse with a crystal clock. The clock runs continuously at 50MHz (for a 5Mbps system). The initiate pulse is used to enable a divide-by-ten circuit so that the phase of the resulting 5MHz clock is controlled by the initiate pulse (within 36°).

The above method of center-pulse timing will produce optimum clock timing, responsive to changes in received signal strength. Under ordinary data bus conditions, the complexity of the above method may not be justified. A much simpler approach which has been successfully tested at station separations of up to 2000 feet is outlined below.

During breadboard evaluation of the MD technique it was apparent that the signal level, as indicated by the width of received pulses, does not vary appreciably for surprisingly large station separations. This is especially true when the transmitter generates square pulses as described in Part 5 and does not use a Sin WT filter as discussed in Part 3. At 5Mbps the pulse width variation over 2000 feet of TWC 1244 cable was small enough to use the timing system described below.

Figure A4-6 illustrates the simplified timing circuitry. The set line of the flip-flop is activated by the first received edge. Assuming that a

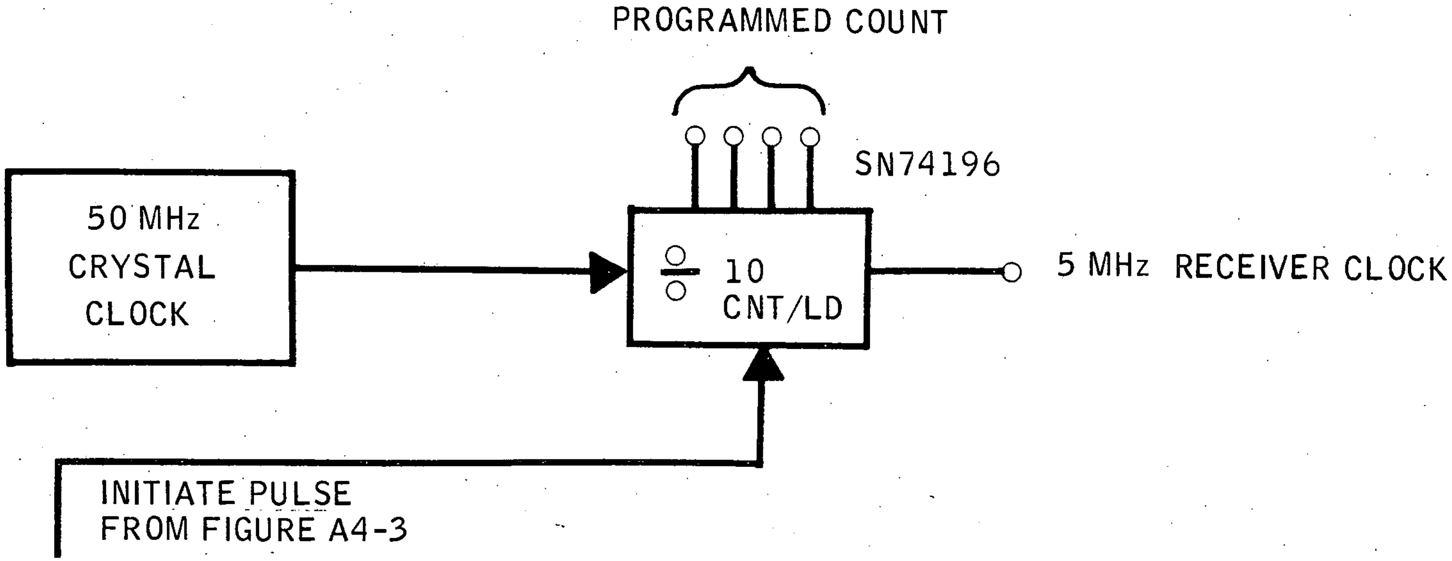


FIGURE A4-5
CLOCK PHASING CIRCUIT

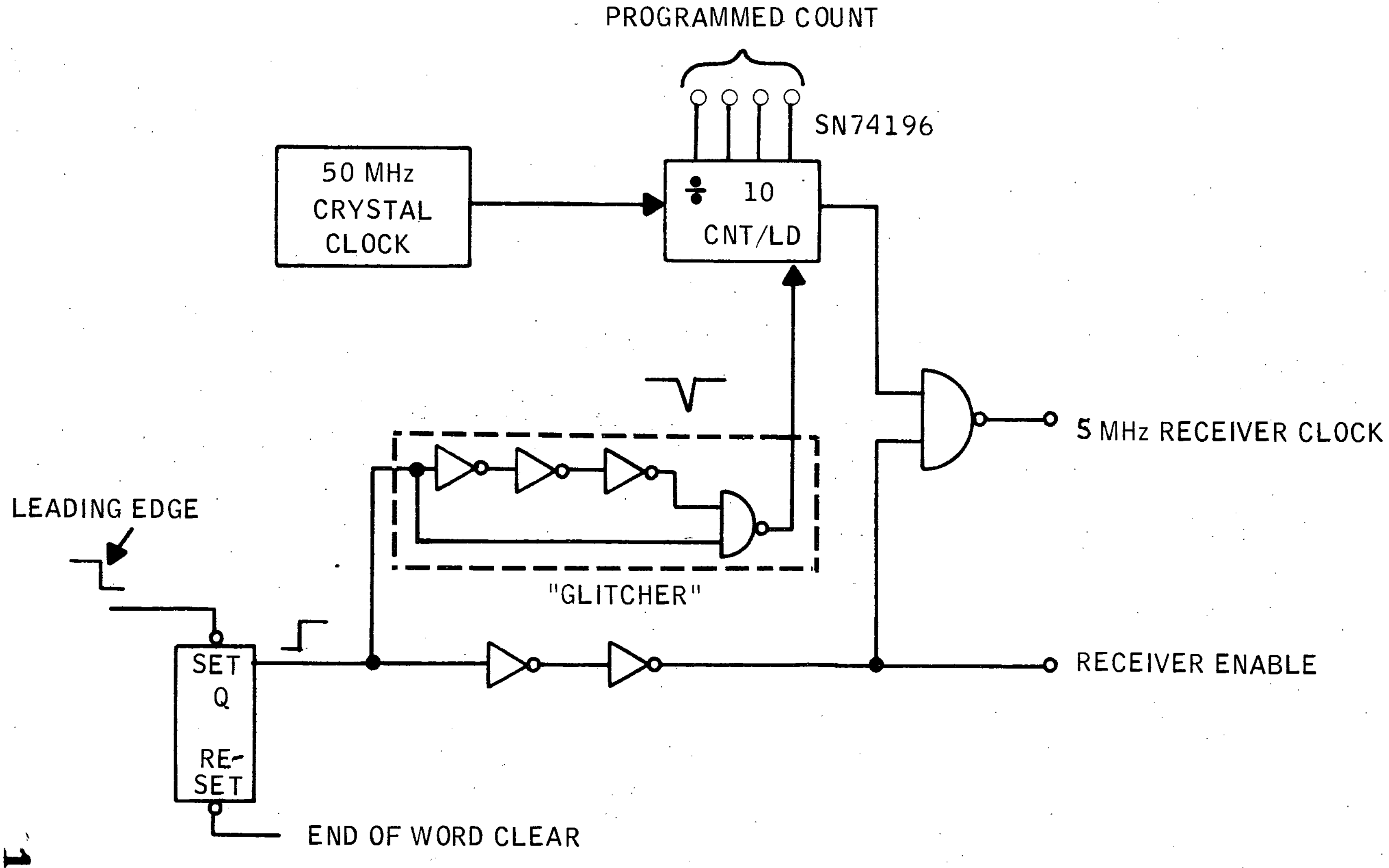


FIGURE A4-6
SIMPLIFIED MD TIMING

1 preceeds every transmission, the leading edge is detected and used to set the F/F. When the Q output goes high in response, the "glitcher" circuit produces a narrow load pulse to the divide-by-ten. The Q output is also used to enable the remainder of the receiver and to gate the 5MHz clock to the receiver. The 5MHz clock generated in this fashion has a predictable relationship to the leading edge of the received word. This phase relationship can be controlled and altered by changing the programmed count which is loaded into the SN74196. The receiver logic is responsible for keeping track of the number of received bits and resetting the F/F after the word has been received.

When the expected signal-to-noise ratio is high (as in the usual case due to the excellent shielding properties of balanced lines) the clock can be resynchronized with every received leading edge. This allows the use of long data words without allowing the accumulated phase error to become excessive. When only the first pulse is used for synchronization, the following relationship is useful:

$$\Delta\phi = 2\pi|f_T - f_R|T$$

The above equation can be used to determine how precise the transmitter and receiver frequencies must be. $\Delta\phi$ is the total phase slippage (in radians) during the length of the word. A reasonable tolerance on phase match between the receiver clock and the received data is 25% or $\pi/2$ radians. The timing circuit produces a 10% (worst case) skew so that 15% ($.3\pi$) is allowable due to frequency mismatch. As an example, consider $T = 8 \times 10^{-6}$ sec. (corresponding to a 40 bit word at 5Mbps). Solving for Δf in the above equation results in a maximum allowable frequency difference of

$$\Delta f = \frac{.3\pi}{2\pi T} \simeq 20KHz$$

Each oscillator can contribute 10Khz of the frequency difference so that 5MHz ± 10KHz is the necessary tolerance. This is .2%, easily achieved with crystal clocks.

A method of transmitting and receiving MD without the use of crystal clocks is discussed in Part 5. This method, although not completely evaluated, shows promise of providing very inexpensive data bus hardware for non-critical applications.

Part 5 - Signal Generation and Receiving Circuits

Several methods of generating MD signals have been developed. These will be discussed below.

The natural method of generating MD signals is indicated by the discussion in Part 3. This method is illustrated in Figure A5-1. Two design problems are apparent from the Figure: The design of a suitable impulse generator (especially difficult at high data rates) and the design of a reasonable approximation to the desired j Sin WT filter function. Neither of these problems is insurmountable but they were not attacked during the course of this study.

The approach indicated in Figure A5-1 is more difficult than the alternate methods discussed below but offers at least one advantage. The energy is completely confined to a band below half the data rate. Because of this feature, a frequency multiplexed data bus system using single sideband techniques can be considered. Part 7 discusses this concept further.

If complete containment of the signal energy to a band below half the data rate is not necessary, a great simplification in transmitter hardware can be obtained. The discussion in Part 3 concerning the precoding operation involved in MD indicates a transmitter as shown in Figure A5-2. The timing diagram for this transmitter is shown in Figure A5-3. It is easily verified that the line signal shown in Figure A5-3 follows the rules discussed in Part 3 for MD transmission.

The circuit operation is based on the discussion of precoding in Part 3. Flip-flops one and two simply delay the data so that all data zeros inhibit

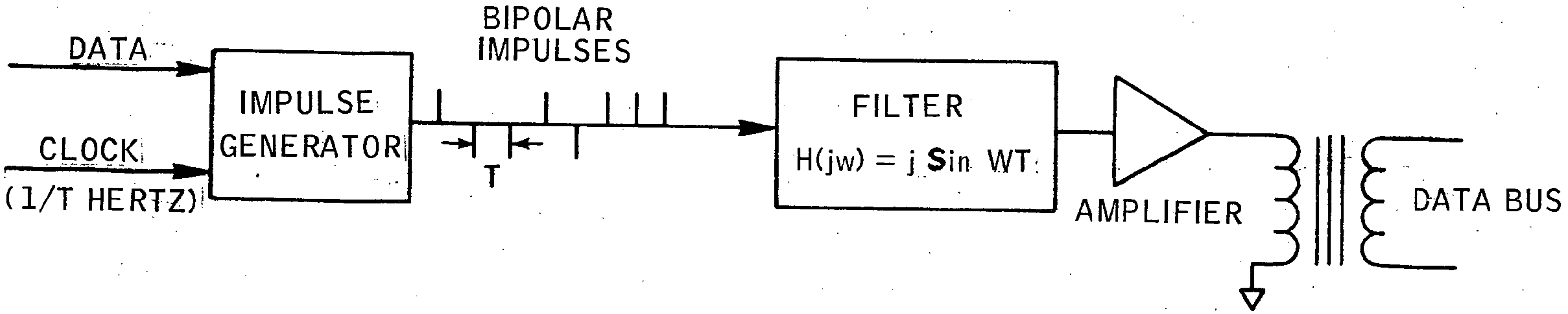


FIGURE A5-1
NATURAL GENERATION OF MD SIGNALS

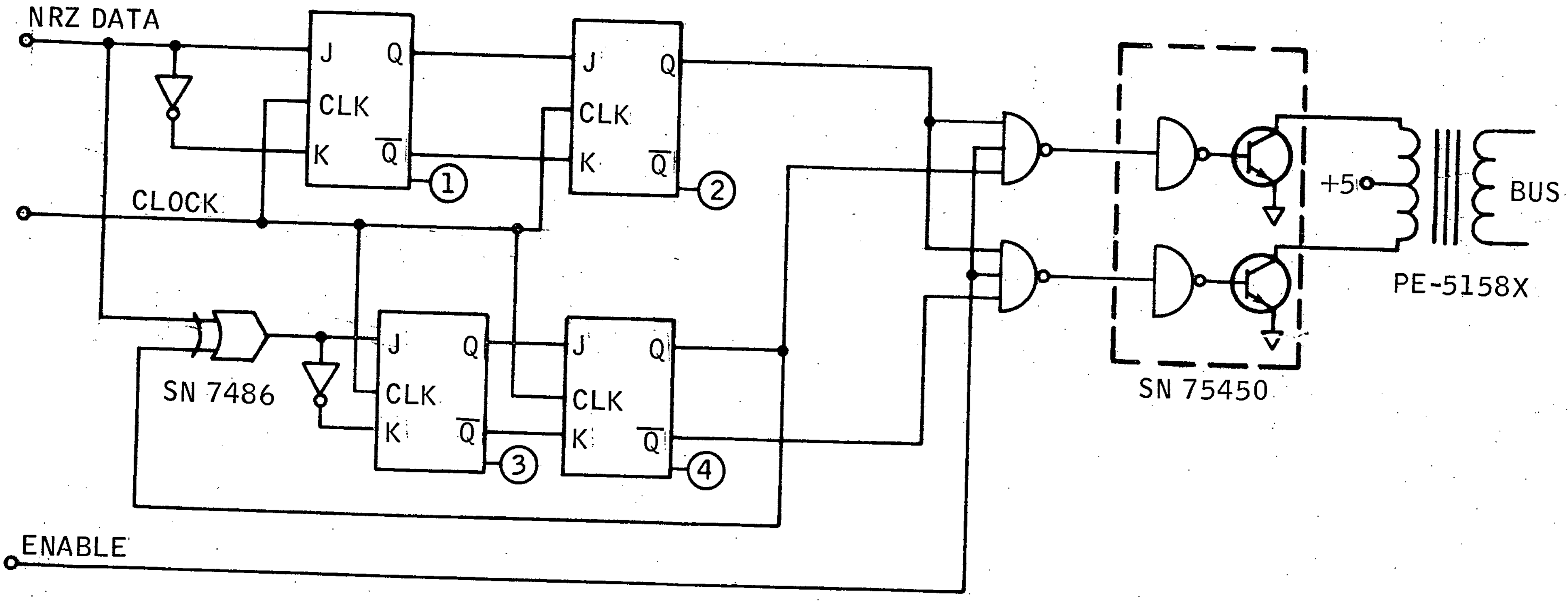


FIGURE A5-2
SIMPLIFIED MD TRANSMITTER

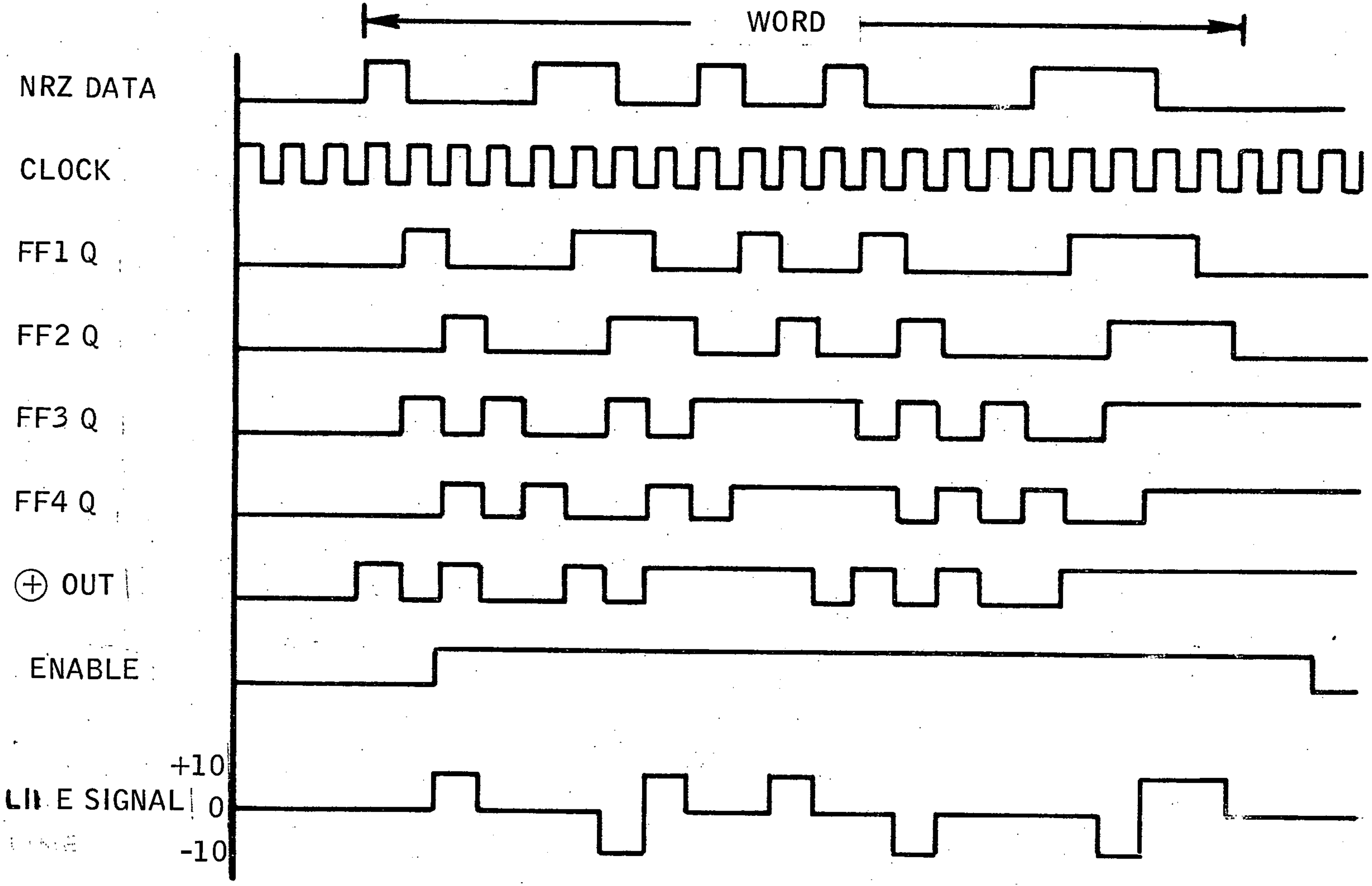


FIGURE A5-3
TIMING DIAGRAM FOR MD TRANSMITTER

the three input NAND gates. These gates are enabled for ones occurring in the data. The gates are then controlled by the outputs of flip-flop 4 to determine which polarity pulse is to be sent. The circuit consisting of flip-flops 3 and 4 and the exclusive -OR gate (SN7486) is derived from the discussion in Part 3. The enable line is necessary to prevent the gates remaining enabled after a word. With the enable line low, the transistors are forced off and the transmitter presents no load to the bus. The transformer can therefore be used for reception as well. This gating also prevents transformer saturation and subsequent transistor destruction.

The line signal illustrated in Figure A5-3 is easily seen to follow the MD rules. As previously indicated, the energy is not confined to frequencies below half the bit rate but actually extends to relatively high frequencies as a result of the "square wave" generation used. The signal value at the sampling points (the center of each bit time) is attenuated relatively slowly however so that the benefit of small spectral occupancy of the MD signal is retained. The voltage levels indicated in Figure A5-3 result from the implementation of Figure A5-2. Figure A5-4 illustrates the signal deterioration after 2000 feet of TWC-1244 cable at a 5Mbps data rate. The receiver described below and shown schematically in Figure A5-5 is capable of correctly decoding this waveform with a bit error rate of less than 10^{-6}. Of particular interest is the reduction in the p-p amplitude of the waveform. 1.25MHz is the single frequency most representative of the waveform for 5Mbps data (see Figure A3-23). Using the equation for db/100 feet attenuation for the TWC-1244 cable at 1.25MHz a total loss of about 8.3db is expected. The ratio of the received p-p voltage ($\sim$ 8 volts) to the transmitted p-p voltage ($\sim$ 20 volts) is 8db,

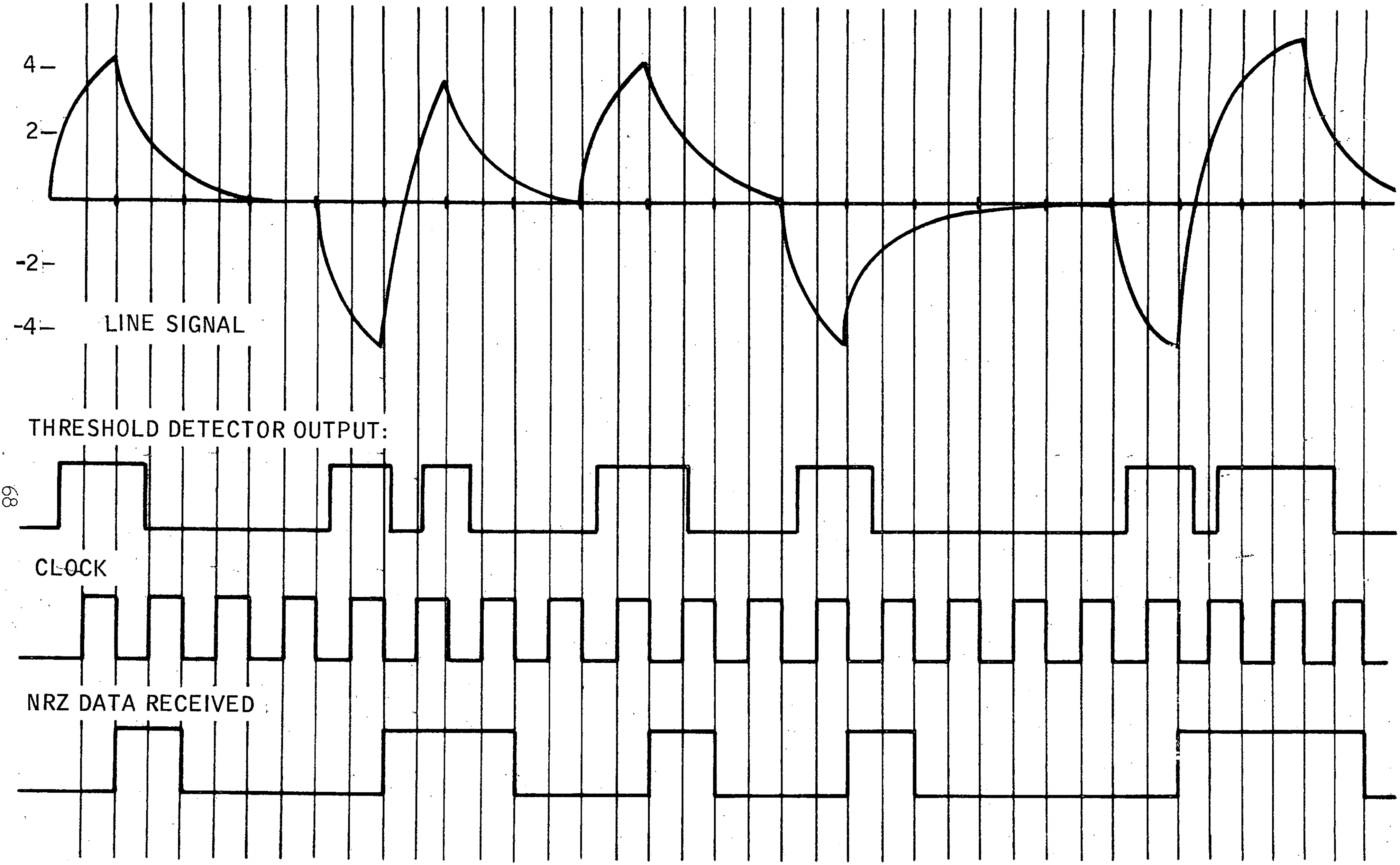


FIGURE A5-4
RECEIVER WAVEFORMS AFTER 2000 FEET OF CABLE

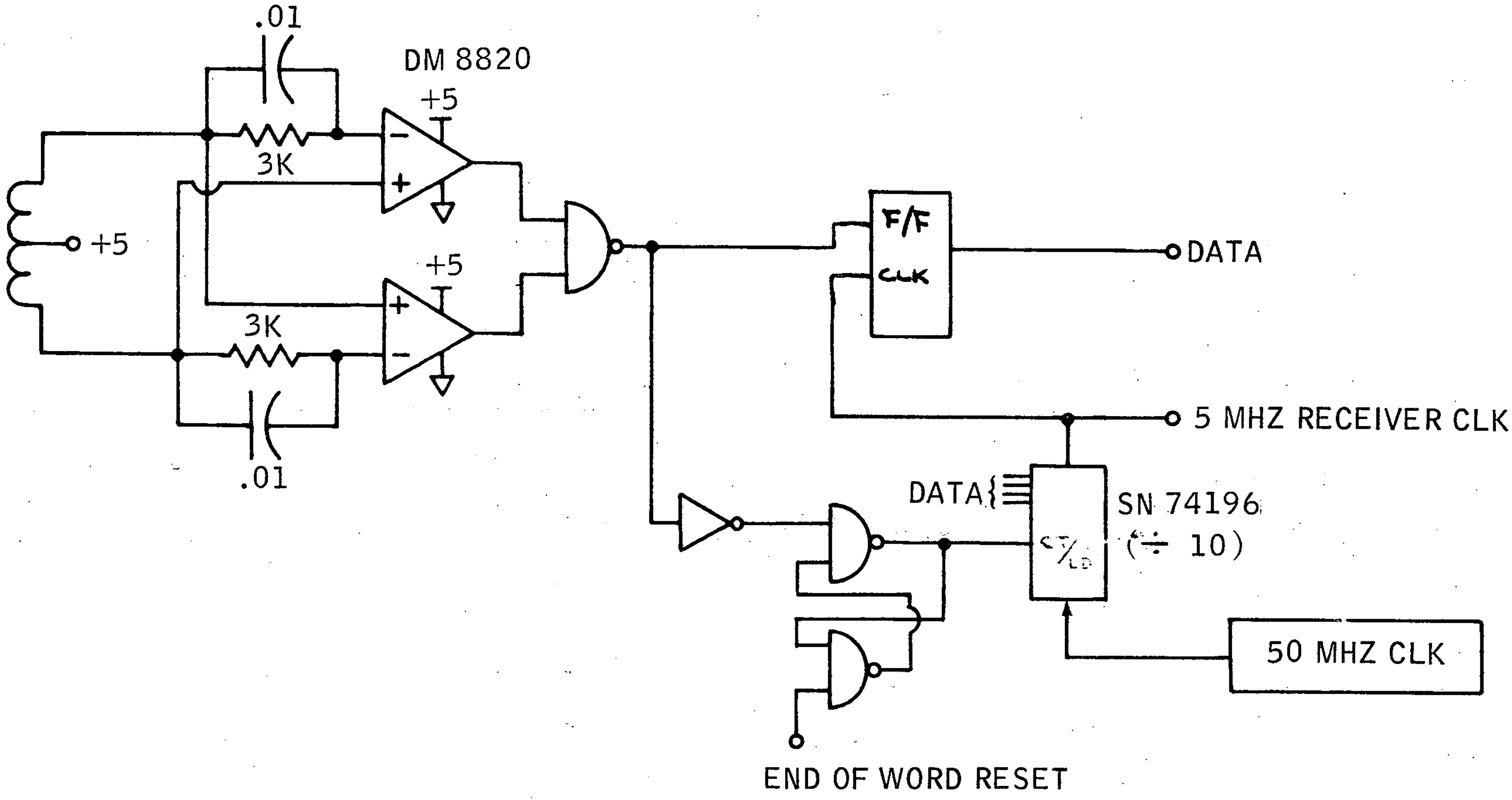


FIGURE A5-5

MD RECEIVER

demonstrating that although the signal generated by the transmitter of Figure A5-2 does not occupy the smallest bandwidth possible, the signal is attenuated the same as a true MD signal.

A receiver design which performs well at any distance up to 2000 feet is shown in Figure A5-5. The leading edge is used to set the gate latch and enable the divide by ten (SN74196). The resulting 5MHz clock is properly phased when a count of 1 is loaded prior to enabling the SN74196. The flip-flop is then used to re-time the data and present it to the receiver logic. Following the word, a short end of word clear pulse is required to reset the latch.

The 3K Ω resistors shown in Figure A5-5 serve as a convenient means of setting the receiver threshold. The threshold is easily calcuated from the voltage divider formed by the 3K Ω resistor and the inverting input impedance of the DM8820 (~ 5K).

Note that the transformer center tap is tied to +5 volts so that the same transformer can be used with the transmitter circuit of Figure A5-2. An additional feature of both the transmitter and receiver designs is that a single +5 volt supply is all that is required.

One additional method of signal generation and recovery is illustrated in Figures A5-6 through A5-8. This method is intended for non-critical applications and fairly low data rates. The circuits require no clocks for operation as the necessary timing is derived from delay lines.

The clockless transmitter is shown in Figure A5-6. The NRZ data is used to gate two clocks, 1 and 2. The clock 1 oscillator runs at 1/T Hertz

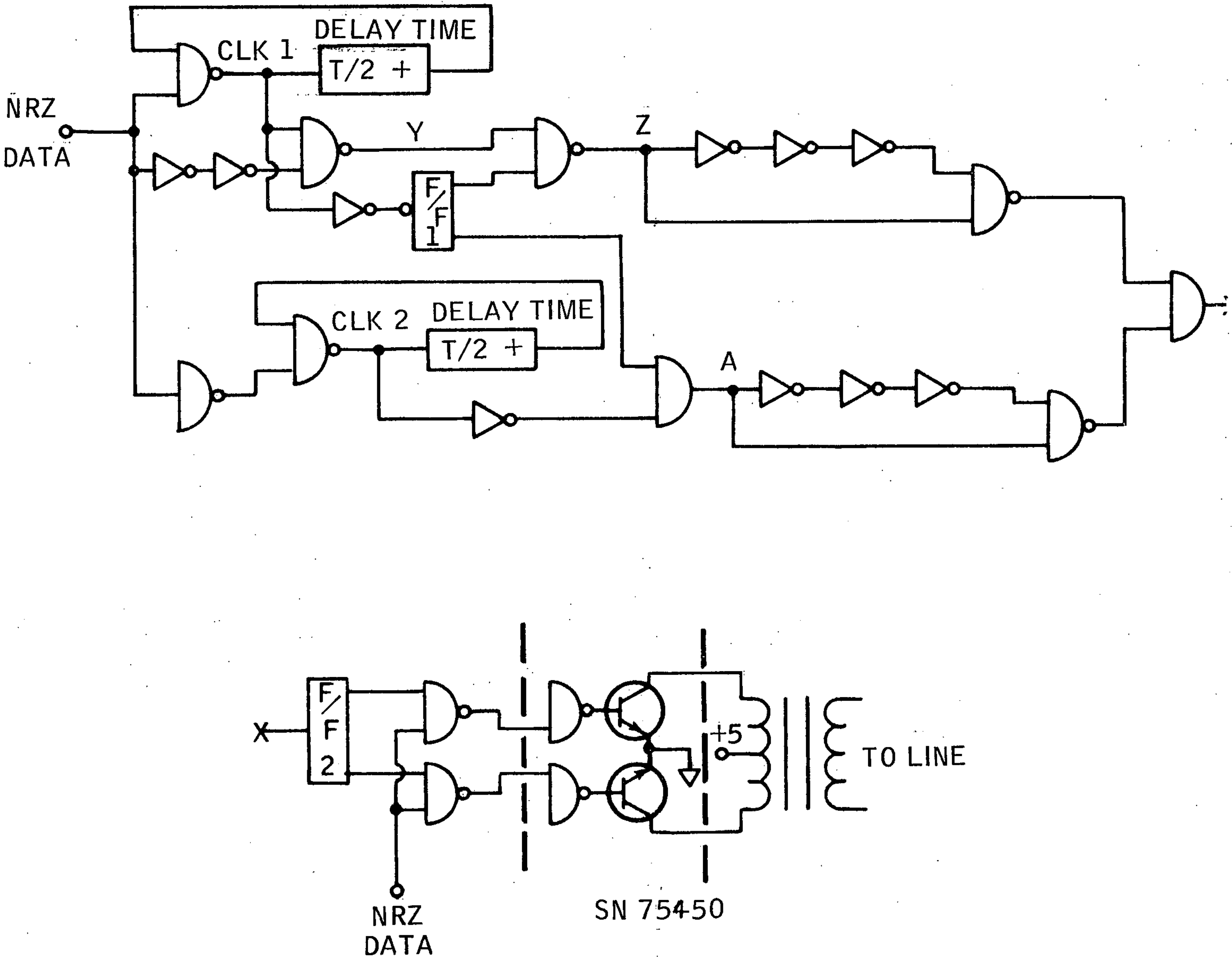


FIGURE A5-6
CLOCKLESS MD TRANSMITTER

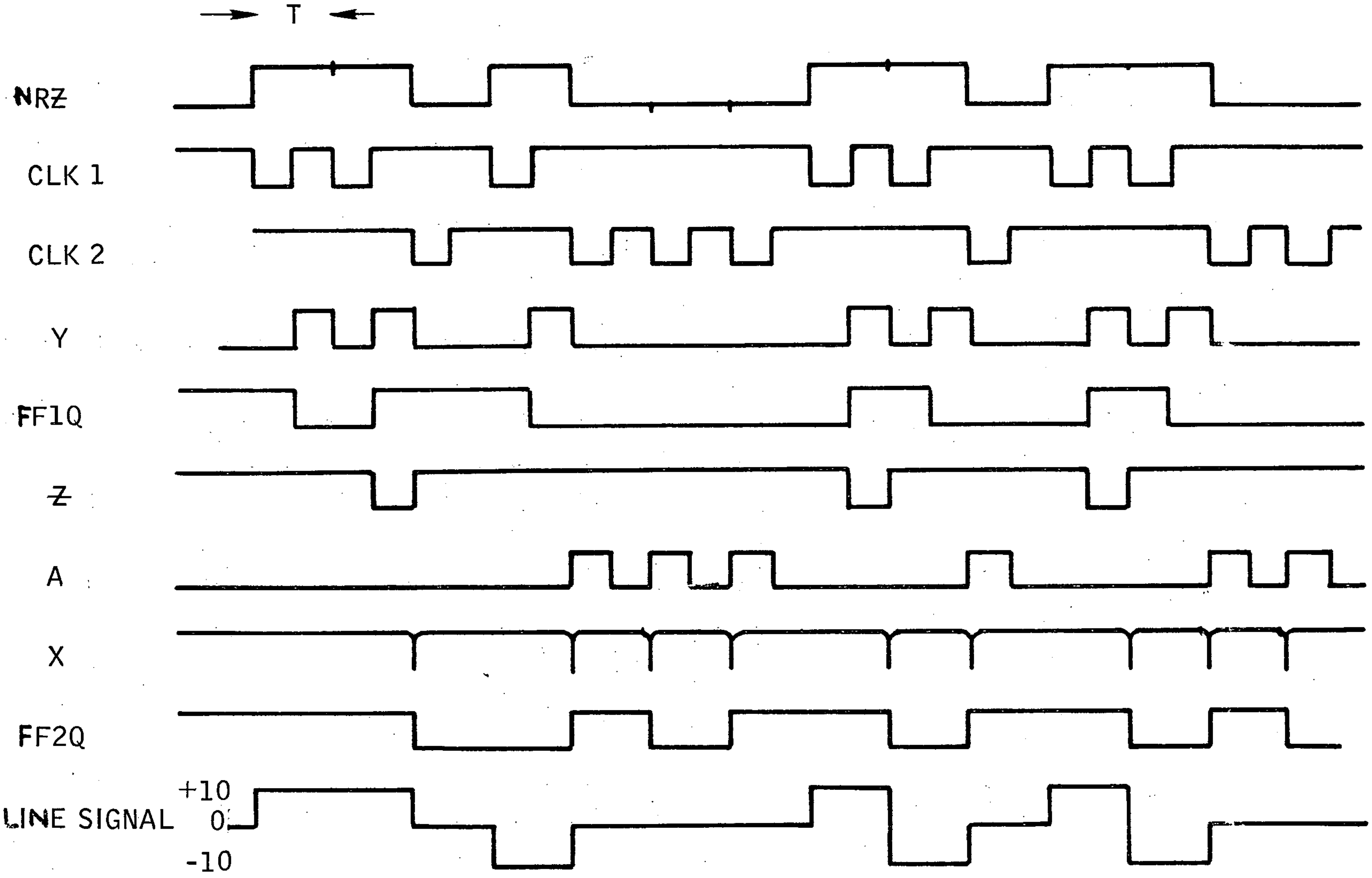


FIGURE A5-7
CLOCKLESS MD TRANSMITTER TIMING DIAGRAM

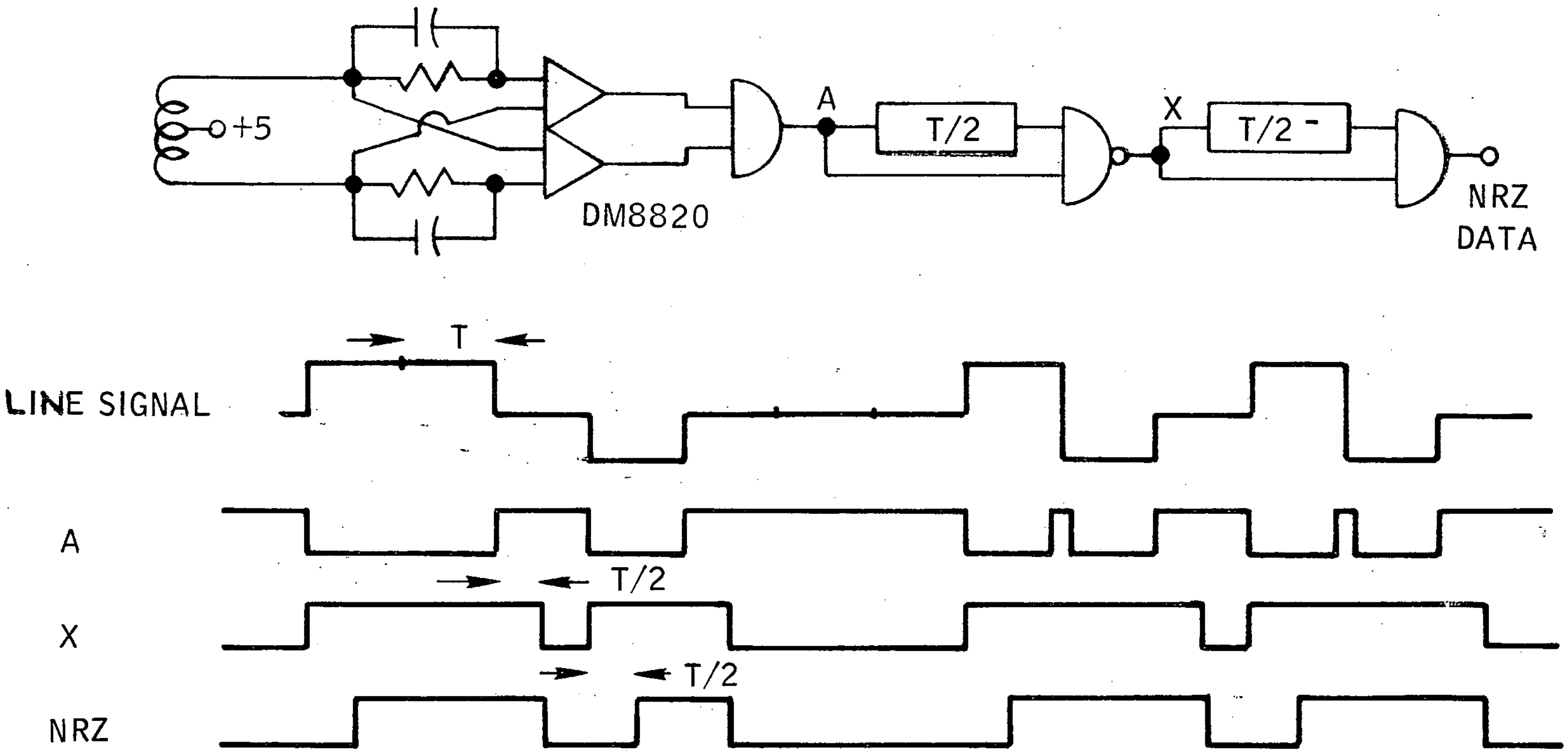


FIGURE A5-8.
CLOCKLESS MD RECEIVER CIRCUIT AND TIMING DIAGRAM

during logical 1's in the data and the clock 2 oscillator runs during logical 0's. The rules for MD signals, previously discussed are used to combine the ones and zeros information into F/F2. This flip-flop is used to steer the output pulses as was done in the transmitter of Figure A5-2. The data is used directly to gate the line driver (SN75450) so that zeros are sent as a zero level. The timing diagram for a typical data sequence is illustrated in Figure A5-7.

The clockless receiver is shown in Figure A5-8 along with a typical timing sequence. The signal at A is close to being NRZ as received. The "glitches" in the data result from the line signal changing polarity between successive ones. The two delay circuits eliminate the "glitches" and present clean NRZ data as an output. If a clock is required, it can be easily generated with two delay line oscillators as used in the transmitter of Figure A5-6 and appropriate gating.

It is felt that the clockless transceivers could find application at low data rates or short distances in non-critical situations. The spread in delay time parameters and gate delays would probably limit the data rate for such transceivers to 5Mbps at short distances (about 300 feet). Slower rates could result in successful operation over longer distances. In such applications, the cost of these transceivers would be minimal.

Part 6 - Error Control

Error control for a data bus can be either error detecting or error correcting. The turn around and delay times for an asynchronous data bus are relatively short so that simple error detection followed by retransmission should suffice for most applications.

Following detection of an erroneous reception, the receiving station must request retransmission of the data. In a system such as that described as "Approach A" in Part 2 of this report, the retransmission request can consist of a failure to return the "OK" to the transmitting station. The transmitter would repeat the transmission until an "OK" was returned or until it decided that there was no hope.

Approach B as described in Part 2 of this report would offer no advantages over Approach A since some form of "OK" would have to be added to the system.

The MD signal contains built in error detection capability as previously mentioned (see Part 3). The capability results from the fact that the transmitter followed the rules of MD when generating the line signal. If the rules are not followed by the received data, an error indication can be generated. This form of error detection can supplement the more powerful approach described later (BCH codes). In certain applications, the detection of single bit errors may be sufficient. As an example, in a short bus the raw error rate will be very low. The most probable event will be a single bit error/word which can be detected. It may not be worth the requisite redundant bits and logic complexity to implement a more sophisticated method of error control.

The error detection circuitry is based on the MD rules. It is possible to predict the polarity of the next '1' in the MD signal from a knowledge of the preceding data. This was illustrated in Part 3. The circuit of Figure A6-1 does this prediction and compares the predicted polarity with the received polarity of each '1' in the signal. The circuit timing diagram is presented in Figure A6-2 for a 16 bit word with no received errors detected.

The circuit operation is straightforward. F/F1 toggles with a data-gated clock. The output of F/F1 thereforeindicates if there has been an even or an odd number of preceding '1''s in the received data. This information is used to gate the clock to F/F2 which therefore indicates if the number of '0's between a pair of received '1''s is even or odd. F/F2 predicts if the next bit, if it is a one, should be a +1 or a -1. The exclusive OR gates are used to check if the predicted polarity is correct.

The check will obviously fail during the received 0's so that the signal at A is required to gate the check signal to F/F3 only during received '1''s.

The edges of this gated signal which rise or fall near a falling edge of the clock signal are generated from the clock signal. Due to this, the transitions of interest on F/F3J always occur after the point in time at which F/F3 is clocked (negative going clock transitions). This assures proper operation and results in the "no detected error" indication from F/F3Q as shown.

Figure A6-3 illustrates the detection process if bit #8 is incorrectly received as a zero. As can be seen, the detection is not instantaneous.

3

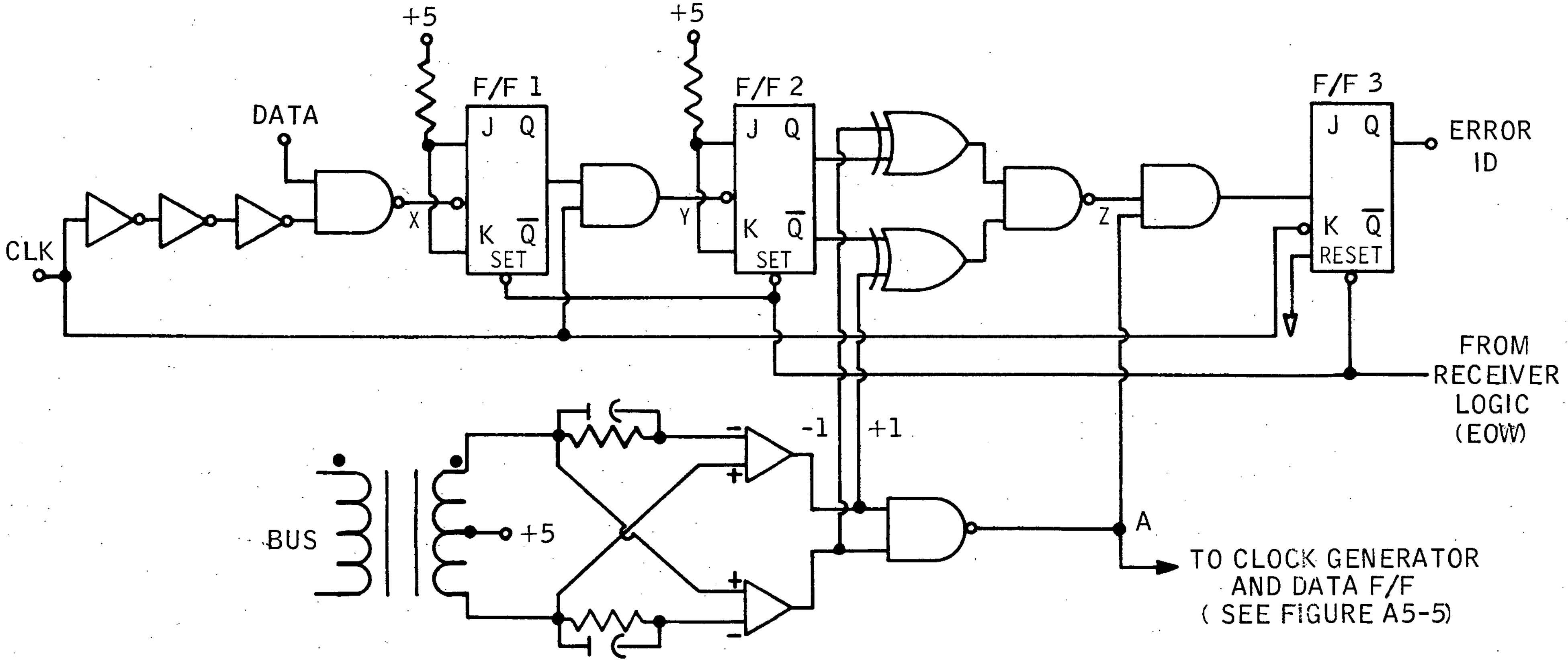


FIGURE A6-1

MD ERROR DETECTION CIRCUIT

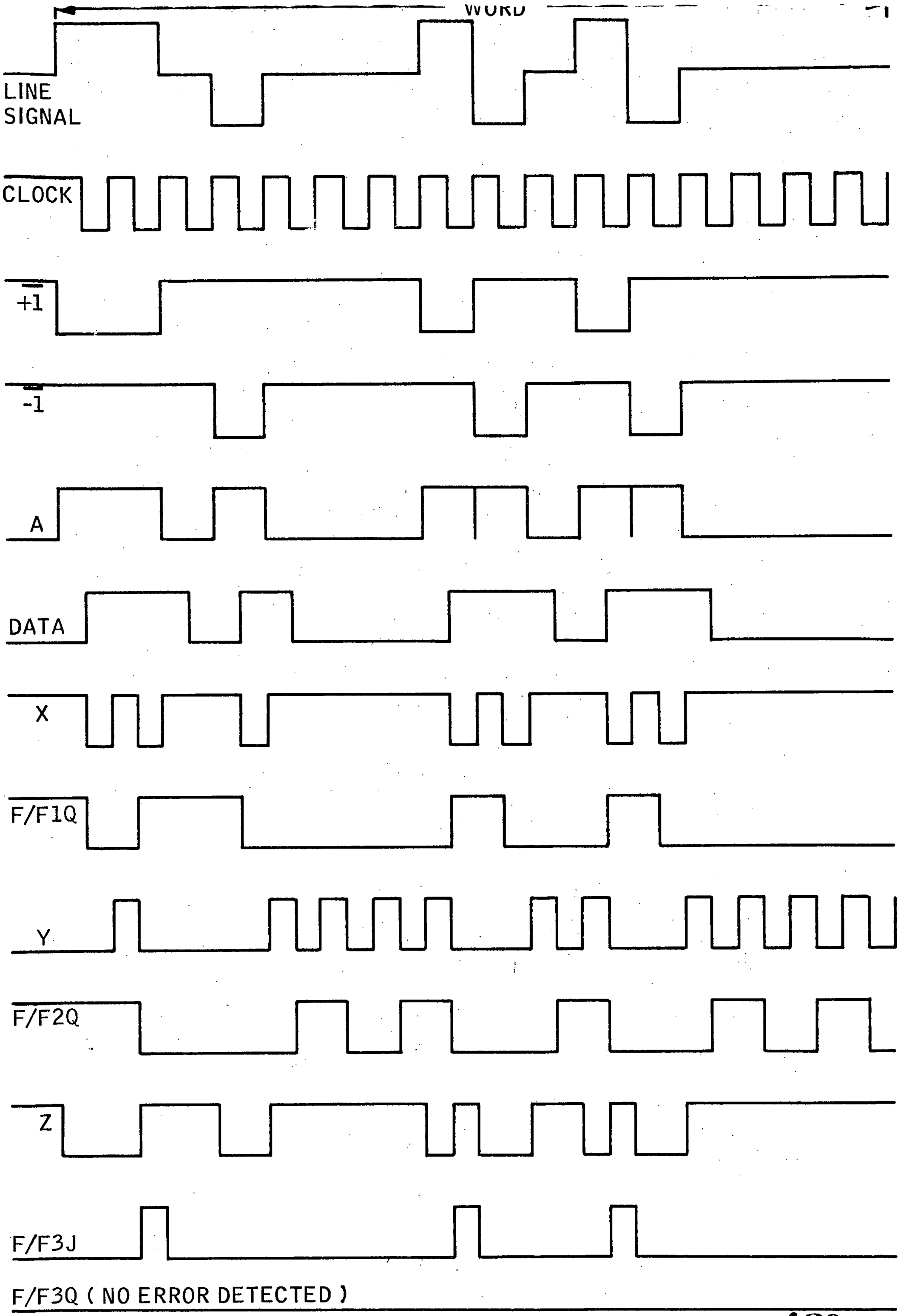


FIGURE A6-2
MD ERROR DETECTION TIMING DIAGRAM

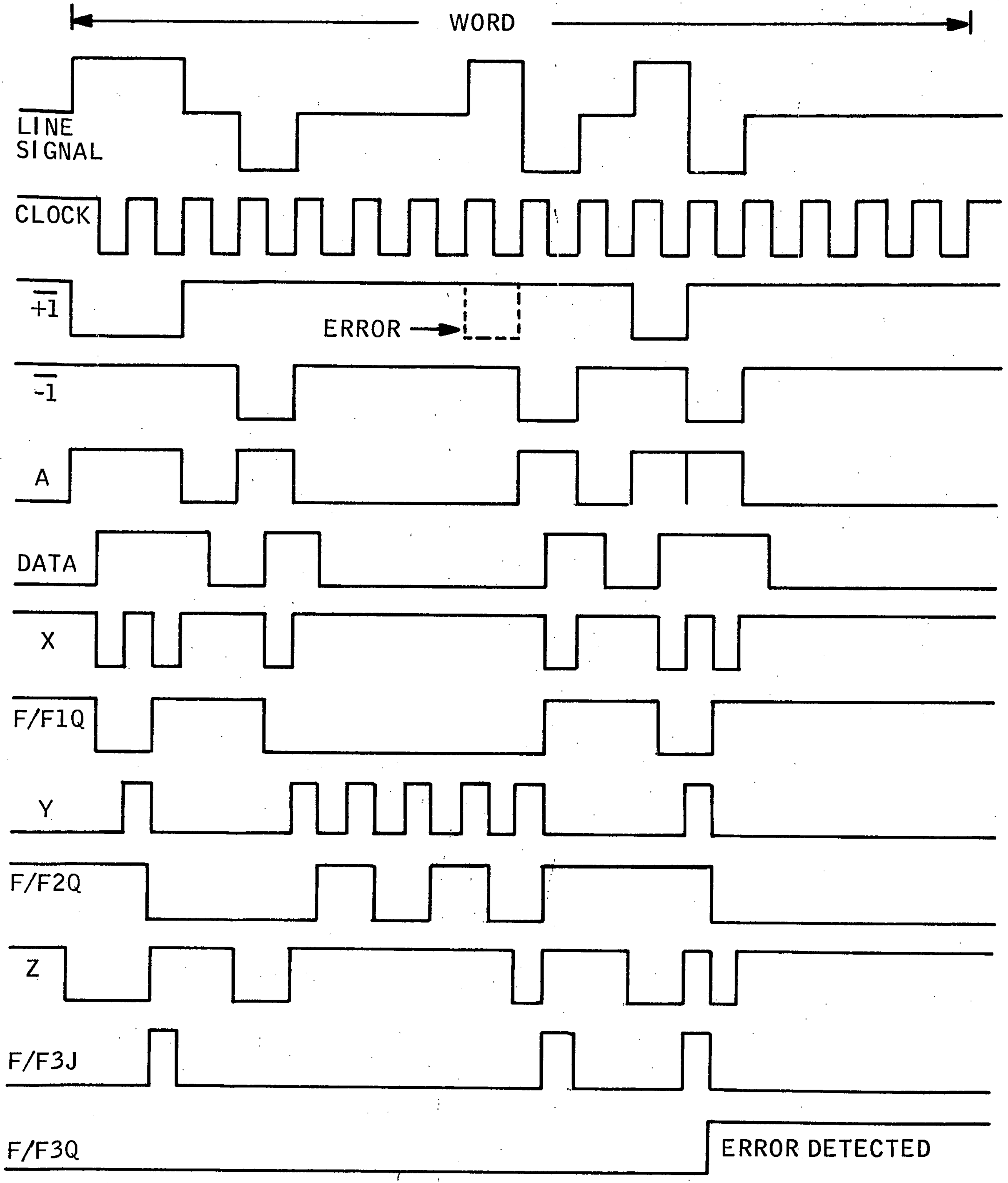


FIGURE A6-3
MD ERROR DETECTION TIMING DIAGRAM

This is expected as no MD rule is broken by the (incorrect) received pattern until bit 12. This is easily verified by considering the MD rules (see Part 3). The comments made above concerning the relative timing of the signal at F/F3J and the clock are applicable here. With these in mind it is seen that an error ID will occur as indicated in Figure A6-3.

Figure A6-4 is an additional example of the detector's performance. Bit 7 is assumed incorrectly received as a '-1'. The MD rules are first broken by bit 9 at which time an error ID is generated.

The detector described will detect all single bit errors which are followed by the correct reception of at least two consecutive '1's. Problems arise if consecutive '1's do not follow the reception error. As an example, the data pattern in Figure A6-5, erroneously received as indicated in the second line, does not provide a means of error detection. If two consecutive '1's are appended at the end of all transmission, errors of the above sort are detected. This is demonstrated in Figure A6-5.

The above technique will also detect most, but not all, two bit transmission errors.

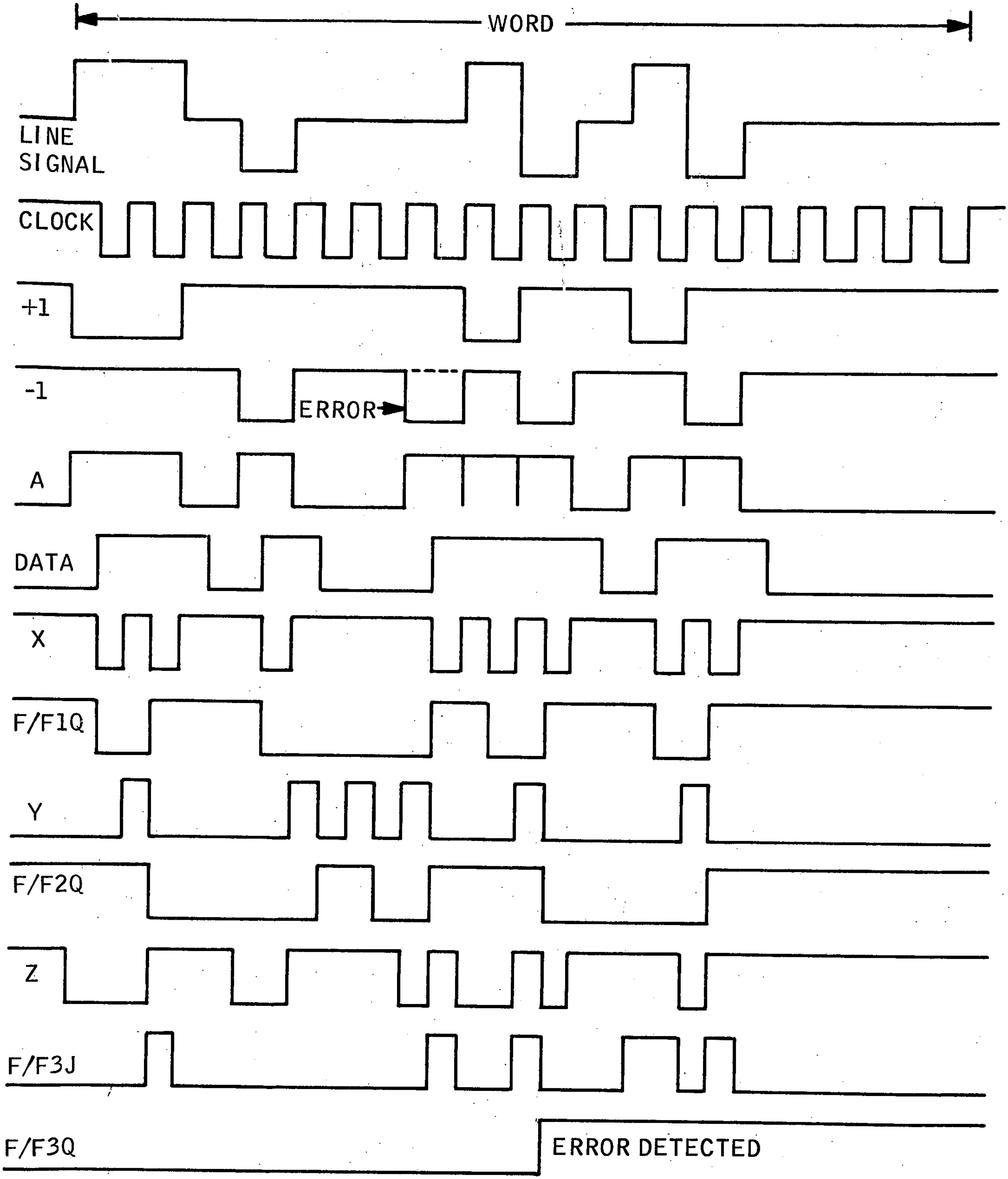


FIGURE A6-4
MD ERROR DETECTION TIMING DIAGRAM

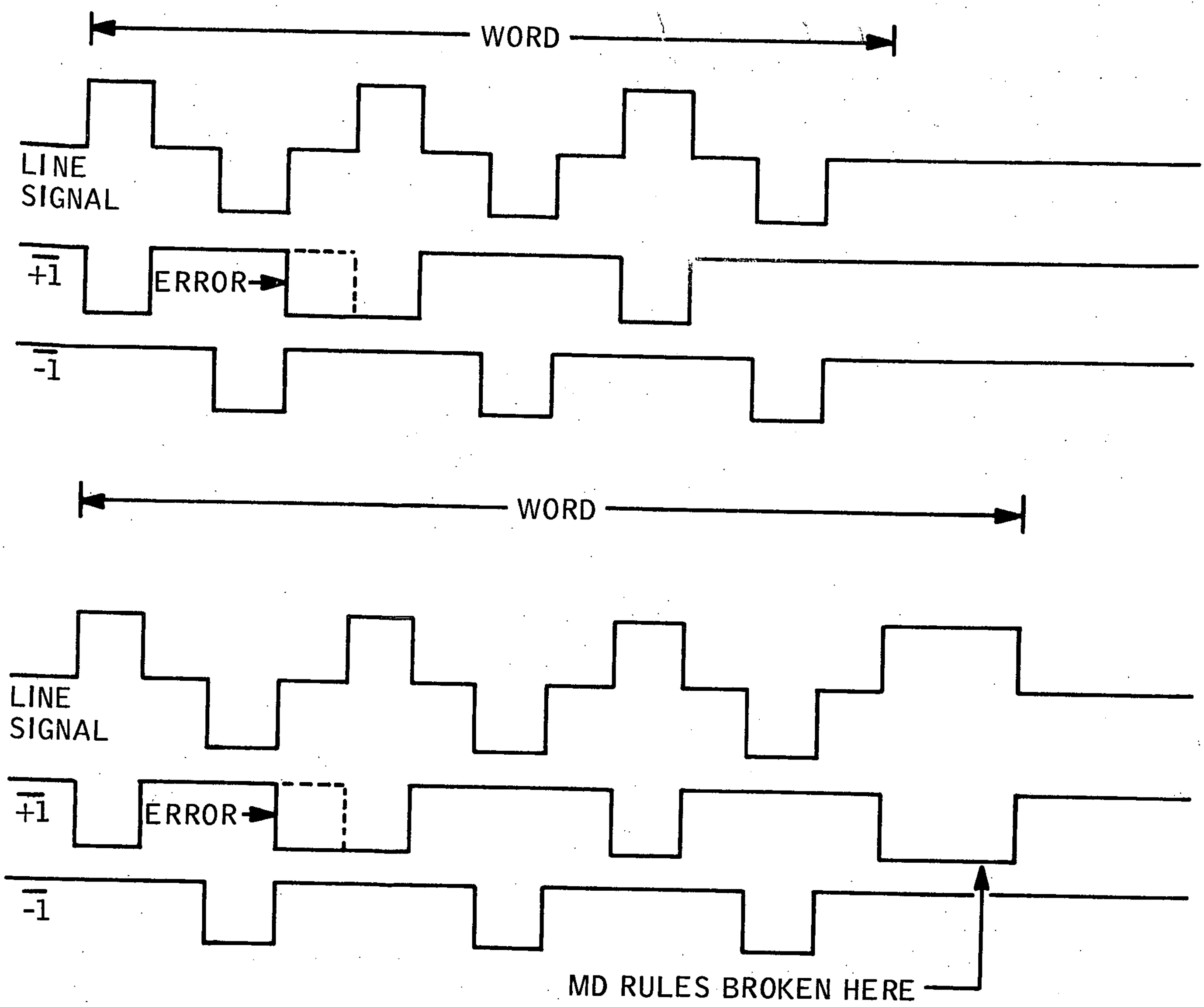


FIGURE A6-5
UNDETECTABLE ERROR PATTERN AND NECESSARY WORD MODIFICATION

A BCH code can be easily generated which is described by the following parameters:

Block Length: $n = 2^m - 1$, m integer

Number of check Bits: $r = n - k \leq mt$

Minimum Distance: $d \geq 2t + 1$

The BCH codes, being cyclic codes, are extremely easy to use, especially for error detection. A method of obtaining a generator polynomial of a BCH code to satisfy given block length and error detection capability requirements will be illustrated below. First, the logic implementation of BCH codes, given the generator polynomial, will be discussed.

Let the generator polynomial be represented as follows:

$$g(x) = 1 + g_1 x + g_2 x^2 + \ldots + g_{r-1} x^{r-1} + x^r$$

where g_i equals either a zero or a one. All generator polynomials will be of the above form, that is the coefficients of x^o and x^r are always one and the degree of the polynomial is r.

To encode the data systematically (that is into a block where the first k bits are the undisturbed data bits and the following r bits are the check bits, $n = r + k$) the circuit of Figure A6-6 is used. The circuit consists of r flip-flops, r (or fewer) exclusive -OR gates, 1 AND gate, r + 1 inverters and three NAND gates. The exclusive -OR gates associated with zero coefficients from the generator polynomial are not required and the output of the preceding flip-flop then directly feeds the next flip-flop.

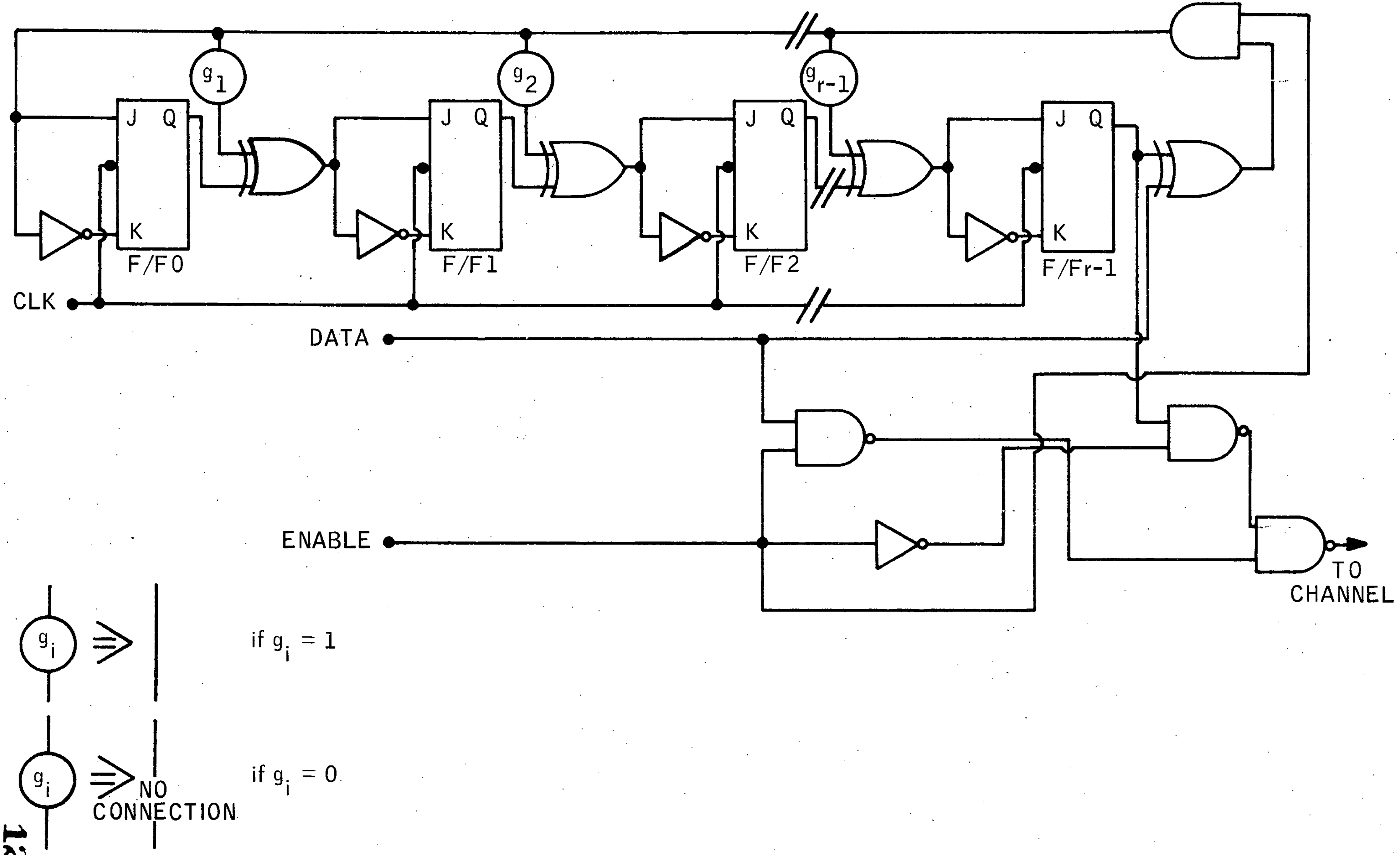


FIGURE A6-6
BCH ENCODER CIRCUIT

The circuit operation is as follows:

1. With the enable line high and the register originally cleared, clock the data into the channel and register simultaneously (k clock pulses).
2. With the enable low, shift the register contents into the channel (r clock pulses).

Figure A6-7 illustrates the encoder which results from the generator polynomial

$$g(x) = 1 + x^4 + x^6 + x^7 + x^8.$$

The coefficients of x, x^2, x^3, and x^5 are zero so that only 4 exclusive -OR gates are required.

Figure A6-8 shows the decoder required for error detection. The operation is as follows:

1. With the enable line high and the register cleared, the received word in systematic form is shifted into the register (n clock pulses).
2. With the enable line low, the register contents are shifted out (r clock pulses). If the register contains all zeros, no detectable errors occurred in transmission and the data is assumed valid. If the register contains one or more 1's, a detectable error has occurred and has been detected.

2a. If speed is important, the $\overline{Q}$ outputs from the flip-flops can be fed into an r-input NAND gate. At the end of step 1 above the NAND output indicates if an error has been detected as follows. If the gate output is high, a detectable error has been detected. If the

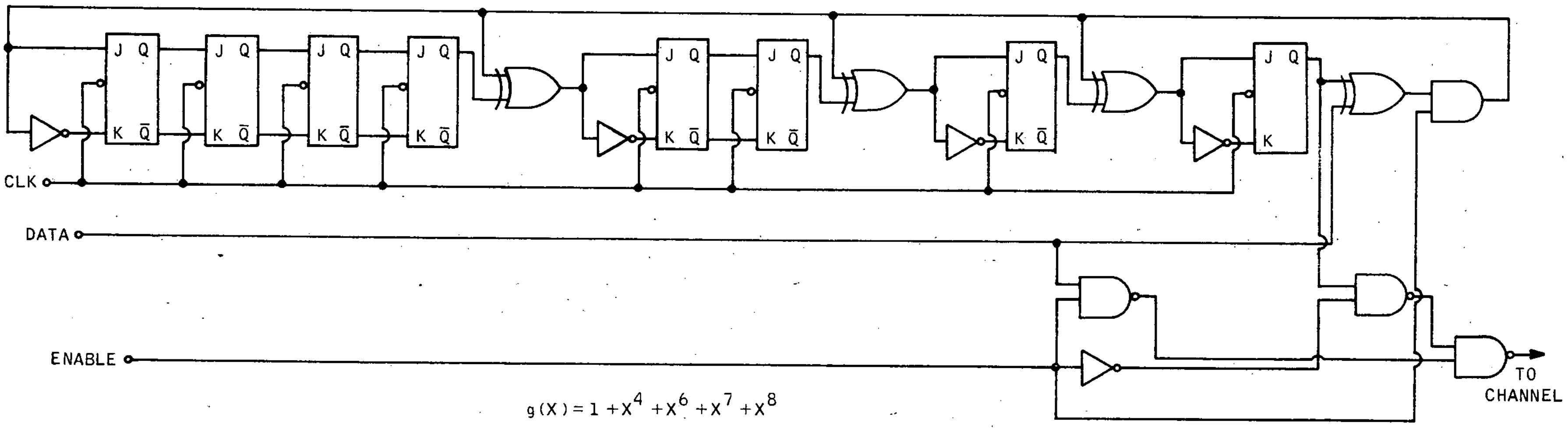


FIGURE A6-7

BCH ENCODER FOR $g(x) = 1 + x^4 + x^6 + x^7 + x^8$

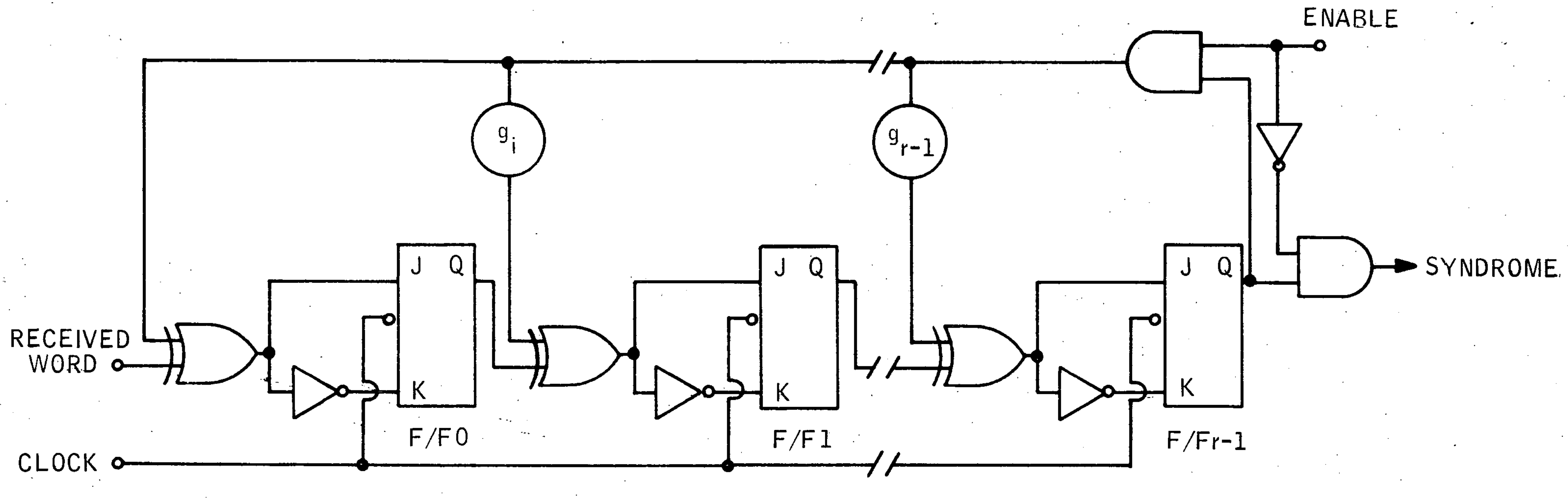


FIGURE A6-8
BCH ERROR DETECTING DECODER

gate output is low, no detectable errors have occurred and the received data is assumed valid.

A method of obtaining a $g(x)$ to satisfy given system requirements will now be discussed. This will be followed by several examples.

Formally, the $g(x)$ is obtained as follows:

Let α be a primitive element of the Galois Field of 2^m elements ($GF(2^m)$). Consider the sequence $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{2t}$ where t is defined above in the description of BCH codes. If $m_i(x)$ is the minimum polynomial of α^i, the generator polynomial is given as

$$g(x) = \text{LCM } (m_1(x), m_2(x), \ldots, m_{2t}(x)).$$

Fortunately, it is not necessary to understand the above derivation to use it! A major simplification is obtained by discarding the even numbered m_i's. This is allowable because of a property of Galois fields, namely that if

$$m(\beta) = 0$$

it follows that

$$m(\beta^{2\ell}) = 0, \ \ell = 1, 2, \ldots$$

Let i be an even integer. Then

$$i = i' \, 2^{\ell} ; \quad \ell \geq 1, \ i' \text{ odd.}$$

Because of the above property, factors appearing in $m_i(x)$ will be the same as those in some previous $m_{i'}(x)$. Since we are intqrested in the LCM (least common multiple) of the minimum polynomials it follows that

$$g(x) = \text{LCM } (m_1 (x), m_3 (x), \ldots, m_{2t-1} (x)).$$

It is interesting to note that the highest possible degree of $g(x)$ is mt (highest possible degree of $m_i(x)$ is m, there are at most t such factors comprising $g(x)$). The number of stages in the encoding shift register is therefore limited to mt or less so that r (the number of check bits) is also mt or less.

Tables of minimum polynomials are readily available. Peterson has a fairly complete table. Several examples using this table follow :

Example A

Suppose we desire a code with a minimum distance of 3 and capable of sending 11 data bits per block. A BCH code of m = 4 is seen to satisfy the requirements with

$$n = 15$$

$$r \leq 4$$

$$d \geq 3 \quad (t = 1)$$

The generator polynomial will be $g(x)=\text{LCM}\ (m_1\ (x)) = m_1\ (x)$.
Using Peterson's table, for m = 4, $g(x) = x^4 + x + 1$

Example B

Suppose we desire a minimum distance of 5 ($t = 2$) with the capability of 7 data bits. A BCH code with

$$n = 15$$

$$r \leq 8$$

$$d \geq 5$$

is seen to satisfy the requirements.

Therefore

$$g(x) = \text{LCM}\ (m_1\ (x), m_3\ (x)),$$

From the table,

$$m_1 = x^4 + x + 1$$

$$m_3 = x^4 + x^3 + x^2 + x + 1$$

or

$$g(x) = x^8 + x^7 + x^6 + x^4 + 1$$

Example C

Suppose we must transmit blocks of 30 data bits over a channel with raw bit error rate of 10^{-6} and be able to specify the probability of a undetected word error as 10^{-25}. A BCH code can be easily found as follows. First, the minimum distance necessary must be found. To do this, consider the probability of i errors in an n bit block over the specified channel:

$$Pi = \binom{n}{i} p^{n-i} q^i$$

where $p = 10^{-6}$, $q = 1 - 10^{-6}$, $\binom{n}{i}$ is the binomial coefficient. Guessing n to be 45 (15 check bits assumed), $P_5 \simeq 10^{-25}$, $P_6 << 10^{-25}$. Therefore a code capable of detecting all 5 (or fewer) bit errors in a block is sufficient ($d_{min} = 6$).

The smallest t necessary is therefore $t = 3$.

Now try $m = 5$ ($n = 31$).

$$r \leq 5 \cdot t = 15 \Rightarrow k = 16.$$

This is not satisfactory. Try $m = 6$ ($n = 63$).

$$r \leq 6 \cdot t = 18 \Rightarrow k = 45.$$

The first 15 data bits can be "sent" as zeros, resulting in an effective block length of 48, $d \geq 7$. The generator polynomial is therefore

$$g(x) = \text{LCM}\ (m_1(x), m_3(x), m_5(x))$$

For m = 6,

$m_1(x) = x^6 + x + 1$

$m_3\ (x) = x^6 + x^4 + x^2 + x + 1$

$m_5\ (x) = x^6 + x^5 + x^2 + x + 1$

Performing the indicated multiplication results in:

$g(x) = x^{18} + x^{17} + x^{16} + x^{15} + x^9 + x^7 + x^6 + x^3 + x^2 + x + 1.$

Part 7 - Conclusions and Recommendations

The purpose of this study was to investigate various techniques of data communication applicable to computer systems intercommunication, particularly in a random access data bus or party line configuration (see Figure A1-1). The desired result was a recommendation of a preferred technique for achieving such communication.

As previously pointed out, a major limitation to high speed data transfer in a data bus system is the low pass characteristic of the cable. Because of this consideration, various signal formats were investigated in terms of the bandwidth required for proper transmission.

Of the three standard approaches to digital modulation (ASK, PSK, FSK), binary PSK or split-∅-C offers ease of implementation and is certainly the best of the three for the intended application. The major disadvantage to split-∅-C is the large bandwidth required (10MHz bandwidth for a 5Mbps data rate, see Figure A3-5 and A3-6).

Multi-level signalling provides a means of achieving signals with arbitrarily small bandwidths (see Figure A3-10). These techniques were not pursued due to the relative complexity of the transceiver hardware required.

Modified duobinary (11) is an example of partial response (12) systems and offers several advantages for the intended application. The principle feature of the MD signal is the narrow bandwidth required (2.5MHz for a 5Mbps data rate, see Figure A3-23). By considering the impulse response of the MD filter (see Figure A3-24) and the effect of proper pre-coding on the waveform (see Figure A3-25 and 26), simple rules for generating a MD signal are obtained.

These rules can be implemented with standard logic elements to make a particularly simple transmitter. The circuit for this is shown in Figure A7-1 (14). This circuit produces MD waveforms without the two clock cycle delay inherent in the transmitter of Figure A5-2. The receiver of Figure A5-5 operates off the same transformer as the transmitter and is usable to distances of 2000 feet over BL-1244 (124 Ω Balanced Line) cable. This transceiver (Transmitter of Figure A7-1, Receiver of Figure A5-5) is recommended for data bus use at 5Mbps up to 2000 feet of cable.

The discussion in part 2 concerning simultaneous line accession by two or more stations needs to be modified for the recommended transceiver design. The MD transmitter presents effectively a zero source impedance during the transmission of logical ones, making it impossible to check for additional signals on the line. During the transmission of logical zeros, however, the source impedance is large as the transmitter is effectively off. This allows the receiver associated with the transmitter to listen for additional line signals during the transmission of a word. Two conditions must be met if interference is to be detectable and therefore ignored: 1) the word duration must exceed twice the time required for a signal to propagate the length of the cable, and 2) the transceiver must be designed so that the transmitter is shut down if interference is detected during a transmission. In this way, complete word transmissions are not possible if interference exists. Receivers can be designed to ignore incomplete transmissions. This approach, coupled with error control coding as described in part 6 should offer reliable data transfer. The reliability obtained will be largely a function of how sophisticated the coding system is.

The modified duobinary signal offers intriguing possibilities for further

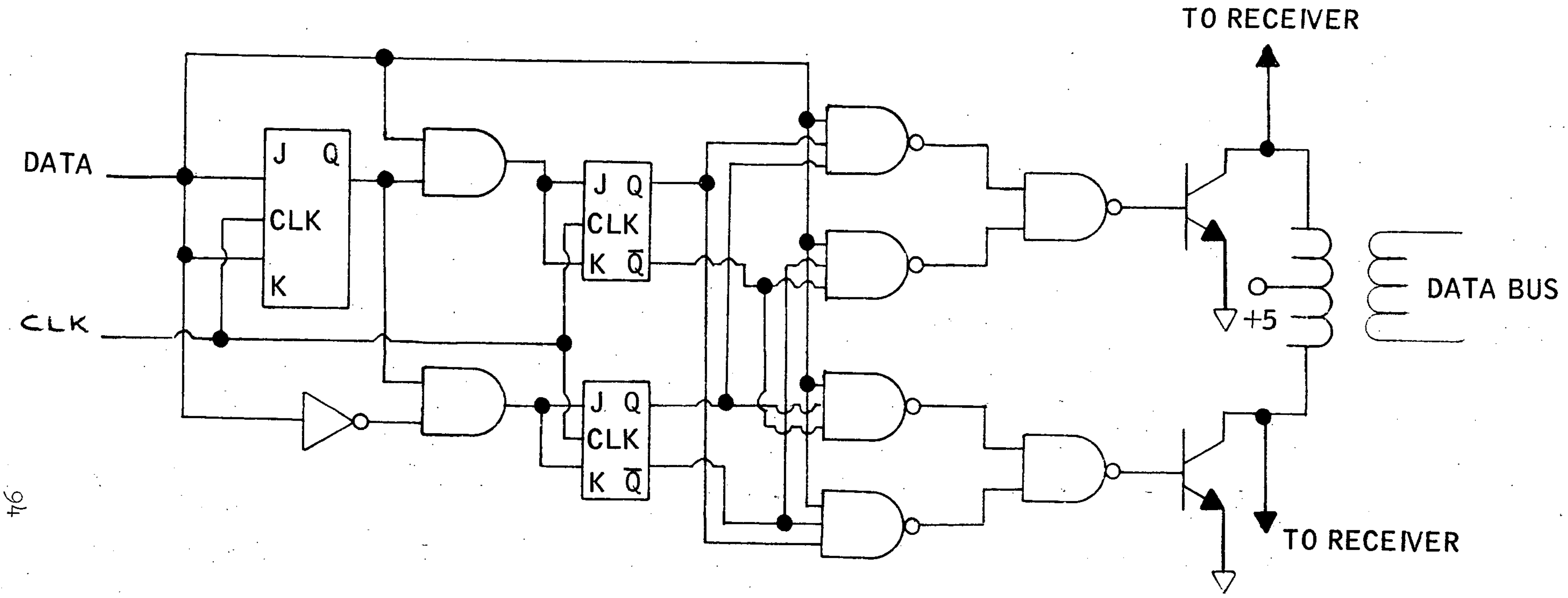


FIGURE A7-1
MD TRANSMITTER

development due to the low bandwidth required. For example, using SSB techniques two complete channels, each capable of a 5Mbps data rate, could be designed to occupy a 5MHz bandwidth. Each channel would be capable of independent, two-way communication on the bus. A pilot tone could be inserted at 2.5MHz to provide uniform timing to all the stations. Such an extension of the use of MD signals would provide an extremely flexible and reliable data bus system.

The transmitter of F.gure A7-1 can be easily modified to transmit logical ones as alternating polarity pulses. This results in a signal with a wider bandwidth than the MD format, but with slightly better DC balance. Over certain cable types (particularly long lengths of low bandwidth cable) the improved balance is a desirable feature. For example 15 , over Trompeter 78 balanced line the peak to peak amplitude after 1000 feet is reduced by this signal format but the balance between +1's and -1's is improved. This effect is not significant on the Timer BL-1244 cable at up to 2000 feet.

References

(1) Shannon, C. E., "Communication in the Presence of Noise", Proceedings of IRE, pp 10-21, January 1949.

(2) Arthurs, E. and H. Dym, "On the Optimum Detection of Digital Signals in the Presence of White Gaussian Noise - A Geometric Interpretation and a Study of Three Basic Data Transmission Systems", IRE Transactions on Communication Systems, pp 336-372, December 1962.

(3) Ziemer, R. E., "Character Error Probabilities for M-ary Signalling in Impulsive Noise Environments", IEEE Transactions on Communication Technology, Vol. COM-15, No. 1, pp 32-44.

(4) Nyquist, H., "Certain Factors Affecting Telegraph Speed", Bell System Technical Journal, pp 324-346.

(5) Shannon, C. E., "A Mathematical Theory of Communication, Bell System Technical Journal, Vol. 27, No. 3, pp 379-423 (Concluded on pages 623-656 or B. S. T. J., Vol, 27, No. 4), July 1948

(6) Slepian, David, "Bounds on Communication", Bell System Technical Journal, pp 681-707, May, 1963.

(7) Engel, J. S., "Digital Transmission in the Presence of Impulsive Noise", Bell System Technical Journal, pp 1699-1743, Octover 1965.

(8) Pierce, J. R., "Information Rate of a Coaxial Cable with Various Modulation Systems", Bell System Technical Journal, Vol. 45, No. 8 pp 1197-1207, October 1966.

(9) Jelonek, Z., "A Comparison of Transmission Systems", Transmission Systems and Coding, Chapter 3, pp 44-81.

(10) Mahoney, M. et. al., "Digital Communications", RCA Institutes, Third Edition, c 1968.

(11) Lender, A., "Correlative Level Coding for Binary - Data Transmission", IEEE Spectrum, pp 104-115, Feb., 1966.

(12) Kretzmer, E. R., "Generalization of a Technique for Binary Data Communication", IEEE Transations on Communication Technology, February 1966, pp 67-68.

(13) Lucky, R. W. et. al., Principles of Data Communication, New York, McGraw-Hill, c 1968.

(14) Trainor, W. F., Private Communication.

(15) Mallory, W., Private Communication.