# LOGIC DESCRIPTION

# DRUM STORAGE SYSTEM

C. V. Ravi

University of California, Berkeley

Document No. 20.70.30

Issued September 1, 1965

Contract SD-185

Office of Secretary of Defense

Advanced Research Projects Agency

Washington 25, D.C.

A. GENERAL:

The magnetic drum and its associated hardware form an essential part of the ARPA time-sharing system at Berkeley. The time-sharing is being done on an S.D.S. 930 computer--the hardware of which has been modified.

A block diagram of the whole system is shown in Fig. 1. It must be stressed here that the whole system is in the process of evolution and thus the block diagram represents only what has been developed so far.

The drum serves as a large capacity fast secondary storage medium and stores part of the executive program, the active programs, and data as well.

The main purpose of the hardware which will be described in this report is to allow records to be transferred between the core memory of the S.D.S. computer and the drum. The hardware allows information to be swapped between core and drum. Information can thus be stored on drum whenever it is not necessary for the information to be in core. When necessary this information can be moved back into core. Thus, the drum serves as a "scratch-pad" memory used to store information not always used--but information that is either used often or information to be used in the near future.

It is hoped that the time-sharing system will eventually have a core memory of continuously addressable 64K.*

---

*1K = 1024 words.

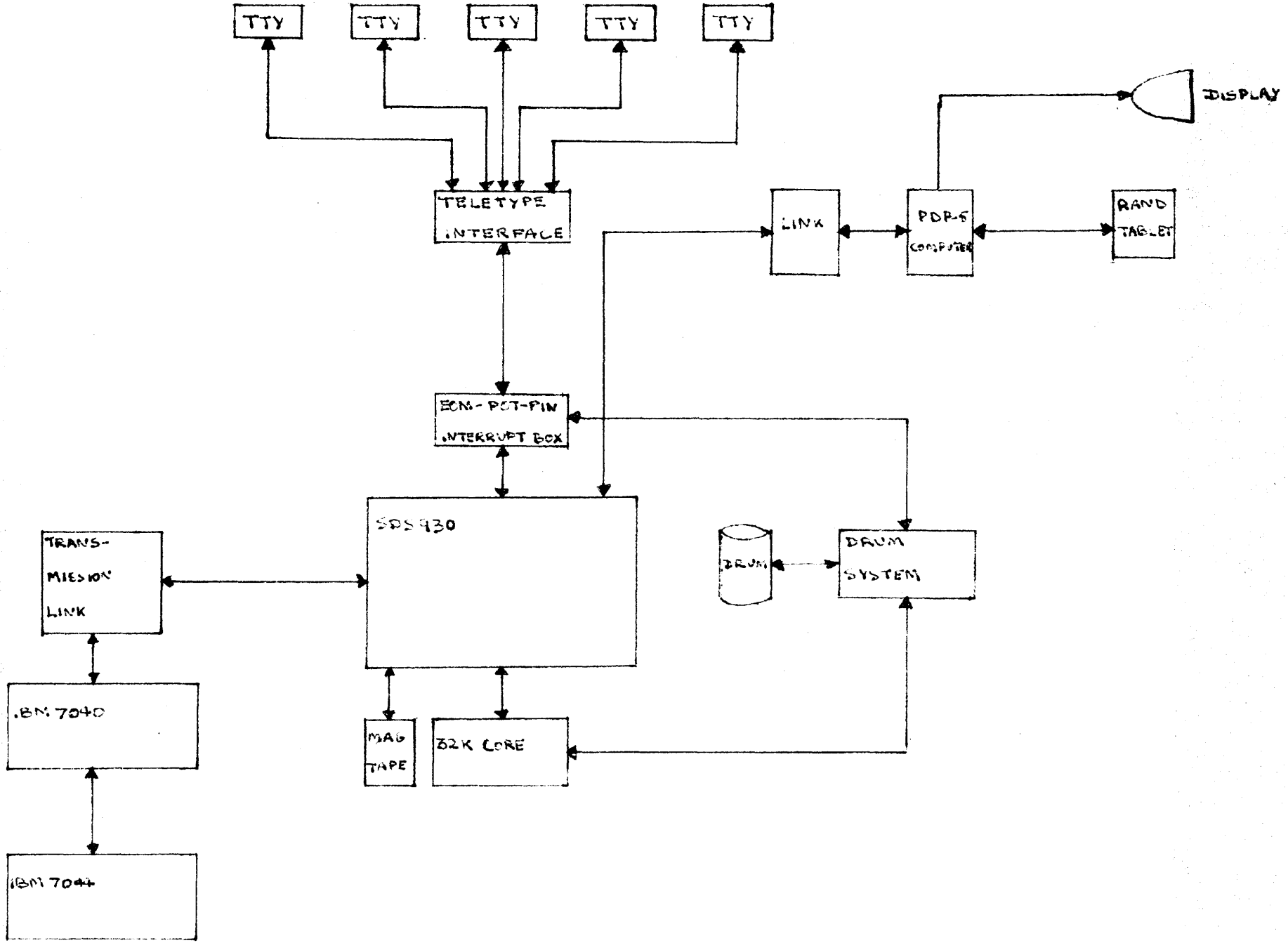

TTY
TTY
TTY
TTY
TTY
DISPLAY
TELETYPE INTERFACE
LINK
PDP-5 COMPUTER
RAND TABLET
ECM - PCT-PIN INTERRUPT BOX
SDS 930
TRANS-MIESION LINK
DRUM
DRUM SYSTEM
IBM 7040
MAG TAPE
32K CORE
IBM 7044

FIG. 1

The S.D.S. 930 computer does not normally allow addressing of more than 16K words. The hardware of the computer has thus been modified to allow the addressing of more than 16K words.

The S.D.S. 930 computer presently has a memory of 32K of core (see Fig. 2.). The memory consists of two modules--each of 16K--which can be accessed independently. Odd addresses pertain to one module and even addresses to the other module. As the stacks can be accessed independently, the drum system can access one module (when the drum system is in active status) while the computer references the other. If both wish to reference the same module, the computer reference is delayed by one cycle. It is hoped that during active status of the drum system, computer memory referencing will be 70 percent of that when the drum system is in wait status.

The drum system as it has been designed is capable of performing record transfers from or into 64K of core memory.

The S.D.S. 930 computer has a memory cycle time of 2 microseconds. The bit density, the speed of rotation of the drum, and the format of recording are such that a transfer rate of 500,000 words per second results. Also, the computer clock is driven from pulses recorded on the drum. Thus, the drum system is compatible with the computer.

It must be mentioned here that, in the present system, real core-swapping does not take place. That is, from the point of view of the drum system, simultaneous read and write on

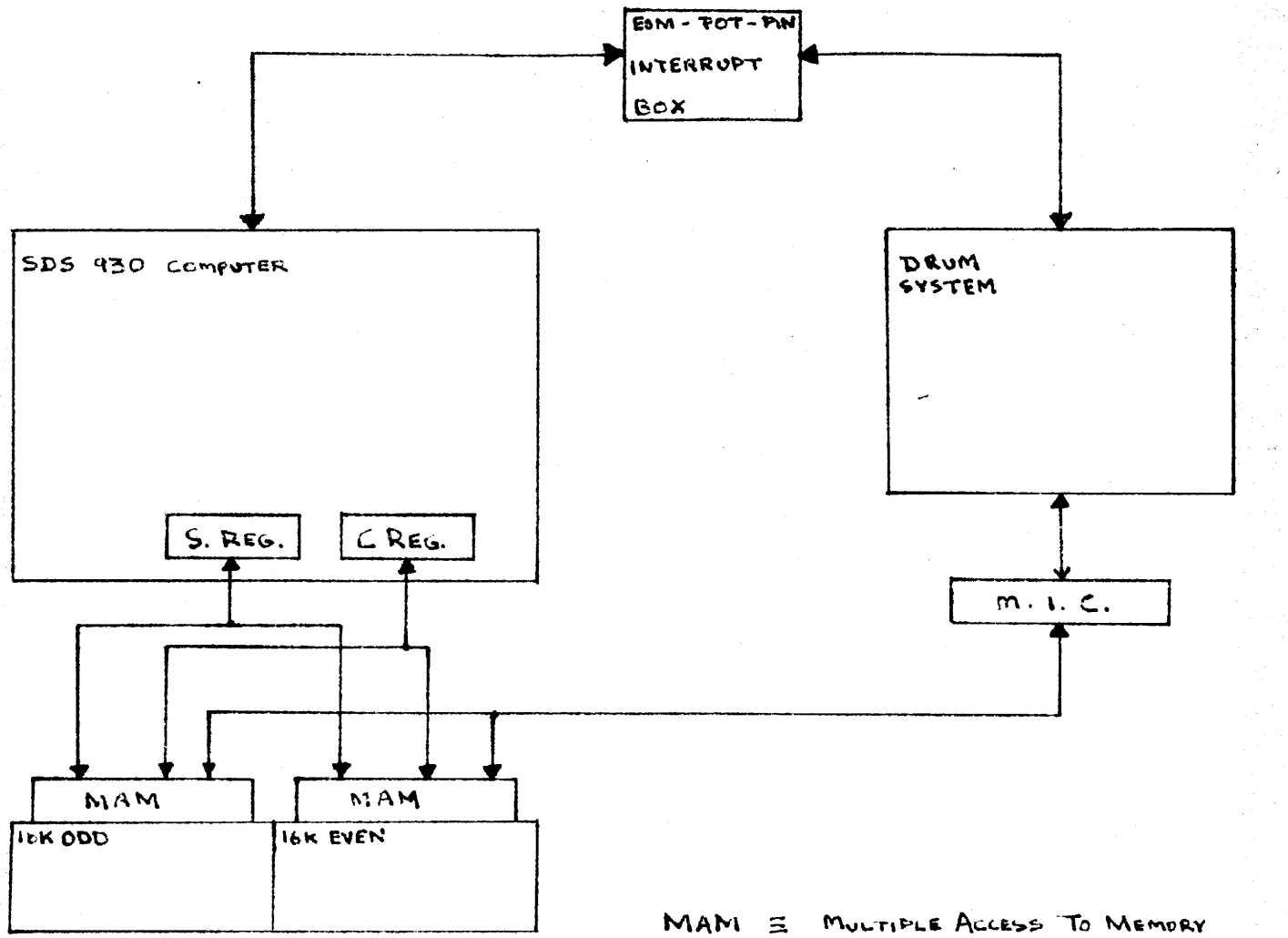

EOM - POT - PW
INTERRUPT
BOX
SDS 930 COMPUTER
S. REG.
C REG.
DRUM
SYSTEM
m. i. c.
MAM
MAM
16K ODD
16K EVEN
MAM ≡ MULTIPLE ACCESS TO MEMORY
MIC ≡ MEMORY INTERFACE CONNECTION

FIG. 2

different parts of the drum are not possible. The reason for this is that simultaneous read/write would involve duplication of many registers--which would increase the complexity and cost of the system. Program complexity would also be increased.

A list of commands for the drum is stored in the memory of the computer. The commands in the list are then executed by the drum system. These commands make the drum system disconnect, or perform record transfers, or branch to a new command list, or interrupt the computer. The branch to a new command list operation requires a little explanation. The command lists are normally stored in adjacent, sequentially addressable core memory locations. However, this is not necessary as with a DCBR command, command lists can be placed in non-sequentially addressed locations.

The drum system is almost completely autonomous and executes the command lists independently, once operations have been initiated; that is, the computer does not have to be idle during active status of the drum system.

Also partial records can be read depending on certain conditions.[1]

B. DRUM DESCRIPTION:

The drum was manufactured by Vermont Research Corporation. It has a storage capacity of 1.3 million, 24-bit words.

---

[1]"Drum storage preliminary reference manual," Michael G. Hurley. Document No. 20.70.20. Issued March 30th, 1965.

The nominal speed of rotation is 1800 R.P.M. The data is stored on 84 drum bands--each band consisting of 12 tracks. In addition to these tracks there are 3 clock tracks. One of these tracks has pulses recorded at one microsecond (nominal) intervals. Another contains just one origin pulse. The third contains pulses recorded at 11/8 megacycle (nominal) intervals.

The frequency obtained from the third track will be multiplied by 4 to generate the computer clock. Thus confusion will not result from the drum speed changing due to changes in line frequency or voltage.

When data is written, the 24-bit S.D.S. 930 word is broken into two 12 bit half-words, and each half-word is written in parallel. The second half-word follows the first half-word after one bit time.

Each band is divided into 8 equal parts called sectors. Each sector is again divided into two blocks--the data block and the gap block. Data is written only on the 8 data blocks in each band. Nothing other than a parity half-word is written on each of the 8 gap blocks within a band. Each data block is 2K bits long. The length of the gap is 236 words.

This configuration of the magnetic surface allows 16K words of data to be written on each band of the drum.

The drum is continuously addressable and, as far as the programmer is concerned, the gaps need not exist--save for one point. For every record written on the drum, a guard word is written on the location following the last location of the

record. This guard word consists of two half-words. The first half-word indicates the parity of the record written on the drum while the second half-word serves as a guard band between records. Thus, if a record ends on any location on a data block other than the last location, an addressable memory location is lost. However, if the record ends on the last address of a data block; that is, on an address that is (n . 2k-1) where n = 1, 2, 3, 4, 5, 6, 7, 8.

The guard word is written in the first location in the gap. Thus, in this case, an addressable memory location is not lost. From the previous discussion, it is obvious that the gap blocks on the drum are not addressable. It should be mentioned here that if a record is longer than 2K words; that is, when it effectively runs across a gap, a guard word is not only written at the end of the record but also in first location/locations of the gap/gaps across which the record runs.

The parity half-word is generated in the drum system when data is being written on the drum. This is the half-word which constitutes the first half of the guard word. This is checked with parity generated when the information is being read from the drum. If an error occurs, the error flip-flop is set (see Fig. 2.).

The gaps not only serve to make the transfer rate compatible with the computer, but also allow head switching between bands to take place (without the loss of one rotation

when a record is written on two or more different bands). During a record transfer, head switching is done automatically whenever necessary.

C. DRUM SYSTEM DESCRIPTION:

The drum system consists of the magnetic drum and associated hardware. The hardware is discussed in this section. For this section refer to Fig. 3.

The hardware basically consists of

a) The DAT register:

The data register serves as an exchange register between the computer and the drum system. It is a buffer between the M.I.C. (Memory Interface Connection) and the drum system.

On input to the drum system, information from the core memory of the computer is strobed into the data register. If this information consists of commands, the drum system is set up to perform what the commands specify. If the information consists of a record or records (data) to be written on the drum, the contents of the DAT register are transferred to the RW register 12 bits at a time. Needless to say, data is preceded by commands.

In the case of output from the drum system, information is put into the DAT register by the rest of the drum system. This is then strobed into the computer memory via the M.I.C. The information may consist of words indicating the status of a certain counter or flip-flop in the system (see D), or of

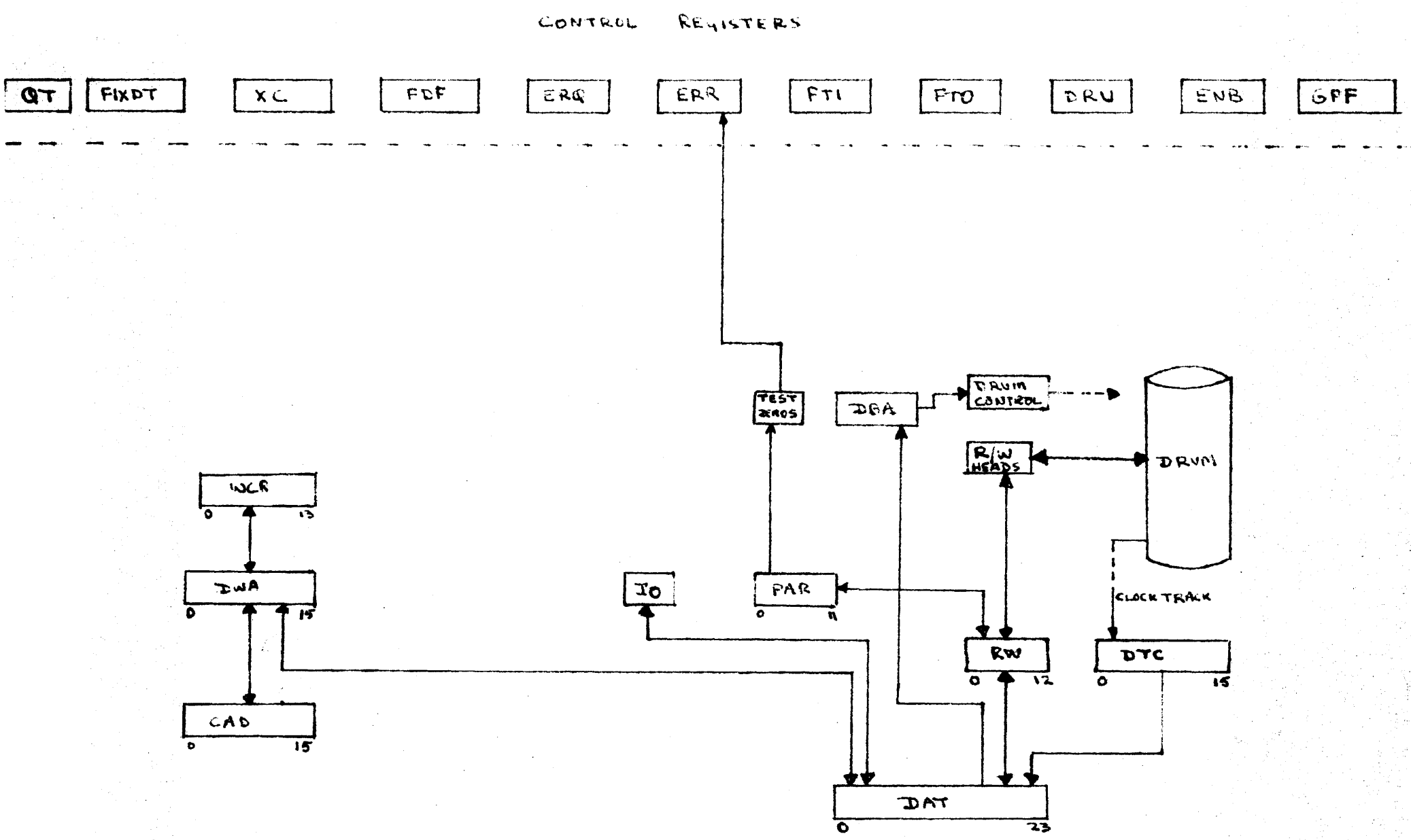

CONTROL REGISTERS
QT
FIXPT
XC
FDF
ERQ
ERR
FTI
FTO
DRU
ENB
GPF
TEST ZROS
DBA
DRUM CONTROL
R/W HEADS
DRUM
WCR
0
13
DWA
0
15
CAD
0
15
IO
PAR
0
11
CLOCK TRACK
RW
0
12
DTC
0
15
DAT
0
23

FIG. 3

data which has to be transferred into the core memory. In the latter case, data is transferred from the RW register into the DAT register 12 bits at a time.

The data register is 24 bits long.

b) The RW register:

The Read/Write register serves as a buffer between the drum and the data register. A word, as explained previously, is recorded on the drum in two 12 bit half-words, each half-word in parallel, the second half-word being recorded one bit time later than the first half-word. Thus, when words have to be written or read into or from the drum, the appropriate heads on the drum are energized and connected to the RW register.

c) The CAD register:

The core address register is a sort of remote memory address register for the core memory of the computer: that is it contains the address of the core location into/from which information must be written/read.

The contents of this register are incremented in DWA by one every word time during a record transfer.

The register is 16 bits long--allowing 64K of core to be addressed.

d) The IO register:

The input-output register in reality is just one flip-flop, the status of which indicates the direction of record

transfer (when a record transfer is to be or being done). It is also used for control purposes.

IO = 1. Transfer from core to drum

IO = 0. Transfer from drum to core

IO controls the access to core and drum.

e) The WCR register:

This register is 14 bits long. The word length of the record to be transferred is stored in this register. This register is counted down in the DWA register every time a word has been transferred.

The contents of this register are tested to know when to conclude the record transfer.

f) The DWA register:

This register is a counter and is one of the most important parts of the system. The drum word address register initially stores the starting address on the drum of the record to be transferred.

However, once the transfer of a record begins, it serves to increment the contents of the word count register, (as the contents of the word count register are in complement form, the word count is essentially counted down), and the CAD register.

This register is 16 bits long because the CAD register is 16 bits long.

g) The PAR register:

The parity register is responsible for the generation

of parity. Parity is generated both when information is being written on the drum and when information is read off the drum. Every half-word "passing through" the RW register is checked for odd parity.

The parity register is 12 bits long--thus being the same length as the RW register. The parity generation is straightforward except for one point. The contents of the PAR register are ring shifted once after each half-word in the RW register has been checked for parity. This is just a design detail with one specific purpose. The bit density on the drum is more than the track density and a dust particle is more likely to cause two consecutive errors along a track than on adjacent tracks. Parity checking for error fails when there are two (or multiples of two) errors. The ring shifting helps to alleviate the problem considerably.

During a core-to-drum transfer, the half-word of parity, as mentioned earlier, is written on the drum. When the record is read back later during a drum-to-core transfer, parity is again generated. This is compared with the half-word written earlier.

h) The ERR register:

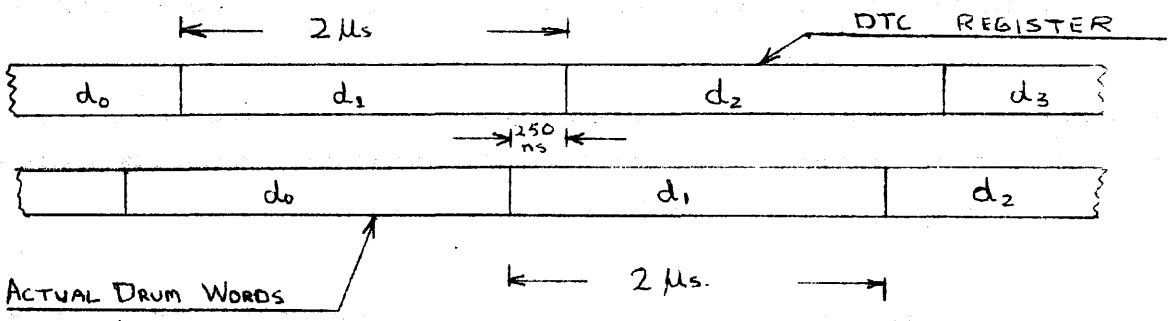
This also, like the IO register, is just one flip-flop. It is set whenever a parity error is detected during read-out. This flip-flop, when in the set state, can be reset with a DCRP command or a Reset Drum System Computer instruction.

i) The DBA register:

The drum band address register selects the drum band. This register is also a counter and is incremented during the transfer of a record, if the record continues past the data block in the last sector of one band into the data blocks of the sectors in the next band.

j) The DTC register:

This register is a counter. The drum timing counter is a 500 kilocycle counter. It is incremented by the QT counter (see p. 12 ) and is phased ahead of the drum by 1.75 microseconds (see Fig. below).




The significance of the contents of the DTC register follows:

1) Bits 0-2 indicate the sector (0-7) under the read/write heads.

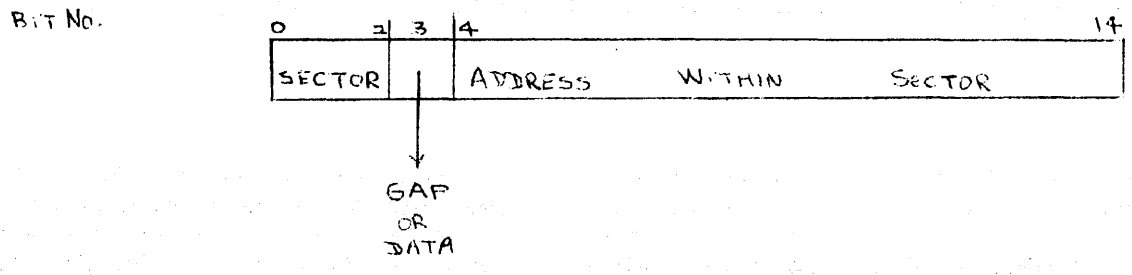
2) Bit 3 indicates whether the heads are above a data block or a gap block within a sector.

3) Bits 4-14 indicate the address of the cell under the read/write heads. These bits are all zeroes at the beginning of a data block and at the end of the data block, they become all ones. The length of the gap is 236 words. Thus, at the beginning of a gap block the register is set so that these bits indicate the binary equivalent of (2K - 236). At the end of the gap block, they become all ones.

NOTE:

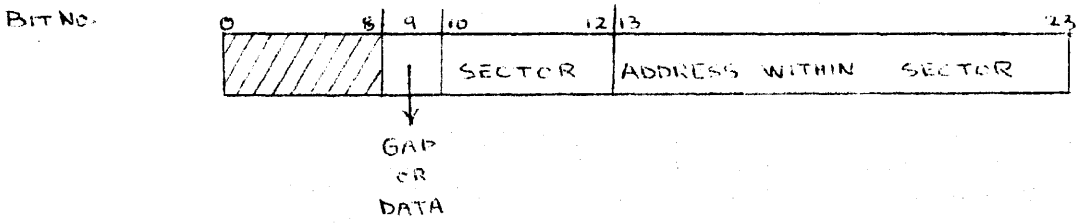
In the programmers reference manual, it is mentioned that the status of the DTC register is placed in the DAT register when an EOM 00230024 instruction is issued. This has a different format from the DTC register. This was done in order to make it easier for the programmer (see Fig. below).

Format of DTC register



Format of DAT register after

EOM 00230024 instruction




The DTC counter as can be seen above is 15 bits long.

It should be mentioned here that, during the initial stages of the transfer operation, these registers may hold other data. This is explained in the next section.

In addition to these registers, there are others essential for the logical sequencing of operations.

These are:

a) The XC counter:

This is the main sequencing unit. System operation is broken up into "states." This counter keeps track of the states.

b) The QT register:

The timing of the logical operations is derived from this 4-megacycle shift register. It is incremented by the following procedure. The frequency of pulses recorded on the drum is doubled twice (to obtain 4-megacycles) and then these pulses are fed into the QT register. It is also synchronized once every rotation with the origin pulse on

the drum.

c) The ERQ flip-flop:

This is for the convenience for writing logic equations. It is set when memory is accessed.

d) The DRU flip-flop:

The Read/Write heads on the drum are energized when this flip-flop is set.

e) The FTI and FTO flip-flops:

These flip-flops control the transfer of information in and out of the DWA register respectively.

f) The FDF flip-flop:

Due to the slow rise time of buffers following the circuits that decode the XC counter to determine the state the system is in, operations cannot be performed reliably during the first bit time of a state.

The FDF flip-flop is used in such a way that operations <u>can</u> be performed during the first bit time of a state.

g) The ENB flip-flop:

During the debugging of the system, a problem that occurred again and again was the rippling of carries wnenever a counter was cleared .

The ENB flip-flop (Enable flip-flop) helped in solving this problem. Carries are allowed to ripple down only when this flip-flop is high.

It was found that ENB had to be dropped some time before the register (DWA) could be cleared.

D. DRUM SYSTEM OPERATION:

The operation of the drum system is naturally dependent on the commands given by the programmer. However, once the commands are given, the operation can be divided into three phases i.e.

a) The control phase

b) The core to drum transfer phase

c) The drum to core transfer phase

a) The control phase:

The control phase is the phase in which commands are recognized and acted upon. A discussion of the basic commands from the hardware point of view follows:

Commands are of three types--Computer instructions, Computer test instructions, and Drum commands.

Computer instructions:

1) When the drum system is in wait status, and an EOM 00230024 instruction is issued by the computer, the contents of the DTC counter are placed in core location 21.

i.e. 21 ——————→ CAD

(DTC) ——————→ DAT

(DAT) ——————→ core location 21

2) When the drum system is in wait status and an EOM 00230004 instruction is issued by the computer, the drum system is energized and the address field of the contents of core location 20 is used as a pointer to the command list in memory. Thus,

to activate the drum system, this command is necessary.

i.e. 20 ⟶ CAD

(20) ⟶ DAT

bits 9-23 of DAT ⟶ bits 1-15 of CAD

(L.A.) ⟶ DAT

where L.A. (list address) ≡ bits 1-15 of CAD

After this, the operations depend on the (L.A.).

3) When the computer issues an EOM 00230044 instruction, all operations are terminated and the drum system returns to wait status. The ERR flip-flop is turned off.

Computer Test Instructions:

1) With an SKS 04030004 instruction, the computer skips the next instruction and executes the following instruction if the drum system is in wait status. The next instruction is executed by the computer if the drum is in active status.

2) With an SKS 04030024 instruction, the next instruction is skipped and the following instruction executed if ERR is not set.

If ERR is set, the next instruction is executed.

The above two instructions don't involve anything special from the point of view of hardware. However, they are presented here for the sake of completeness.

Drum Commands:

The first 3 bits of the (L.A.) i.e. the 3 most significant bits are examined and depending on the combination,

commands are executed.

There are 8 possible combinations of 3 bits. Seven of these are recognized.

When bits 0-2 of (L.A.) are

1) 000: DCTX (Drum Channel Transmit)

This is recognized as a drum channel transmit command which consists of three words. These 3 words, from L.A., L.A.+ 1, L.A.+ 2 are read into the drum system and L.A.+ 3 is read back into location 20 of core.

i.e. (20) ≡ L.A.

(L.A.) ⟶ Drum System

(L.A. + 1) ⟶ Drum System

(L.A. + 2) ⟶ Drum System

L.A. + 3 ⟶ 20

Thus, by reading (20), the computer can monitor the progress of the drum system. As explained in the programmers manual, the three words contain data pertaining to the type of command, the word length of the record to be transferred (read into the WCR register), the direction of transfer (read into IO), the starting address in core (read into the CAD register), the drum band address (read into the DBA register), and the drum word address (read into the DWA register). The last two define the starting location on drum.

It must be emphasized here that the above discussion refers to the ultimate destination of the data, and registers

may hold other data temporarily.

After the information of the first 3 words has been transferred into the appropriate registers, the operation of the drum system enters phase b) or c) depending on the status of the IO register.

Phase b)

If IO = 1, operation enters phase b).

In this phase, a transfer from core to drum is made. The record is read from sequential core locations and written onto the drum. The following paragraph holds good for phase c) also.

As soon as a COMPR (compare) is detected i.e. the heads are just ahead of the starting location on drum ((DWA) = (DTC)), this is "remembered" by the XC counter and the (DWA) are destroyed. After this, the DWA register increments the contents of the CAD register and counts down the contents of the WCR register. The status of the contents of WCR determines the end of transfer i.e. when the word count is effectively zero.

Parity is generated and written on the drum.

Phase c)

If IO = 0, the operation enters phase c).

Information is read off the drum and written into the appropriate core locations.

Parity is generated and checked with the half-word

previously written on the drum. If an error is detected, the ERR flip-flop is set.

2) 001:- DCCT (Drum Channel Conditional Transmit)

This is the same as 1) if the ERR flip-flop is off. If ERR is set, the drum system is disconnected and an interrupt signal is issued to the computer.

3) 010:- DCBR (Drum Channel Branch)

With this command, the next command is not read from L.A. 3 but from the address field of the command. This allows command lists to be scattered all over the core memory--if necessary.

4) 011:- DCRP (Drum Channel Reset Parity Indicator)

The ERR flip-flop is reset.

5) 100:- DCDX (Drum Channel Disconnect)

The drum channel returns to wait status.

6) 111:- DCDI (Drum Channel Disconnect and Interrupt)

The drum channel returns to wait status and issues an interrupt signal to the computer.

7) 101:- DCDC (Drum Channel Disconnect and Interrupt if parity Indicator Set)

The drum channel returns to wait status and if ERR is set, issues an interrupt signal to the computer.

Thus from the above, it can be seen that phases b) or

c) can occur only if the first 3 bits of ((20)) are 000 or with ERR reset and the first 3 bits of ((20)) being 001.