# Burroughs
# B6500
## Data Communications Processor
### REFERENCE MANUAL

# Burroughs
## B 6500

# DATA COMMUNICATIONS PROCESSOR

## REFERENCE MANUAL

**B**

### Burroughs Corporation
Detroit, Michigan 48232

$5.00

TABLE OF CONTENTS

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

# TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

# TABLE OF CONTENTS (cont)

# LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (cont)

LIST OF TABLES

# INTRODUCTION

The Burroughs Data Communications Processor (DCP) is the heart of the data communications network. Four of these processors may be connected to each I/O Multiplexor; with each processor able to communicate with up to 256 remote lines. The DCP is programmable and contains the necessary registers and circuits to carry on such communications functions as polling, identification of control codes and delimiters, and all other line-checking functions, independently of the main processor. The modular design of the data communications network allows each line to be adapted to serve virtually any data communications device. The programmable nature of the DCP also allows generalized adapters to be used so that any remote device by a given data set may be handled over one line.

Because the Burroughs B 6500 system is designed for continuous multiprocessing and time sharing, it readily accommodates applications and procedures requiring data communications; real-time operations, remote computing, remote inquiry, and on-line programing merely become additions to the multiprocessing mix.

This manual provides a hardware description of the Data Communications Processor, the Adapter Clusters and the Line Adapters that are used to configure a data communications network.

The software used with the DCP is covered in separate manuals which include:

Burroughs B 6500 Data Communications Functional Description.

Burroughs B 6500 Data Communications Network Definition Language Reference Manual.

Burroughs B 6500 Data Communications Message Control System Reference Manual.

SECTION 1

B 6500

DATA COMMUNICATIONS SYSTEM CONFIGURATION

GENERAL

The B 6500 Data Communications System consists of a combination of the following units:

    a.   Data Communications Processor.

    b.   Adapter Clusters.

    c.   Line Adapters.

DATA COMMUNICATIONS PROCESSOR (DCP).

The Data Communications Processor (DCP) is a small special purpose computer that contains sufficient registers and logic to perform all basic functions associated with the sending and receiving of data. Up to four DCP's can be connected to an I/O Multiplexor, with each DCP capable of accommodating from one to 256 communications lines (figure 1-1). On a two-multiplexor system, this provides a B 6500 with the ability to service 2048 communications lines.

The DCP's gain access to main memory through the word interfaces on the I/O multiplexor. All the DCP's on a single multiplexor share that multiplexor's memory bus with the multiplexor. Only the multiplexor or one DCP can use this memory bus at a time, but this use is interleaved automatically to prevent any one of these devices from obtaining exclusive use of the memory bus.

ADAPTER CLUSTER.

The Adapter Cluster services a heterogeneous group of line adapters. Up to 16 Adapter Clusters can be connected to one Data Communications Processor. The same Adapter Cluster can be connected to two Data Communications Processors when it is desirable to have exchange facilities between the clusters and the DCP's.

As stated previously, the B 6500 Data Communications System can accommodate up to 2048 communications lines; however, when servicing this number of lines, the Adapter Cluster cannot be connected to
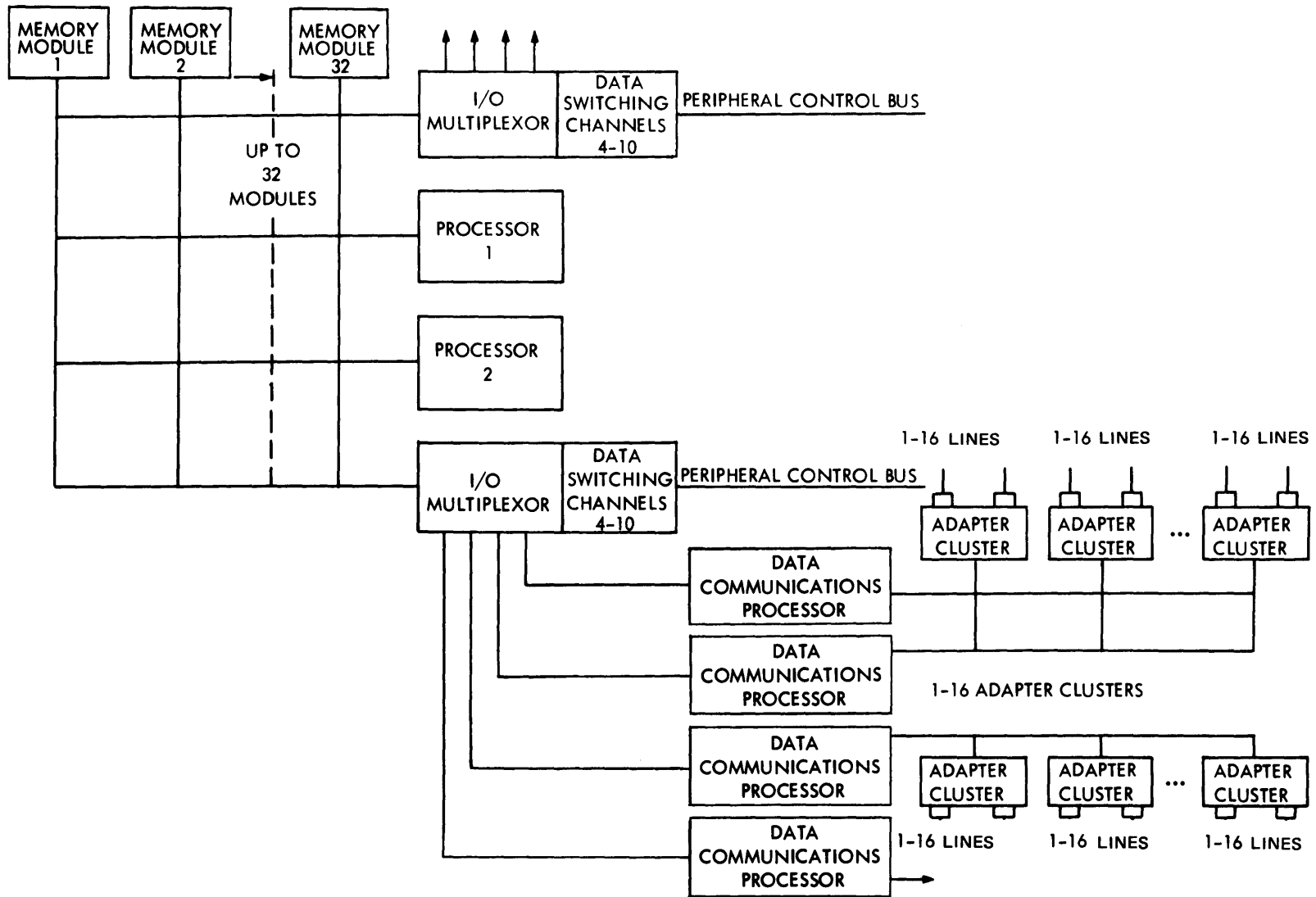
Figure 1-1. B 6500 System Configuration Including Data Communications

two DCP's.

## LINE ADAPTERS.

Up to 16 line adapters can be placed in each Adapter Cluster. Each communications line requires one line adapter; some lines or terminals may require two, i.e., full duplex terminals.

Line adapters are available to interface to the following data sets (using the RS 232 interface), line types, or terminals:

    WE 103A, 103F 108
    WE 202G, 202D
    WE 201A3, 201B
    WE 801 A and C Series
    WE 403E3/403D5
    Direct connection to WE teletypes, Models 28, 33, 35, and 37
    Direct connection to Burroughs units

The 16 line adapters are connected into the backplane of their associated Adapter Cluster.

## OPERATION.

The DCP executes instructions residing in its optional local memory or in B 6500 main memory (MM). These instructions monitor the Adapter Clusters and the B 6500 Processors for required attention. They cause data, control requests, and information to be passed to and from both main memory and the Adapter Clusters.

These instructions are also used to set up to control information to tell the cluster what is to be done; i.e., write data, accept data, answer an incoming call, etc. This technique allows the DCP to handle the different line control disciplines, speeds, and character formats while requiring only a limited number of general adapter types. These different adapter types are primarily for the different electrical interfaces.

5×14 3 PANEL A
5×14 2 PANEL A
5×14 1 PANEL A
5×14 0 PANEL A
5×36 PNL. B
5×14 4 PANEL A
5×14 5 PANEL A
5×14 6 PANEL A
5×14 7 PANEL A

SIDE SKINS
(8) REF.

PIN SIDE

CARD SIDE

DATA COMM. PROC.

ADAPTER CLUSTERS
(1 MIN., 8 MAX.)

POWER
SUPPLY

LOGIC
AREA
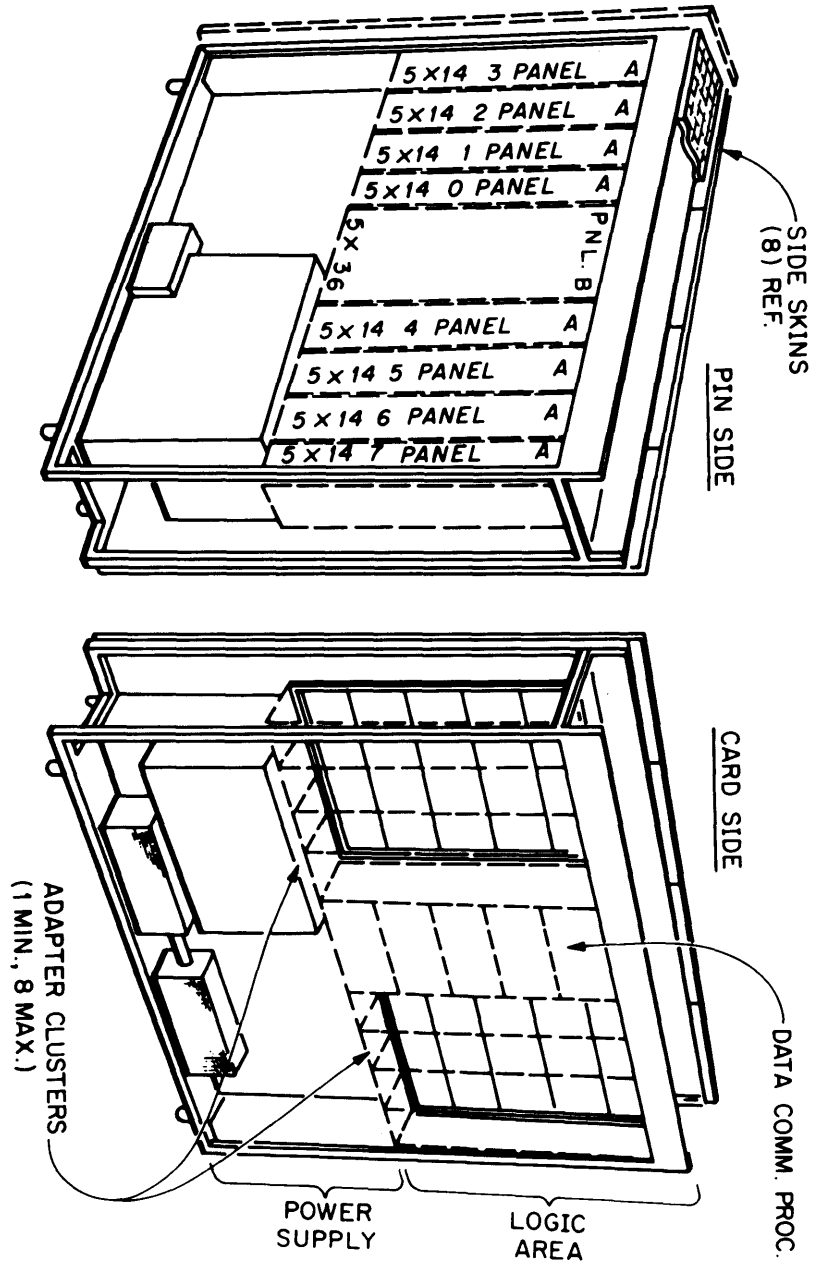
Figure 1-2.    B 6500 Data Communications Cabinet Type A

DATA COMMUNICATIONS CABINET.

One Data Communications Processor and eight Adapter Clusters (each with 16 line adapters) can be installed in a Data Communications Cabinet (see figure 1-2).

When up to eight Adapter Clusters are connected to one Data Communications Processor, they must be in the same cabinet as the Data Communications Processor. If more than eight (up to 16) Adapter Clusters are connected to the same DCP, the additional ones must be in an adjoining cabinet.

If an Adapter Cluster is connected to two Data Communications Processors, the DCP's must be in adjoining cabinets with interconnection provided via the required control cable.

CLUSTER CABLE.

If four or fewer Adapter Clusters are connected to a Data Communications Processor, they can be connected individually with separate cables.

If more than four Adapter Clusters are connected to a Data Communications Processor, the Adapter Clusters must be placed in adjacent groups of four. Each group of four is connected to the Data Communications Processor by one special interconnecting cable (five connectors on each cable).

NOTE

The DCP has provisions
for four (4) cables.

CLUSTER EXCHANGE.

The Cluster Exchange connects each Adapter Cluster to the two Data Communications Processors. The circuitry of the Cluster Exchange is included in the Adapter Cluster, but the crosspoint control for the exchange is within the DCP's.

NOTE

The Adapter Cluster has pro-
vision for two (2) cables
(one each to connect to a DCP).

## DATA COMMUNICATIONS PROCESSOR OPTIONAL LOCAL MEMORY.

The Data Communications Processor can utilize an optional local
memory (LM).  The optional local memory is intended for instruction
and control information storage.  Its use permits higher system
performance, i.e., larger throughput and faster response.

SECTION 2

B 6500

DATA COMMUNICATIONS PROCESSOR

GENERAL.

The Data Communications Processor (DCP) is an auxiliary processor
which controls the Adapter Cluster for answering and terminating
calls, observing formal line procedures, polling repetitiously,
and handling all the line discipline message formatting for the
information received and transmitted on many communication lines.

A DCP has access to the B 6500 main memory along with the other
main frame units, such as the B 6500 Processor  and the B 6500
Peripheral Control Multiplexor.  Main memory allocation for the
DCP is controlled by the interaction of the B 6500 Master Control
Program and the DCP programs.  The interaction also allows blocks
of information to be exchanged.

The DCP code may reside in B 6500 main memory or in optional
local memory.  The use of the local memory reduces instruction
fetch time and correspondingly increases the DCP throughput .

The DCP contains a small array of intercommunicating registers,
a simple arithmetic-logical unit, an eight-word scratchpad memory,
and possibly an optional local memory.  It is designed to be micro-
programed.  The instruction repertoire consists largely of two  and
three-address instructions, operating on eight-bit bytes and executed
in a single clock time.  The byte organization fits into a basic
half-word (three-byte) structure which permits efficient half-word
transfers within the DCP along with byte transfers.  Figure 2-1
is a block diagram of the Data Communications Processor.

NOMENCLATURE.

BIT NUMBERING.

The bits in a 52-bit word are numbered 0 through 51 from right to
left (figure 2-2).  Bit 0 is the least significant bit.  Bit 47 is
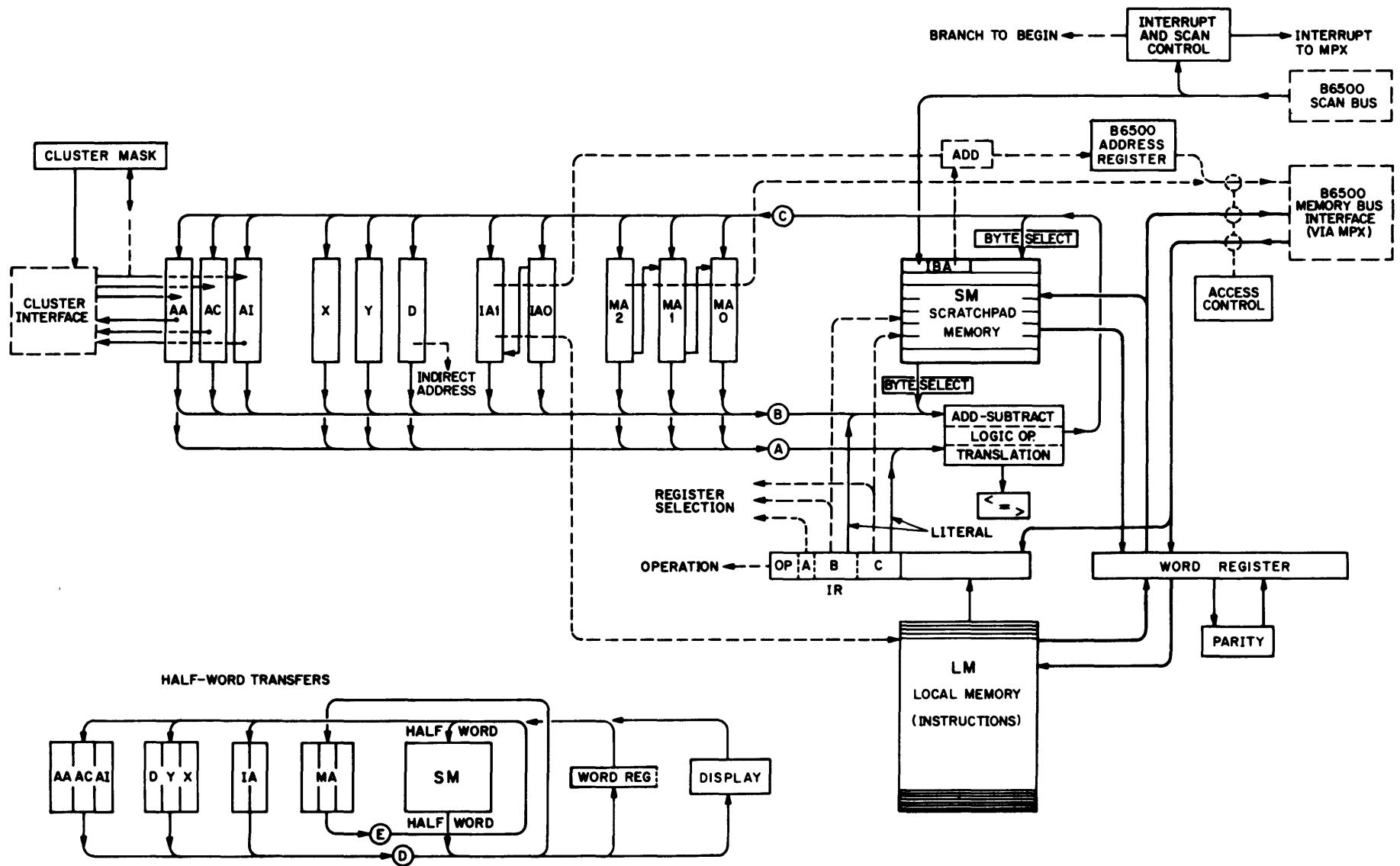
Figure 2-1. DCP Block Diagram

the most significant bit of the information part of the word.  Bits
48, 49  and 50 are tag bits.  Bit 51 is a parity bit (odd parity).

| 51 | 50 | 49 | 48 | 47 |  | 0 |
|----|----|----|----|----|--|---|
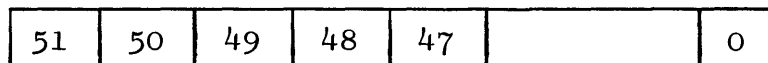
Figure 2-2.  Basic Word Structure

In general, the bits of an n-bit register or byte are numbered 0
through n-1; bit 0 is the least significant bit, and bit n-1 is
the most significant bit.

DESIGNATION OF FIELDS.

A particular field in a register R is identified by using the nomen-
clature R [m:n], where m donotes the starting bit position of a
field extending n bits to the right.  R may also be the OP field of
a register or a register column (figure 2-3).

Example:

AC [4:2] identifies a two-bit field in the AC register
consisting of bits 4 and 3.

AC REGISTER

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

|← 4:2 →|
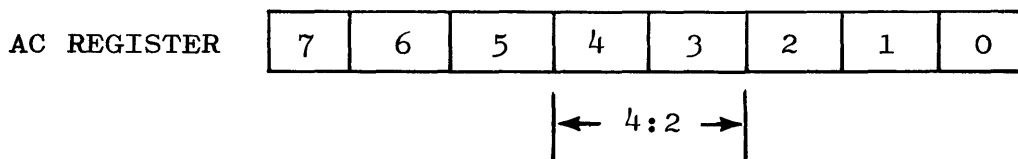
Figure 2-3.  Field Designation

DESIGNATION OF SINGLE BITS.

A particular bit in a register R is identified by using the nomen-
clature Rn, where n donotes the desired bit.  R may also be a field
within a register such as the OP field or a register column such
as the A register column.

Example:

AC4 identifies bit 4 in the AC register.

Bit 47 ⌐                                        ⌐Bit 0

| 6 | 0 | 1 | 2 | 3 | 4 | 5 |
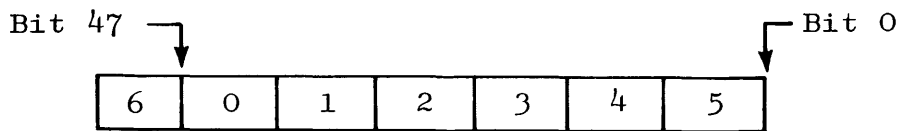
Figure 2-4.   Byte Designation

BYTE DESIGNATION.

The 48-bit information part of the 52-bit word is divided into six,
8-bit bytes.  The byte designations are 0 through 5 from left to
right.  The designation matches the normal sequence of serial-by-
character input or output; high-order character first, or left to
right across a line.  If each byte contained a digit or a number,
the high-order digit would be in byte 0 and the low-order digit
would be in byte 5.  This is illustrated in figure 2-4; note that
the tag field, if present, is designated as byte 6 although it
contains only four bits.  The nomenclature is r-b, where r is a word
or register and b is a byte number.

Example:

   W-2 identifies byte 2 of the W register.

|          L          |          R          |
| ◄─ half-word ─► | ◄─ half-word ─► |

| 0 | 1 | 2 | 3 | 4 | 5 |
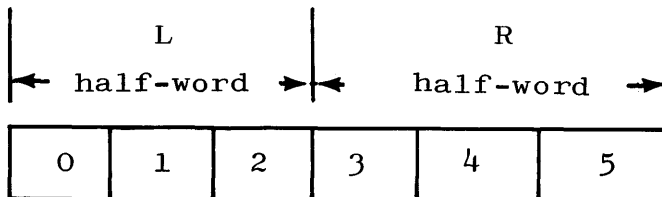
Figure 2-5.   Half-word Designation

HALF-WORD DESIGNATION.

The left half-word, consisting of bytes 0, 1 and 2 is referred
to as half-word L.  The right half-word, consisting of bytes 3,
4 and 5 is referred to as half-word R.  The nomenclature is r-h,
where r is a word or register and h is the half-word designator,
L or R (figure 2-5).

Example:

    W-L identifies bytes 0, 1  and 2 of the W register.

ADAPTER CLUSTER INTERFACE REGISTERS.
All information, data, or control signals transferred between the
Adapter Cluster and the DCP passes through these registers.

ADAPTER ADDRESS REGISTER (AA).
This eight-bit register has designations AA $[7:8]$ and A $[23:8]$. The
AA register contains an adapter designation.  An adapter is actively
designated only during the execution of an Adapter Read, Adapter
Write, or Adapter Interrogate instruction.  When the Adapter Cluster
is used, AA $[7:4]$ contains the cluster number and AA $[3:4]$ contains
the adapter number within the cluster.

ADAPTER CONTROL REGISTER (AC).
This eight-bit register has designations AC $[7:8]$ or A $[15:8]$.
The AC register contains bits which typically describe the infor-
mation on the adapter interface.  For example, a particular code
in the AC register may signify that the AI register contains a
data byte, whereas other codes may identify the AI register contents
as control information of various types.  Only five bits, AC $[4:5]$,
are defined for the Adapter Cluster.

ADAPTER INFORMATION REGISTER (AI).
This eight-bit plus separate parity bit (I21) has designations
AI $[7:8]$, or A $[7:8]$ plus I21.  AI is the primary information
register for the adapter interface.  It can contain data or control
information.

GENERAL PURPOSE REGISTERS.
D REGISTER.
This eight-bit register has designations D $[7:8]$ or G $[23:8]$.  The
D register is an indirect address register that can address other
registers or local memory bytes.  Otherwise, its use is unrestricted
as a general  purpose register.

Y REGISTER.

This eight-bit register has designations Y [7:8] or G [15:8]. The Y register contains an indirect source address when one is called for, but it is not used as an address register. When an indirect source address is used, the contents of the Y register are copied in the Instruction register. Otherwise, its use is unrestricted as a general purpose register.

X REGISTER.

This eight-bit register has designations X [7:8] or G [7:8]. The X register is referenced in the Branch Relative instruction. Otherwise, its use is unrestricted as a general purpose register.

COMPARISON REGISTER (CF).

This eight-bit register has designations I [23:8]; compare bits CF1 and CF0, or I [17:2]. The CF register contains eight special control flip-flops. The register is addressable, but in general, the flip-flops should not be programmatically controlled because it inhibits their extraordinary inputs.

Table 2-1

CF Flip-Flop Designations

| Bit Position | Flip-Flop Designation | Special Function |
|---|---|---|
| 0 | I16 (CF0) | Carry flip-flop. Controlled by the carry/borrow signal from the arithmetic unit (when it is used). Cleared when a branch occurs or by a Shift MA. |
| 1 | I17 (CF1) | Zero-result flip-flop. Controlled by the result = zero signal from |

Table 2-1 (cont)

CF Flip-Flop Designations

| Bit Position | Flip-Flop Designation | Special Function |
|---|---|---|
| 1 | I17 (CF1) (cont) | the arithmetic unit (when it is used). Cleared when a branch occurs or by a Shift MA. |
| 2 | I18 | Internal Carry flip-flop. Used in place of I16 (CFO) during the IA + IBA addition preceding a MM fetch and also during a Half-word Add or Subtract. Otherwise zero. |
| 3 | I19 | Fault Control flip-flop. Set when the special fault instruction is executed. If I19 = 1, the fault interrupt action is modified. |
| 4 | I20 | IA Counted flip-flop. Set when the address is counted up normally; otherwise cleared when the instruction is completed. |
| 5 | I21 | AI parity bit. Controlled by the AI8 signal from the cluster interface and by the Parity instruction. |
| 6 | I22 | MM interlock flip-flop. Must be zero. (If a MM access is separately buffered, I22 = 1 defeats the "wait" which occurs when the MA or W register is addressed before the access is completed). |

Table 2-1 (cont)

CF Flip-Flop Designations

| Bit Position | Flip-Flop Designation | Special Function |
|---|---|---|
| 7 | I23 | Main System Attention Needed flip-flop. Set by the scan-out of Set Attention Needed. Cleared when it causes a branch in the instructions which explicitly test it. |

The CF1 and CF0 flip-flops are set by the result of arithmetic and logical instructions to denote the conditions listed in table 2-2 which control the conditional branches.

Table 2-2

CF1 and CF0 Conditional Results

| CF1 | CF0 | Compare Condition | Other Significance |
|---|---|---|---|
| 0 | 0 | > | Cleared state |
| 0 | 1 | < | Carry or borrow |
| 1 | 0 | = | Zero result |
| 1 | 1 | | Carry and zero result (unlikely) |

MEMORY ADDRESS REGISTERS.

MEMORY ADDRESS REGISTERS (MA-2, MA-1, and MA-0).
The three, 8-bit registers have designations MA-2 [7:8], MA-1 [7:8], MA-0 [7:8]; or M [23:24]. These registers are used for addressing B 6500 main memory and the DCP local memory. The 20-bit B 6500 main memory address excludes MA-2 [7:4]; the maximum 14-bit DCP local memory excludes MA-2 and MA-1 [7:2].

The MA registers are always used in half-word transfers and may be
used in full-word transfer operations. The MA registers receive a
half-word selected from a variety of sources (on the D bus), and
simultaneously becomes the source of a half-word that is sent to
one of several destinations (on the E bus). The MA-2, MA-1 and
MA-0 registers are used implicitly in the Half-word Add and Subtract
instructions.

These registers may be concatenated in various ways by the Shift
Right MA instructions in which their contents are shifted right.
The MA registers may also be used individually as general purpose
registers.

INSTRUCTION ADDRESS REGISTERS.
The two, 8-bit registers have designations IA-1 [7:8] and
IA-0 [7:8]; or I [15:16]. The concatenated IA registers hold
the instruction address to address the DCP local memory directly
or provide the relative part of an address for the B 6500 main
memory. The most significant bit, IA-1 [7:1] or I15, selects
the proper memory for this address:

      a.    I15 = 0    IA has a relative address for the B 6500
                       main memory.

      b.    I15 = 1    IA has a direct address for the DCP local memory.

The least significant bit, IA-0 [0:1] or I00, selects one of the
two half-word instructions in a full instruction word.

The IA registers are counted up automatically as each instruction
is loaded. They are loaded by the branch instructions, and may
also be addressed like the other registers. If an IA is addressed
as a designation, a new instruction fetch occurs after the current
instruction is completed.

B 6500 ADDRESS REGISTER (BA).
This 20-bit register has a designation BA [19:20]. The B 6500
address register contains the actual instruction address for the

instruction words in B 6500 main memory.  The actual address is
the sum of the relative address in the IA register and the instruc-
tion base address.  The B 6500 address register is not addressable.
When a B 6500 memory access is required, the register is loaded
either from the MA register, or indirectly from the IA registers.

FULL WORD REGISTERS.

INSTRUCTION REGISTER (IR).

This 52-bit register has designations IR [51:52] or P [51:52];
P-L denotes P [47:24] and P-R denotes P [23:24].  The Instruction
register holds a full instruction word containing two, 24-bit
instruction words.  It is loaded from either the DCP local memory
or the B 6500 main memory.  The tag field, P [51:4], must have a
correct odd parity bit in P [51:1]  and P [50:3] must contain a six.

P-L is the active half of the IR in that instructions are executed
only from P-L.  The second instruction is transferred from P-R to
P-L for execution.

WORD REGISTER (W).

This 52-bit register has a W [51:52] designation; W-L denotes
W [47:24]  and W-R donates W [23:24].  It is a data buffer used for
transferring full words to or from the scratchpad memory, the DCP
local memory, and the B 6500 main memory.  The parity bit in
W [51:1] is automatically generated and checked.

The W register is divided into six individually-addressable, eight-
bit registers plus a four-bit tag register.  W-L and W-R may also
be addressed as separate half-word register columns.

NOTE
The contents of the other registers
(except P) can be transferred to
the Word register  and they can be
loaded from it.

<u>CLUSTER MASK REGISTER.</u>

This register contains 16 independent flip-flops, one for each of the 16 possible Adapter Clusters. Each flip-flop in this register gates the corresponding cluster attention needed signal when it is on. If a Cluster Mask flip-flop is off, the DCP does not detect a cluster attention needed signal from that Adapter Cluster. In systems where an Adapter Cluster is connected to two DCP's, the corresponding Cluster Mask flip-flops in each DCP should be loaded so that only one DCP responds to a cluster attention needed signal.

This register is loaded with the Set Cluster Mask instruction.

<u>MEMORIES.</u>
SCRATCHPAD MEMORY (SM).
The scratchpad memory is an integrated circuit memory which contains eight, 52-bit words. Information can be read out or stored in full words, 24-bit half-words, or individual eight-bit bytes. Read and write are independent and can occur simultaneously in different locations. Read-out is non-destructive.

Read access time is 40 nanoseconds. Write access time is 50 nanoseconds including clock pulse. Access times do not include the address decoding time.

<center>NOTE</center>
<center>The information in SM is</center>
<center>lost if power goes down.</center>

SM is used for fast-access, temporary, data storage. The SM locations are like flip-flop registers except that the same location cannot be used both as a source and a destination when the result is a complementary function of the source operand. If the same byte is improperly addressed both as a source and destination, an invalid operator fault interrupt occurs.

RESERVED LOCATIONS. Table 2-3 defines the half-words that are used in hardware functions. The number within the parenthesis and following SM is the word address.

Table 2-3

Hardware Half-word Functions

| Half-word | Use |
|---|---|
| SM (0)-L | Instruction base address |
| SM (0)-R | Fault interrupt, Branch on Breakpoint, Initialize |
| SM (1)-R | Fault interrupt |

LOCAL MEMORY (LM).

Local memory is a 52-bit word memory used for instruction or data storage. Its use optimizes instruction fetch time and data fetch time. It is available in 4096 word modules which require the space of an Adapter Cluster in the DCP cabinet. One DCP may have as many as four of these memory modules. The cycle time for this memory is 650 nanoseconds.

BASIC HALF-WORD AND BYTE ORGANIZATION.

The data registers and scratchpad memory half-words are organized in a matrix that is illustrated in figure 2-6.

|  | A | G | I | M | SM-L | SM-R | W-L | W-R |
|---|---|---|---|---|---|---|---|---|
| 2 | AA | D | CF | MA-2 | 0 | 3 | 0 | 3 |
| ROWS  1 | AC | Y | IA-1 | MA-1 | 1 | 4 | 1 | 4 |
| 0 | AI | X | IA-0 | MA-0 | 2 | 5 | 2 | 5 |

Figure 2-6. Matrix Representation

The registers are interconnected along the rows by busses. The principal busses are two half-word busses named the D bus and the E bus. Each bus consists of three, 8-bit sections; one section for each row.

D BUS.

The D bus is a multiple source bus with three destinations. Any of the register columns or SM half-words can be chosen as its source. It is not necessary to select all of the registers in the column;

if there is no source for a section of the bus, the section carries zeros.

The destination of the D bus may be the MA register column, the W-L column, or the B bus.

E BUS.

The E bus has three sources and many destinations. The source depends upon the instruction being executed. The information on the E bus can come from the MA register column, the W-L column or the C bus.

Any of the register columns or SM half-words can be chosen as the destination of the E bus. However, not all registers in a column need be selected; if no destination is specified for a section of the E bus, no information is stored.

FULL WORD TRANSFER.

Full word transfers between the W register and a selected SM word location are made possible by separate busses which are used in addition to the D and E busses. The separate busses interconnect the R half-words including the tag field.

ARITHMETIC, LOGICAL, AND TRANSLATION SECTION.

FUNCTIONS.

The DCP performs the following arithmetic, logical, and translation functions on 8-bit bytes:

    a.   Add.

    b.   Subtract.

    c.   Logical AND.

    d.   Logical OR.

    e.   Exclusive OR.

    f.   Parity

    g.   Translations (EBCDIC to USASCII or BCL, and USASCII or BCL to EBCDIC).

◇ C = BUS IDENTIFICATION
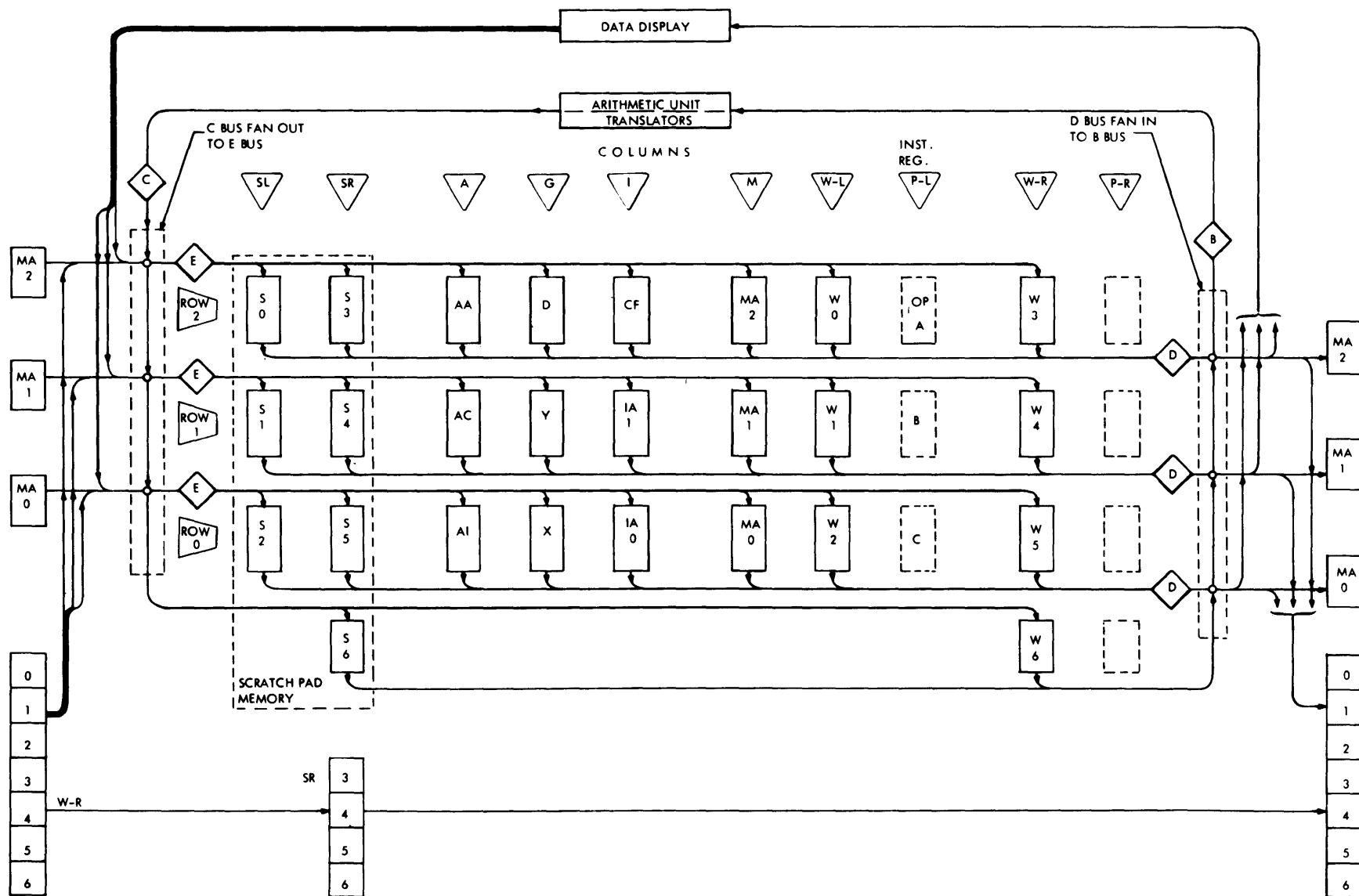


Figure 2-7.   B 6500 DCP Register Organization.

The section containing the functional circuitry appears as a block with two input busses and one output bus. Each bus has an eight-bit (one byte) capacity. The input busses are named the A bus and the B bus. The output bus is named the C bus (figure 2-7).

A BUS.

The A bus is a multiple source bus. One of seven registers can be selected as the source: AA, D, Y, MA-2, MA-1, X, and MA-0. If a register is not selected, the A bus carries zeros.

B BUS.

The B bus is the logical OR of the three sections of the D bus. For normal byte operations, only one register should be selected as the source of the D bus.

C BUS.

For byte operations, the C bus becomes the source for each section of the E bus. The other sources of the E bus are gated off. For normal byte operations, only one register should be selected as the destination of the E bus.

INSTRUCTION FORMAT.

The 52-bit, full-word instruction is loaded into the Instruction register on a fetch cycle. The instruction word contains two, 24-bit instructions, must have odd parity, and contain a tag field of IR [50:3] = 6. If these conditions are not met, the instruction word is detected as invalid and the instructions in it are not executed.

INSTRUCTION HALF-WORD.

The 24-bit instruction half-word is divided into four fields as follows:
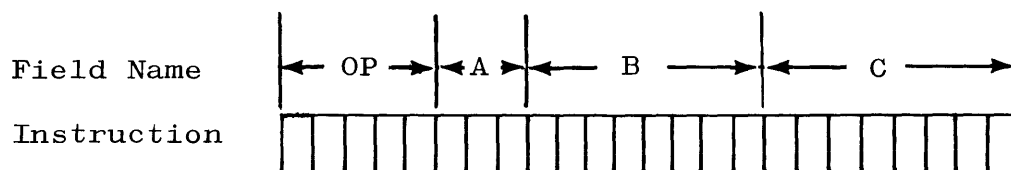


Figure 2-8. Instruction Half-Word Format

The OP field contains the basic instruction code.

The A field may be an extension of the OP field, or it may contain a register address.

The B field typically contains the address of the source. It may also contain a literal.

The C field typically contains the address of a destination. It may also contain a literal.

OP FIELD.

The OP field contains the basic instruction code. The actual instructions are chosen in a pattern that allows the instructions to be divided readily into five groups. The manner in which the other fields of an instruction are interpreted depends on the group the instruction is in (table 2-4).

Table 2-4

Instruction Groups

| Group | Op Codes | Description |
|-------|----------|-------------|
| 0 | 0 0 0 - - | Various instructions that do not contain addresses (exception: branch address in Breakpoint). |
| 1 | 0 0 1 - - | Instructions in which the B:C fields contain only a branch address. |
| 2 | 0 1 0 - - | Instructions which move half-words and load or unload the Word register. |
| 3 | 0 1 1 - - | One-operand byte conversion instructions and special half-word instructions. |
| 4 | 1 - - - - | Two-operand arithmetic and logical byte instructions. |

A FIELD.

For instructions not in Group 4, the A field is an extension of the OP field. Typically the A field specifies variants of the basic instruction. For the instruction in Group 4, the A field contains a register address as shown in table 2-5.

Table 2-5
Group 4 A Field Addressing

| A field | | | Instruction is not "literal in C field" | Instruction is "literal in C field" |
|---|---|---|---|---|
| A2 | A1 | A0 | Register selected as source | Register selected as destination |
| 0 | 0 | 0 | None (data = zeros) | None (no data stored) |
| 0 | 0 | 1 | AA | Indirect (The B field selects both the source and the destination) |
| 0 | 1 | 0 | X | X |
| 0 | 1 | 1 | Y | Y |
| 1 | 0 | 0 | D | D |
| 1 | 0 | 1 | MA0 | MA0 |
| 1 | 1 | 0 | MA1 | MA1 |
| 1 | 1 | 1 | MA2 | MA2 |

B FIELD.

The B field can contain any of the following items:

a. Literal.

b. B:C branch address.

c. B:C local memory address.

d. Register address.

e. Scratchpad memory address.

f. Word register byte address.

g. Indirect address designation.

LITERAL. The B field contains a literal in most group 0 instructions and in the group 3 and 4 instructions that specify a literal in the B field. The function or use of the literal depends on the instruction.

B:C BRANCH ADDRESS (GROUPS 0 AND 1). The B field is concatenated with the C field and together they contain a branch instruction address.

| B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
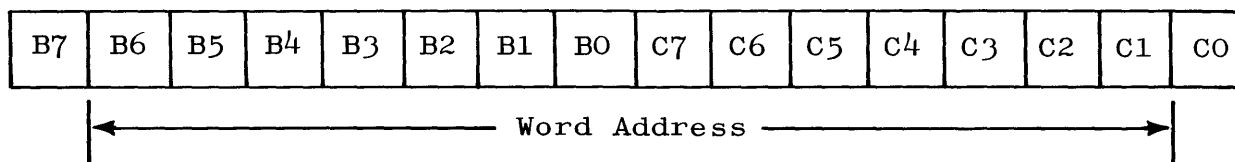
$\longleftarrow$ ———————— Word Address ——————— $\longrightarrow$

Figure 2-9. B:C Branch Address Format

B [7:1] contains the instruction source bit as shown below:

    B7 = 0 Branch to MM

       = 1 Branch to LM

C [0:1] controls the half-word selection as follows:

    C0 = 0 Branch to half-word-L

       = 1 Branch to half-word-R

The B field is transferred to the IA-1 register and the C field is transferred to the IA-0 register. When the branch is taken, a new instruction fetch cycle is initiated (see fetch cycle).

Applicable Instructions:

| BKP | DBYZ | GOTO |
| ARWN | DBYN | GOX |
| ARIN | BRAN | GOI |

B:C LOCAL MEMORY ADDRESS. In the "local memory direct address" instructions in group 2, the B field is concatenated with the C field and together they contain a local memory data address.
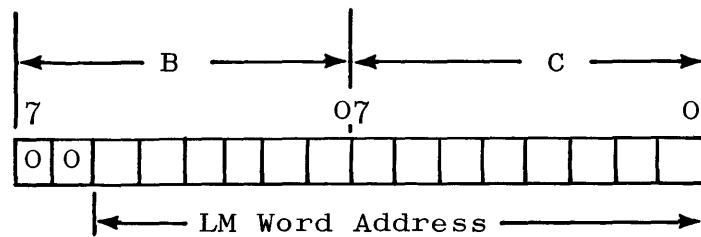


Figure 2-10.   LM Direct Address Format

B [7:2] should contain 00.  When used, the direct LM address in the B:C field is transferred to the MA1:MA0 registers.

Applicable instructions:
    LMRD          LMWD

B ADDRESS.  The B field by itself contains an address in group 2 instructions (except the "local memory direct address" instructions) and in group 3 and 4 instructions that do not specify a literal in the B field.  The B address is usually the address of a source.

The source data bits are all zeros if the B address does not select a location.

Register Address.

The methods used for obtaining a byte operand or a half-word operand when given an eight-bit address are defined below:  The following descriptions all refer to the B field.

A register address is identified by B [7:2] = 0.  B [5:3] contains a register column address and B [2:3] contains row selection bits (figure 2-11).

B field:



Row Selection
Column Address
Register Address ID
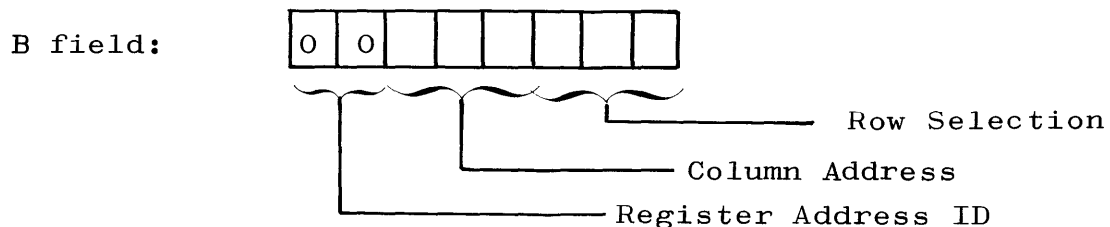
Figure 2-11.  Register Address Formats

2-19

| B [5:3] | B2 = 1 | B1 = 1 | B0 = 1 |
|---------|--------|--------|--------|
| 000 | none* | none* | none* |
| 001 | AA | AC | AI |
| 010 | D | Y | X |
| 011 | MA-2 | MA-1 | MA-0 |
| 100 | CF | IA-1 | IA-0 |
| 101 | W-0 | W-1 | W-2 |
| 110 | W-3 | W-4 | W-5 |
| 111 | reserved | reserved | reserved |

* If source then zeros

Figure 2-11 (cont).    Register Address Formats

The row select bits are specified independently.  One, two, or
three registers in the same column can be selected simultaneously.
All registers in a column are selected if B [2:3] = 111.  If a row
select bit is 0, the data bits from that row are all zeros.  No
register is selected if B [2:3] = 000.

NOTE

In the Half-word Add and Half-word Subtract
instructions, B [2:3] should contain 000; the
rows are internally selected in order.

For the byte instructions in groups 3 and 4, only one register
would normally be selected.  If more than one register is selected
in a register column, the data in the corresponding positions in
the different rows is logically ORed.

Scratchpad Memory Address (Groups 2, 3 and 4).
B [7:2] = 10 identifies a scratchpad memory address.  B [5:3]
contains a scratchpad word address and B [2:3] contains a half-
word selection bit or a byte designation.

B field:
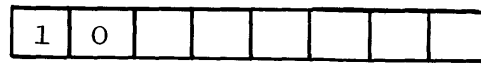
```
        ┌─┬─┬─┬─┬─┬─┬─┬─┐
        │1│0│ │ │ │ │ │ │
        └─┴─┴─┴─┴─┴─┴─┴─┘
         ╰─╮╯ ╰─╮╯ ╰─╮╯
            │      │      └──── Half-word Selection
            │      │
            │      └──────── SP Word Address
            │
            └──────────── SP Memory ID
```

| Code | | | Scratchpad |
|---|---|---|---|
| B5 | B4 | B3 | Memory Word |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 2 |
| 0 | 1 | 1 | 3 |
| 1 | 0 | 0 | 4 |
| 1 | 0 | 1 | 5 |
| 1 | 1 | 0 | 6 |
| 1 | 1 | 1 | 7 |

Figure 2-12. Scratchpad Word Selection

| Code | | | Group 2 Half-word Selected | Group 3 Half-word Selected | Group 3 or Group 4 Byte Selected |
|---|---|---|---|---|---|
| B2 | B1 | B0 | | | |
| 0 | 0 | 0 | L | L | None |
| 0 | 0 | 1 | L | X | 0 |
| 0 | 1 | 0 | L | X | 1 |
| 0 | 1 | 1 | L | X | 2 |
| 1 | 0 | 0 | R | R | 3 |
| 1 | 0 | 1 | R | X | 4 |
| 1 | 1 | 0 | R | X | 5 |
| 1 | 1 | 1 | R | X | 6 (Tag field) |

X = Improper Code

Figure 2-13. Half-word and Byte Selection

In group 2 instructions, B [2:1] contains the half-word selection bit and the B [1:2] bits are not significant. B [2:1] should be zero in the instructions that address a scratchpad full word.

In the Half-word Add and Half-word Subtract instructions in group 3, B [1:2] should contain 00. In the other group 3 instructions, B [2:3] contains a byte designation as in group 4 instructions.

NOTE

If B designates both a source and a destination, the result of the instruction must not depend on the source.

Word Register Byte Address.
In group 4 instructions and in the Translate and Parity instructions in group 3, the B address can address the bytes in the Word register as if it were a scratchpad memory word. This mode of addressing augments the addressing of the Word register as two columns, and it is the only way to address the Word register tag field. A Word register byte address is identified by B [7:2] = 11, and B [5:3] must contain 111. B [2:3] contains the single byte designation.
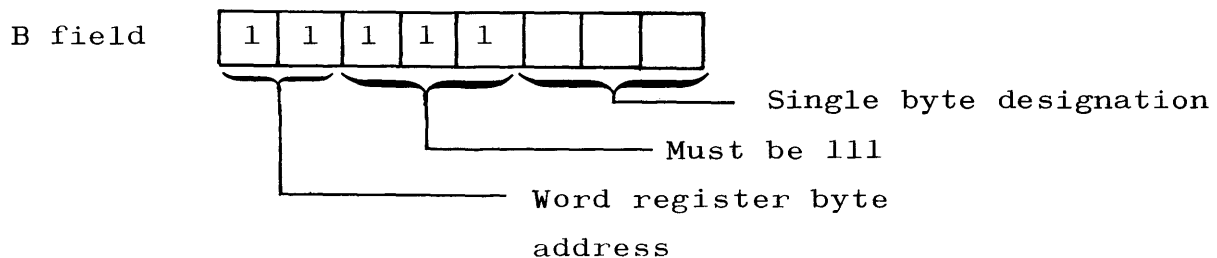


| Code | | | Word Register |
|------|------|------|---------------|
| B2 | B1 | BO | Byte Selected |
| O | O | O | None |
| O | O | 1 | 1 |
| O | 1 | O | 2 |
| O | 1 | 1 | 3 |
| 1 | O | O | 4 |
| 1 | O | 1 | 5 |
| 1 | 1 | O | 6 |
| 1 | 1 | 1 | 7 |

Figure 2-14. Word Register Byte Addressing

NOTE
Although not disabled, Word register byte
addressing should not be used for instructions
in group 2 or for other instructions in group 3.

Applicable instructions:

| | | | |
|-------|-------|-------|-------|
| MOVE  | MMWP  | ADD   | LANC  |
| LMRI  | MWRP  | ADDB  | LOAM  |
| SMRD  | HAD   | ADDC  | LOR   |
| LMWI  | HADB  | SUB   | LORB  |
| SMWD  | HSB   | SUBB  | LORC  |
| MMR   | HSBB  | SUBC  | LEO   |
| MMVW  | TRAN  | LAN   | LEOB  |
| MMWR  | PARY  | LANB  | LEOC  |

## Indirect Address (Groups 2, 3, and 4).

B [7:2] = 01 specifies an indirect address.
The contents of the Y register are used as the B address.  B [5:6]
must be 000000.

B field    | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

Must be all zeros
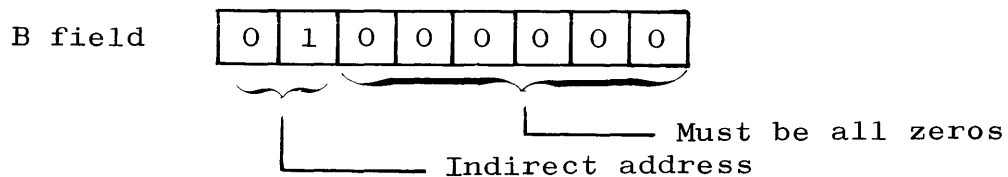
Indirect address

Figure 2-15.  Indirect Address Format

The address format in the Y register is the same as the address
format in the B field, except that Y [7:2] must not be 01.

An indirect address designation in the B address increases the
execution time of an instruction by one unit.

## Special Indirect Address.

In the Shift MA and Set B 6500 Interrupt instructions in group 0,
the B field normally contains a literal with B [7:2] = 00.  If
B [7:2] = 01 in these instructions, the contents of the Y register
are used as the B field literal.  The indirect literal designation
increases the instruction execution time by one unit.

B Address Used As Destination Address.

In group 4 instructions that specify a literal in the C field and in which the A field contains 001, the B address is used as a destination address as well as a source address. The same location is addressed both as a source and as a destination.

C FIELD.

The C field can contain the same items as the B field.

LITERAL. The C field contains a literal in most group 0 instructions and in the group 4 instructions that specify a literal in the C field. The function or use of the literal code depends on the instruction.

C ADDRESS. The C field by itself contains an address in group 2 instructions (except the local memory direct address instructions), the group 3 instructions, and the group 4 instructions that do not specify a literal in the C field. The C address is always the address of a destination.

If the C address does not select a location, information is not stored.

Register Address.

A register address is identified by C [7:2] = 00. Only the registers designated by a row select bit = 1 receive information in the addressed register column. See Register Address on page 2-19.

Scratchpad Memory Address.

A scratchpad memory address is identified by C [7:2] = 10. The use of the other bits is the same as Scratchpad Memory Address on page 2-20.

Word Register Byte Address.

A Word register byte address is identified by C [7:2] = 11, and C [5:3] must contain 111. The use of the other bits is the same as the Word Register Byte Address on page 2-22.

Indirect Address.

An indirect address is identified by C [7:2] = 01. If C [7:2] = 01, the contents of the D register are used as the C address. C [5:6] must be 000000.

The address format in the D register is the same as the address format in the C field, except that D [7:2] must not be 01. An indirect address designation in the C address does not increase the instruction execution time.

FETCH CYCLE.

INSTRUCTION FETCH.

The fetch cycle loads a full instruction word into the Instruction register, either from local memory or from B 6500 memory. The instruction words are addressed by the IA registers.

A full instruction word contains two separate half-word instructions In a normal sequence, the half-word L instruction is executed first, followed by the half-word R instruction. In this case, the fetch time for the second instruction is zero.

INSTRUCTION ADDRESS.

If IA-1 [7:1] = 1, IA-1 [6:7] and IA-0 [7:7] concatenated contain the address of a full instruction word in local memory, IA-0 [0:1] selects the half-word instruction in the full instruction word; 0 specifies half-word L, 1 specifies half-word R.
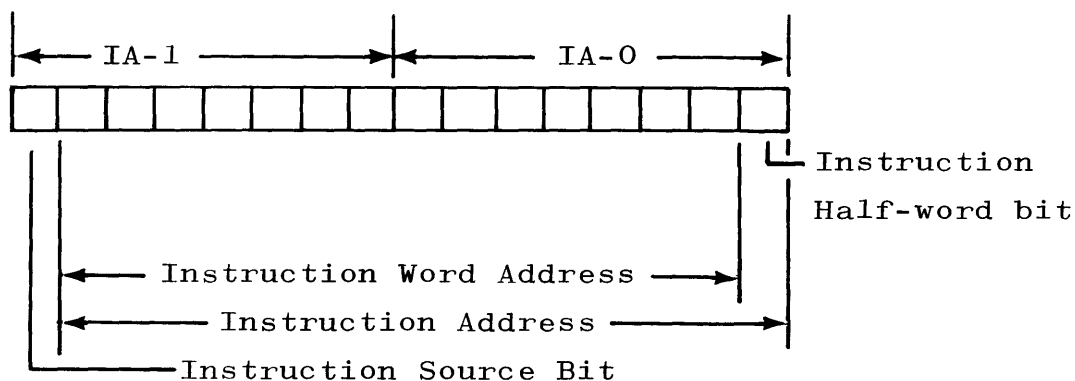
Figure 2-16. Instruction Word Address Format

2-25

If IA-1 [7:1] = 0, IA-1 [6:7] and IA-0 [7:7] concatenated contain a relative address for an instruction word in B 6500 memory.  The base address, called the "instruction base address" or IBA, is presumed to be always present in half-word L of scratchpad memory word 0. IA-0 [0:1] selects the half-word instruction in the full instruction word.

The instruction address in IA-1 [6:7] and IA-0 [7:8] (including IA-0 [0:1] is counted up one by the same clock pulse that starts the execution of an instruction.  There is no count-up into IA-1 [7:1].  See fault interrupts for address overflow.

FETCH FROM LOCAL MEMORY, NORMAL SEQUENCE.

Fetch time:  1 unit

Using the address in the IA registers, the instruction word in local memory is read out and loaded into the Instruction register. The first instruction is executed immediately along with a simultaneous parity check.  See fault interrupts for parity error action.

FETCH FROM LOCAL MEMORY AFTER BRANCH.

Fetch time for instruction-L:  2 units
Fetch time for instruction-R:  3 units

This operation is the same as for normal sequence, except that execution of instruction-R is delayed by 1 unit.

FETCH FROM B 6500 MEMORY.

Fetch time for instruction-L:  3 units +
B 6500 memory read access time.

Fetch time for instruction-R:  4 units +
B 6500 memory read access time.

Probable B 6500 memory read access time via the Multiplexor, if no DCP data access is in process:

Minimum:  5 units (Multiplexor clock)
Typical:  8 units

When B 6500 access control is free, the relative address in the IA registers is added to IBA in Scratchpad Memory word 0, and the sum is transferred to the B 6500 address register.  A B 6500 memory read request is set with program execution inhibited in the DCP until the instruction word is obtained.  When the instruction word is received, it is transferred directly to the Instruction register and is executed.  The instruction word is parity checked immediately after the fetch.  Execution of instruction-R is delayed by 1 unit. See fault interrupts for parity error or for fetch error.

The address addition is performed in three steps:

### IBA (Scratchpad Memory)

$IA-0$ $[7:8]$ + (Word 0 byte 2)                          Address Section 0
$IA-1$ $[6:7]$ + (Word 0 byte 1) + internal carry     Address Section 1
        zero + (Word 0 byte 0) + internal carry     Address Section 2

CF0 and CF1 are not affected by the above address addition.

ACTIONS RELATED TO THE FETCH CYCLE.
PARITY CHECK.  Immediately after a fetch cycle, the new instruction word is checked for odd parity.  If there is a parity error, the execution of the L instruction is inhibited and a fault interrupt occurs.

TAG CHECK.  Immediately after a fetch cycle, the tag field, IR $[50:3]$, is checked for the code 110.  If the tag field does not contain 110, the execution of the L instruction is inhibited and a fault interrupt occurs.

L INSTRUCTION PASSED OVER.  If IA0 $[0:1]$ = 1 when a new instruction word is loaded in the Instruction register, the L instruction is skipped over as if it were a no operation order.  Only the R instruction in the word is executed.

TRANSFER OF R INSTRUCTION. The R instruction is executed after the
L instruction unless a branch occurs in the L instruction. The R
instruction is always transferred to the L half of the Instruction
register before it is executed. The transfer is simultaneous with
the completion of the L instruction, so fetch time is zero. The
transfer is not a fetch cycle.

If the L instruction in a word is not executed, there is a fetch
time of one unit for the R instruction.

INSTRUCTION ADDRESS COUNT UP. The instruction word address and the
instruction half-word selection bit together comprise an instruction
address for the half-word instructions. The instruction address is
counted up by one each time that an instruction to be executed is
transferred into the L half of the Instruction register. Thus the
instruction address always designates the instruction to be exe-
cuted next.

The Execute flip-flop and the I20 flip-flop are set when the in-
struction address is counted up. These flip-flops are cleared when
an instruction is completed unless the next instruction is trans-
ferred into P-L at the same time. The I20 bit is significant
after a fault interrupt.

The instruction source selection bit is subject to count-up along
with the instruction address. If the instruction address is counted
from all ones to all zeros, the instruction source selection bit is
completed. There is no detection of this action in the hardware.

If a branch is taken, or if an instruction addresses the IA-1 or
IA-0 register as a destination, the automatic count-up of the in-
struction address is inhibited and a fetch cycle is ordered.

FAULT INTERRUPTS.
A fault condition, such as a memory parity error, holds up the exe-
cution of a current instruction and inhibits the normal fetch of the
next instruction. The fault condition creates a special Branch on
Breakpoint instruction and inserts it in the Instruction register.

The unconditional branch condition is set and the special instruction is executed normally.  A fault condition creates the following code in the Instruction register:

| OP | A | B | C |
|---|---|---|---|
| 0 0 0 0 1 | 0 0 0 | M 0 0 0 0 0 0 0 | 0 0 0 N N N N N |

Figure 2-17.  Fault Code in the Instruction Register

M is normally set equal to the I15 bit (LM-MM bit).  Thus the branch address is for the memory which contains the program that was interrupted.

N N N N N:  Characterizes the fault condition.  (See fault codes.)

The state of the Execute flip-flop at the time of the fault condition is shown by the I20 bit.  The I20 bit indicates whether or not the instruction address has been counted up.

I20 = 0:  Fault condition occurred between instructions. The IA registers contain the address of the next instruction to be fetched.

I20 = 1:  Fault condition interrupted the execution of an instruction.  The IA registers contain the address of the next instruction to be fetched, but the current instruction was not completed.

The definitive fault branch addresses are created only if I19 = 0. If a fault condition occurs when I19 = 1,  then any normal fault branch address is replaced by the "out of control" branch address for main memory.  If a fault condition occurs when I19 = 1 and I15 = 0, which indicates that a fault-handling program is already running from main memory,  then the Run flip-flop is cleared and DCP operation stops.

Exception:  The scan-out Initialize is exempt from control by the I19 bit.

The I19-bit is set when a fault occurs.  The I19-bit must be cleared programmatically.

Table 2-6

Fault Codes

| | | Branch Address | | | | | | Description |
|---|---|---|---|---|---|---|---|---|
| I20 | I19 | 39 | 28 | 27 | 26 | 25 | 24 | |
| E | O | O | O | O | O | O | O | Initialize from scan-out |
| 1 | O | M | O | O | O | O | 1 | LM data address invalid |
| E | O | M | O | O | O | 1 | O | LM data word (W) parity error |
| E | O | M | O | O | O | 1 | 1 | Adapter cluster error |
| E | O | M | O | O | 1 | O | O | MM protected write denied |
| E | O | M | O | O | 1 | O | 1 | MM data address invalid |
| E | O | M | O | O | 1 | 1 | O | MM data word (W) parity error |
| E | O | M | O | O | 1 | 1 | 1 | MM memory-detected error on data access |
| E | O | 1 | O | 1 | O | O | O | LM inst. tag error, invalid op. |
| O | O | 1 | O | 1 | O | O | 1 | LM inst. address invalid |
| E | O | 1 | O | 1 | O | 1 | O | LM inst. (P) parity error |
| E | O | M | O | 1 | O | 1 | 1 | Unexpected time-out |
| E | O | O | O | 1 | 1 | O | O | MM inst. tag error, invalid op. |
| O | O | O | O | 1 | 1 | O | 1 | MM instruction address invalid |
| E | O | O | O | 1 | 1 | 1 | O | MM inst. (P) parity error |
| E | O | O | O | 1 | 1 | 1 | 1 | MM memory detected error on fetch |
| E | 1 | O | 1 | O | O | O | O | Fault following a fault:  out of control |

E = state of Execute flip-flop at time of
    fault condition.

M = I15 bit (LM-MM bit).

PRIORITY AMONG FAULTS.

If two fault conditions occur simultaneously, one of them takes precedence with the following priority:

2-30

a. A scan-out Initialize or Halt displaces any fault condition.

b. An execution fault takes precedence over a fetch fault.

c. An invalid memory address fault precludes other memory access faults.

d. A memory parity error fault takes precedence over code-produced faults.

e. A MM memory-detected error fault takes precedence over both a parity error fault and a code-produced fault.

Execution faults from different instructions cannot occur at the same time.

FAULT INSTRUCTION EXECUTION.
When the special Branch on Breakpoint (Fault) instruction is executed, the following actions take place simultaneously:

a. The contents of the MA registers are stored in SM word O-R half.

1) Exception: For the out of control fault, the contents of the MA registers are stored in SM word 1-R half.

b. The contents of the I register column are transferred to the MA registers.

c. The branch address in the B:C fields is transferred to the IA registers.

d. The I19 is set to 1.

Following the branch a fetch cycle is ordered as usual.

ATTENTION NEEDED TIMER.
The attention needed timer is used to warn the DCP of unserviced CAN's and SAN's. This timer can time-out thus causing the unexpected time-out for two reasons:

a. After a CAN has occurred and there have been no adapter reads for 500 microseconds the attention needed timer will time-out.

b. A SAN condition exists continually for 0.5 milliseconds.

SCAN BUS.

The B 6500 processor has ultimate control over the Data Communications Processor (DCP) by means of the scan bus. Three different scan-out orders can be sent to the DCP: Initialize, Set Attention Needed, and Halt. The DCP does not accept scan-in orders.

The address/function code format for the DCP's is shown in figure 2-18.



Bit [0:1] =
Address Significance:
0 = DCP address not significant; all DCP's to respond

1 = Only the addressed DCP to respond

Bits [3:3] = DCP address

Bits [7:3] = Function code:
0 0 0 = Initialize
0 1 0 = Halt
1 0 0 = Set Attention Needed
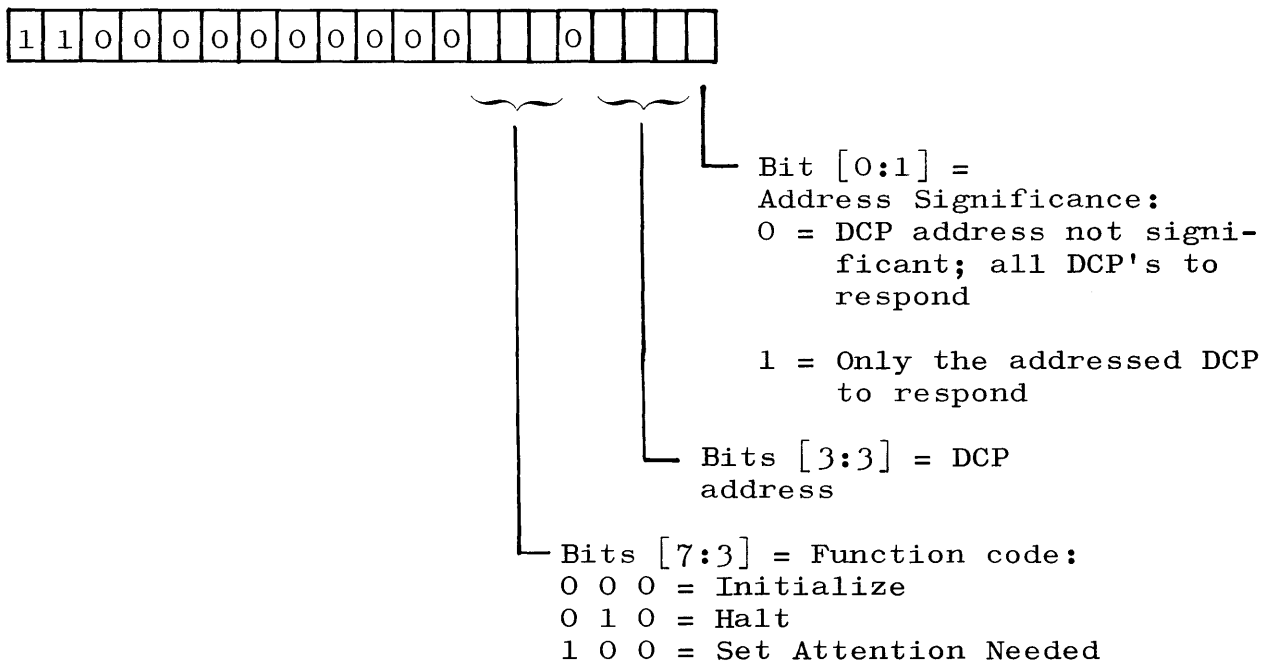
Figure 2-18.   DCP Address/Function Code format

SCAN-OUT INFORMATION WORD.

The scan-out information word is used only in the Initialize order. This 48-bit word has the following format:

Bits [47:28] = all zeros

Bits [19:20] = Instruction base address

READY AND SCAN ACCESS OBTAINED.

Normally, the DCP returns a ready signal on the scan bus when the DCP is addressed by any scan order. The ready signal allows the main system processor to maintain the scan order in anticipation of a scan access obtained signal. The scan access obtained signal is sent by the DCP when it performs the scan-out operation as directed or when it detects an invalid scan order.

NOT READY.

The following conditions prevent the DCP from sending the ready signal when it is addressed by a scan order:

   a.  OFF-LINE/ON-LINE switch in the OFF-LINE position.

   b.  HOLD P-L/STOP ON FAULT switch in the HOLD P-L POSITION.

   c.  STOP IMMEDIATE/STOP ON FETCH switch in the STOP IMME-
       DIATE position.

   d.  Run flip-flop off in the Set Attention Needed scan-out.

The absence of the ready signal on the scan bus is detected by a time-out in the main system processor, which then ends the scan order. The DCP is identified as being not present in the system, or at least not available.

INVALID SCAN ORDERS.

The DCP does not act on scan-out orders in which there is an address/function code parity error or an information parity error. A scan-out function code other than the specified codes is also unacceptable. Any of these conditions causes the DCP to send a scan transmission error signal.

The DCP responds to a scan-in order by sending a scan transmission error signal and an all-zeros information word with a parity error.

INITIALIZE [000].

When recognized by the DCP, an Initialize scan-out turns on the Run

flip-flop and creates a fault interrupt. This fault interrupt takes
precedence over any other fault interrupts. The 20-bit instruction
base address in the scan-out information word is stored in the L
half-word of SM word 0. The interrupt branch address is an all-
zero for main memory. The special stop conditions that might
otherwise prevent the fault actions are inhibited. The first in-
struction word is fetched from the main memory location that is
addressed by the instruction base address.

SET ATTENTION NEEDED [100].
When recognized by a DCP and when the Run flip-flop is on, a Set
Attention Needed scan-out sets the main system attention needed
flip-flop. There is no interference with the other DCP operations.
This scan-out does not cause a fault interrupt.

The DCP does not respond to a Set Attention Needed scan-out when
the Run flip-flop is off. The absence of the ready signal must be
recognized by the main system processor.

NOTE
The Run flip-flop controls
the response so that the
main system processor can
determine if a DCP is run-
ning. The main system pro-
cessor can find out if a DCP
is available in the system by
sending a scan-out Halt.

HALT [010].
When recognized by a DCP, a scan-out Halt creates a fault interrupt
and stops DCP operations (before the fault actions occur) by turning
off the Run and Execute flip-flops.

# SECTION 3
## INSTRUCTIONS

GENERAL.

UNIT TIME.

One unit of time in the DCP is one clock period (200 nanoseconds).

A specified instruction execution time is the minimum number of units required for actual instruction execution before the instruction is released. Most instructions are released simultaneously with the completion of the operations that they order. The following instructions begin operations which may be completed independently after the instruction is released:

      a. Main memory access instructions

      b. Adapter write instructions

A specified instruction execution time does not include the additional time that is required in the following cases:

INDIRECT ADDRESS.

An indirect address designation in the B field increases the specified execution time by one unit. The added time is masked if a main memory access is in process and the actual B address designates the W or MA registers.

MAIN MEMORY ACCESS.

An instruction that addresses the W or MA registers is held up if a main memory access cycle is in progress. The access was ordered by a prior instruction which was released.

ADAPTER WRITE TIMING SPECIFICATION.

Any instruction is held up if the preceding instruction was an Adapter Write instruction (released) and the adapter write operation has not been completed.

UNDEFINED OPERATION CODES.

Operators containing undefined operation codes are not released;
they act like the Idle instruction. The following codes are
exceptions to the above rule:

Table 3-1

Exceptions to Undefined Operation Codes

| Operation Code | Interpreted As | Instruction |
|---|---|---|
| 01000 100 | 01000 000 | Move half-word |
| 01001 000 | 01000 000 | Move half-word |
| 01001 01n | 01000 000 | Move half-word |
| 01100 001 | 01100 000 | Half-word Add |
| 01100 01n | 01100 000 | Half-word Add |
| 01100 101 | 01100 100 | Half-word Subtract |
| 01100 11n | 01100 100 | Half-word Subtract |
| 01101 001 | 01101 000 | Half-word Add, literal in the B field |
| 01101 01n | 01101 000 | Half-word Add, literal in the B field |
| 01101 101 | 01101 100 | Half-word Subtract, literal in the B field |
| 01101 11n | 01101 100 | Half-word Subtract, literal in the B field |
| 01111 01n | 11001 nnn | Exclusive OR |
| 01111 1nn | 11001 nnn | Exclusive OR |

"n" denotes 0 or 1

UNDEFINED ADDRESSES.

Unless an address is specifically detected as invalid, an address
which does not designate an existing location does not interfere

with the instruction execution.  If no source is addressed, read-
out information is all zeros.  If no destination is addressed, no
information is stored.

INVALID INSTRUCTIONS.
The following instruction errors are detected and create an invalid
fault interrupt:

Instructions which address the same location in
scratchpad memory as both a source and a destination.
Exceptions:   Logical OR instructions
Logical AND instructions

INSTRUCTION FORMAT.
The following format is used in this section:

INSTRUCTION
MNEMONIC [OP CODE] [A FIELD] UNIT TIME, GROUP.
Example:
SET CLUSTER MASK REGISTER.
MSKW [00011] [000] 1 unit, Group 0.

GROUP 0 INSTRUCTIONS.
IDLE.
IDLE [00000] [000] Indefinite, Group 0.

A FIELD:
Specified above.

B FIELD:
Not significant.

C FIELD:
Not significant.

This instruction is the DCP idle state.  It is held until the DCP
receives a scan-out Initialize order from the B 6500 processor.
A system master clear signal establishes the all-zero state which
sets up the Idle instruction.

ADAPTER WRITE.

AWI [00010] [0VV] 2 units (if no wait), Group 0.

A FIELD:

000 => Present contents of AC and AI are used.

001 => Present contents of AC used, AI←C then used.

010 => Present contents of AI used, AC←B then used.

011 => AC←B, AI←C then used.

<center>NOTE</center>

<center>Present contents of AA are always<br>used. $AC4$ must be 0 for a write.</center>

B FIELD:

Described above when applicable.

C FIELD:

Described above when applicable.

The AC and AI registers may be loaded with the literals in the B and C fields as specified above. If a register is not loaded with a literal, its present contents are used.

The information in the AC and AI registers and the adapter address in AA [3:4] is sent to the designated cluster. The cluster is ordered to take the information, even if it pre-empts an attention needed condition, and is expected to return either an output accepted or a wait signal. The output accepted signal, signifying that the cluster has accepted the information, causes the program to continue in sequence. The wait signal causes the DCP to wait for the output accepted signal. If neither the output accepted nor the wait signal is present before the second clock, a fault interrupt occurs. The AA, AC and AI registers are not changed. The CF0 and CF1 flip-flops are not affected.

ADAPTER INTERROGATE.

AWRR [00010] [1VV] 3 to 18 units (if no wait), Group 0.

A FIELD:

100 => Present contents of AC and AI are used.

101 => Present contents of AC used, AI←C then used.

110 => Present contents of AI used, AC←B then used.

111 => AC←B, AI←C then used.


B FIELD:

Described above when applicable.


C FIELD:

Described above when applicable.

NOTE

Present contents of AA
always used.  AC4 must
be 1 for an interrogate.

The operation is an Adapter Write combined with a special Adapter
Read.  The information in the Adapter Write is normally some sort
of query and is answered by the read-in information given in re-
sponse.

The control information in the AC and AI registers and the Adapter
Address in AA $[3:4]$ is sent to the designated cluster.  The cluster
is ordered to take the information, even if it pre-empts an atten-
tion needed condition, and is expected to return either an output
accepted or a wait signal.  The output accepted signal, signifying
that the cluster has accepted the information, clears the AC and
AI registers and changes the DCP order to a fixed-address Adapter
Read.  When the cluster responds with the demanded information, the
information  is set into the AC and AI registers, and the program
continues in sequence.  The wait signal causes the DCP to wait for
the output accepted signal.  If neither the output accepted nor the
wait signal is present before the second clock, a fault interrupt
is caused.  If the wait signal goes away before the responsive read
signal occurs, a fault interrupt is also caused.  The CF0 and CF1
flip-flops are not affected.

STOP/BRANCH ON BREAKPOINT.

BKP [00001] 1 unit (if control switch OFF, else 2 units), Group 0.

A FIELD:
Contains the breakpoint code.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

The bits in the A field are compared with the settings of the BREAK-POINT toggle switches on the display panel.  If there is correspondence in all three bit positions, the position of the STOP/BRANCH ON BREAKPOINT control switch determines the resulting action.

If the STOP ON BREAKPOINT/BRANCH ON BREAKPOINT control switch is in the center-OFF position, the instruction is released as a no-op and the program continues in sequence.

If the STOP ON BREAKPOINT/BRANCH ON BREAKPOINT switch is in a control position, the settings of the BREAKPOINT code switches on the display panel are compared with the bits in the A field.  If the switch settings correspond with the A field bits in all three positions, a breakpoint match exists.

<div align="center">NOTE</div>

> The BREAKPOINT code switches have three
> positions: 1, OFF, and 0.  If a switch is
> in the center-OFF position, a match with
> the corresponding A field bit is assured.

If there is a breakpoint match and the control switch is in the STOP ON BREAKPOINT position, the program execution is immediately stopped.

If there is a breakpoint match and the control switch is in the
BRANCH ON BREAKPOINT position, a subroutine entry occurs: simul-
taneously, the contents of the MA registers are stored in the R
half-word of the SM word 0, the contents of the IA register column
are transferred to the MA registers, and the branch address in the
B:C fields is transferred to the IA registers. The instruction is
released. The CF1 and CF0 flip-flops are cleared.

If there is no breakpoint match, the instruction is released and
the program continues in sequence.

SHIFT MA RIGHT.
SHFT [00011] [100] 1+B [2:3] units, Group 0.

A FIELD:
Specified above.

B FIELD:
B [7:2] $\neq$ 01 => B [2:3] specifies the number of shifts desired
minus one.

B [7:2] = 01 => indirect literal; contents of Y register used in
place of B field.

C FIELD:
C [7:2] should be zero (not used).
C [5:3] = MA register interconnection control.
C5 = 1 => Byte circulate (end around on single byte).
C4 = 1 => Shift end around (on selected registers).
C3 = 1 => Concatenate (make connection between MA2, MA1 or MA0).
C [2:3] = Row select code for MA register column.
C2 = 1 => Select MA2
C1 = 1 => Select MA1
C0 = 1 => Select MA0

NOTE
C3 = 1 indicates that two or
more of the shifting registers
are concatenated.

3-7

The MA registers are first selected by C [2:3] and are intercon-
nected as specified by C [5:3]. These selected registers are
shifted right one bit position; this shift is repeated the number
of times specified above by the B field or Y register if selected.

If by selecting combinations of C5, C4 and C3, more than one bit
of information seems to be going to a single bit position, these
input bits are ORed. Also, if no connection to the most signifi-
cant bit position of a shifting register is designated, a zero is
shifted in. The CF0 and CF1 flip-flops are cleared.

Below is a table of the different register concatenations. Byte
circulate is not shown.

Table 3-2

MA Concatenations

| Bit Configuration | | | | | | Shift |
|---|---|---|---|---|---|---|
| C5 | C4 | C3 | C2 | C1 | C0 | |
| 0 | 0 | 1 | 0 | 1 | 1 | MA1:MA0 (":" denotes MA1 [0:1] =⟩ MA0 [7:1]) |
| 0 | 0 | 1 | 1 | 0 | 1 | MA2:MA0 (MA1 bypassed) |
| 0 | 0 | 1 | 1 | 1 | 0 | MA2:MA1 |
| 0 | 0 | 1 | 1 | 1 | 1 | MA2:MA1:MA0 |
| 0 | 1 | 1 | 0 | 1 | 1 | MA1:MA0; MA0 [0:1] =⟩ MA1 [7:1] |
| 0 | 1 | 1 | 1 | 0 | 1 | MA2:MA0; MA0 [0:1] =⟩ MA2 [7:1] |
| 0 | 1 | 1 | 1 | 1 | 0 | MA2:MA1; MA1 [0:1] =⟩ MA2 [7:1] |
| 0 | 1 | 1 | 1 | 1 | 1 | MA2:MA1:MA0; MA0 [0:1] =⟩ MA2 [7:1] |
| 0 | 1 | 0 | 0 | 1 | 1 | MA1, MA0; MA0 [0:1] =⟩ MA1 [7:1] |
| 0 | 1 | 0 | 1 | 0 | 1 | MA2, MA0; MA0 [0:1] =⟩ MA2 [7:1] |
| 0 | 1 | 0 | 1 | 1 | 0 | MA2, MA1; MA1 [0:1] =⟩ MA2 [7:1] |
| 0 | 1 | 0 | 1 | 1 | 1 | MA2, MA1, MA0; MA0 [0:1] =⟩ MA2 [7:1] |

SET SYSTEM INTERRUPT.
HEYU [00011] [101] 5 units, Group 0.

A FIELD:

Specified above.

B FIELD:

[00000011] Specifies a repeat count to hold the interrupt.
B [7:2] = 01 =〉 7 indirect literal; Y is used in place of the B
field.

C FIELD:

[00000000] (reserved).


This instruction sends an interrupt signal on a special control lead
to the peripheral control multiplexor.  The repeat number in the B
field holds the interrupt signal long enough for the multiplexor to
recognize it (five units).  Then the DCP continues in sequence.


SET CLUSTER MASK REGISTER.
MSKW [00011] [000] 1 unit, Group 0.

A FIELD:

Specified above.

B FIELD:

Not significant.

C FIELD:

Not significant.


This instruction places the AC register contents into the left byte
of the CM Register, CM [15:8], and the AI register contents into
the right byte of the CM register, CM [7:8].  The parity bit is
not used.  The CF0 and CF1 flip-flops are not affected.

A one bit in a position of the Cluster Mask register causes the DCP
cluster interface control logic to recognize a cluster attention

needed signal from the corresponding Adapter Cluster. A zero bit prevents the recognition so the DCP will not respond to a cluster attention needed signal. The state of the Cluster Mask register affects the Adapter Read instructions indirectly, but does not affect the Adapter Write or Adapter Interrogate instructions.

READ CLUSTER MASK REGISTER.
MSKR [00011] [001] 1 unit, Group 0.

A FIELD:
Specified above.

B FIELD:
Not significant.

C FIELD:
Not significant.

This instruction places the contents of the left byte of the CM register, CM [15:8] , into the AC register and the contents of the right byte of the CM register, CM [7:8], into the AI register. The parity bit (I21) is cleared. AA is set to 0. The CF0 and CF1 flip-flops are not affected.

GROUP 1 INSTRUCTIONS.
ADAPTER READ WHEN ATTENTION NEEDED.
ARWN [00100] [000] 2 to indefinite, Group 1.

A FIELD:
Specified above.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

NOTE

AA [7:2] designates the cluster
group which may be read first.

When a cluster attention needed (CAN) condition occurs, the cluster
exchange control section finds the cluster which is calling for
attention.  Then the AA, AC, and AI registers are set from the
selected cluster interface signals  and the program continues in
sequence.  The CFO and CF1 flip-flops are not affected.

If only a system attention needed (SAN) condition occurs (no CAN),
the contents of the B and C fields are transferred to the IA reg-
isters and a branch is taken.  The SAN flip-flop is reset and the
CFO and CF1 flip-flops are cleared.

If there is neither a CAN nor SAN condition, the instruction is held
indefinitely.  When they occur simultaneously, the CAN is serviced
first.

NOTE

AA [7:2] designates one of four cluster
cables or cluster groups.  When a cluster
attention needed condition occurs in a
cluster group other than the one being
designated, the cluster exchange control
section sets AA [7:2] to designate a
group with attention needed.  Indepen-
dently, a cluster which needs attention
within that group has been found by
counting up a separate two-bit register.
This register retains the number of one
cluster which last had an attention
needed condition in the group.  The
initial contents of AA [5:2] are not
used in seeking the cluster.

ADAPTER READ IF NEEDED.
ARIN [00100] [001] 1 to 3 units, Group 1.

A FIELD:
Specified above.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

NOTE

AA [7:2] designates the cluster
group which may be read first.

If a cluster attention needed (CAN) condition exists, the cluster
exchange control section identifies a cluster which is calling for
attention.  The AA, AC, and AI registers are set from the selected
cluster interface signals, and a branch to the address in the B
and C fields is taken.  The CF0 and CF1 flip-flops are cleared.

If there is no CAN condition, the program continues in sequence.
The CF0 and CF1 flip-flops are not affected.

The program loop timer is reset by this instruction.

DECREMENT Y, BRANCH IF Y IS ZERO.
DBYZ [00100] [010] 1 unit, Group 1.

A FIELD:

Specified above.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

If Y is zero, the branch address in the B:C field is transferred to the IA registers and the program continues.  If Y is non-zero the program continues in sequence.

After the test, the Y register contents are always decremented by one if they are not already zero.

If the branch occurs, the CF0 and CF1 flip-flops are cleared.

DECREMENT Y, BRANCH IF Y IS NON-ZERO.
DBYN [00100] [011] 1 unit, Group 1.

A FIELD:
Specified above.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

If Y is non-zero, the branch address in the B:C field is transferred to the IA registers and the program continues.  If Y is zero, the program continues in sequence.  After the test, the Y register contents are always decremented by one if it is not already zero.

If the branch occurs, the CF0 and CF1 flip-flops are cleared.

BRANCH IF SYSTEM ATTENTION NEEDED.
BRAN [00100] [100] 1 unit, Group 1.

A FIELD:
Specified above.

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

If the System Attention Needed(SAN) flip-flop is on, the instruc-
tion address in the B and C fields are transferred to the IA
registers and the branch is taken.  The SAN flip-flop is reset and
the CF0 and CF1 flip-flops are cleared.

If the SAN flip-flop is off, the program continues in sequence and
CF0 and CF1 are not affected.

BRANCH.
GOTO [00101] 1 unit, Group 1.

A FIELD:

| A2 | A1 | A0 | BRANCH CONDITION | RESULT SUBTRACTION |
|----|----|----|------------------|--------------------|
| 0 | 0 | 0 | No branch | N/A |
| 0 | 0 | 1 | CF1=0 and CF0=0 | A>B |
| 0 | 1 | 0 | CF0=1 | A<B |
| 0 | 1 | 1 | CF1=0 | A≠B |
| 1 | 0 | 0 | CF1=1 | A=B |
| 1 | 0 | 1 | CF0=0 | A≥B |
| 1 | 1 | 0 | CF1=1 or CF0=1 | A≤B |
| 1 | 1 | 1 | Unconditional branch | N/A |

B FIELD:
Contains a B:C branch address (see B:C branch address).

3-14

C FIELD:

Contains a B:C branch address (see B:C branch address).

The contents of the A field cause the compare flip-flops to be
checked as above.  If the specified compare condition is satisfied,
the branch is effected by transferring the contents of the B and C
fields to the IA registers.  If the compare condition is not satis-
fied, program control continues in sequence.  CF0 and CF1 are
cleared if the branch occurs.

BRANCH RELATIVE WITH X.
GOX [00110] 1, 2 or 3 units, Group 1.

A FIELD:

| A2 | A1 | A0 | BRANCH CONDITION | SUBTRACTION RESULTS |
|----|----|----|------------------|---------------------|
| 0  | 0  | 0  | No branch | N/A |
| 0  | 0  | 1  | CF1=0 and CF0=0 | A>B |
| 0  | 1  | 0  | CF0=1 | A<B |
| 0  | 1  | 1  | CF1=0 | A≠B |
| 1  | 0  | 0  | CF1=1 | A=B |
| 1  | 0  | 1  | CF0=0 | A≥B |
| 1  | 1  | 0  | CF1=1 or CF0=1 | A≤B |
| 1  | 1  | 1  | Unconditional branch | N/A |

B FIELD:

Contains a B:C branch address (see B:C branch address).

C FIELD:

Contains a B:C branch address (see B:C branch address).

The contents of the A field cause the compare flip-flops to be
checked as above.  If the specified compare condition is satisfied,
the branch is effected by transferring the sum of the X register
and the B and C fields to the IA registers.  If the compare condit-
ion is not satisfied, program control continues in sequence.  If

the branch occurs, CFO and CF1 are cleared.

The above address addition does not disturb CFO or CF1.

BRANCH RELATIVE INDIRECT.
GOI [00111] 1, 2 or 3 units, Group 1.

A FIELD:

| A2 | A1 | AO | BRANCH CONDITION | SUBTRACTION RESULT |
|----|----|----|------------------|--------------------|
| O | O | O | No Branch | N/A |
| O | O | 1 | CF1=0 and CFO=0 | A>B |
| O | 1 | O | CFO=1 | A<B |
| O | 1 | 1 | CF1=0 | A≠B |
| 1 | O | O | CF1=1 | A=B |
| 1 | O | 1 | CFO=0 | A≥B |
| 1 | 1 | O | CF1=1 or CFO=1 | A≤B |
| 1 | 1 | 1 | Unconditional Branch | N/A |

B FIELD:
Contains a B:C branch address (see B:C branch address).

C FIELD:
Contains a B:C branch address (see B:C branch address).

NOTE
The Y Register contains a byte
address (see 8-bit address).

The contents of the A field cause the compare flip-flops to be
checked as above. If the specified compare condition is satisfied,
the branch is effected by transferring the sum of the contents of
the Y register byte address and the B and C fields to the IA
registers. If the compare condition is not satisfied, program
control continues in sequence. If the branch occurs, CFO and CF1
are cleared.

The above address addition does not disturb CFO or CF1.

3-16

GROUP 2 INSTRUCTIONS.

MOVE HALF-WORD.

MOVE [01000] [000] 1 unit, Group 2.

A FIELD:

Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a half-word destination address (see 8-bit address).

NOTE

The C field should not address the
MA registers as a destination be-
cause the MA registers are impli-
citly addressed as the destination.

The information specified by the B field is transferred to the MA
register.  If the B field contains zeros, then zeros are transferred
to the MA registers.  If the B field addresses the MA registers, the
contents of the MA registers are retained.

Simultaneously, the contents of the MA registers are transferred to
the destination addressed by the C field.  If the C field is all
zeros no information is transferred.

LOCAL MEMORY READ, INDIRECT ADDRESS.

LMRI [01001] [001] 3 units (if no system wait), Group 2.

A FIELD:

Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create an
address for local memory.  Then this address is stored in MA-1.
MA-0 is used to obtain an entire word (including tag) from local
memory which is stored in the W register.  Parity is checked.

If memory access is not obtained within eight clock times, an in-
valid address fault interrupt is created.

<div align="center">

NOTE

MA-1 [7:2] = 00.

MA [11:12] contains local

memory address.

</div>

SCRATCHPAD MEMORY READ.
SMRD [01000] [010] 1 unit (if no system wait), Group 2.

A FIELD:
Specified above.

B FIELD:
Contains a half-word source scratchpad memory address or register
address.

C FIELD:
Contains a half-word destination address (see 8-bit address).

The B field may select either the L half-word of scratchpad memory
or a register column half-word.  If scratchpad memory L half-word
is selected, the related full word is transferred to the W register.
When a register column is selected, it is transferred to W-L and
W-R will receive zeros.  If B is zero, the W register will receive
all zeros.  Parity is not checked.

Simultaneously, the contents of the MA registers are sent to the
half-word selected by the C field.  If C is zero there is no infor-
mation transfer.

If C selects W-R, MA is
ORed with the information
selected by the B field.

LOCAL MEMORY READ, DIRECT ADDRESS.
LMRD [01000] [011] 3 units (if no system wait), Group 2.

A FIELD:
Specified above.

B FIELD:
B [7:2] = 00.
B [5:6]; C [7:8] contain a local memory source address right justified.

C FIELD:
Specified above.

The B and C fields are transferred to the MA-1 and MA-0 registers, and simultaneously, the MA-1 and MA-0 registers are transferred temporarily to the B and C fields of the Instruction register.  The LM read cycle occurs after the address transfer.  The read-out information is loaded into the W register, MA-1 and MA-0 are restored from the B and C fields, and the W register parity check control is set.  If there is a parity error, a data parity error fault interrupt is created before the next instruction is executed.

If memory access is not obtained within eight clock times, an invalid address fault interrupt is created.  The LM word address is in the MA1 and MA0 registers.

The B:C address selects a full word in local memory which is read into the W register.  Parity is checked.

LOCAL MEMORY WRITE, INDIRECT ADDRESS.
LMWI [01000] [101] 3 units (if no system wait), Group 2.

A FIELD:

Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create an address for local memory. MA [23:10] = 0 and MA [11:10] contain the local memory address. Then the full word (including tag) in the W register is stored into the location selected by the MA register. Odd parity is generated.

SCRATCHPAD MEMORY WRITE.
SMWD [01000] [110] 1 unit (if no system wait), Group 2.

A FIELD:

Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a scratchpad memory address or register address half-word destination.

The information selected by the B field is transferred to the MA registers. Zeros are transferred to the MA registers if B is zero. If the C field addresses scratchpad memory, that word of scratchpad memory is loaded from the W register. Parity is not generated.

If C selects a register column (half-word) the contents of W-L are transferred to the selected registers column. W-R is not transferred. If C is zero, there is no information transfer.

LOCAL MEMORY WRITE, DIRECT ADDRESS.
LMWD [01000] [111] 3 units (if no system wait), Group 2.

A FIELD:

Specified above.

B FIELD:

B [7:2] = OO.
B [5:6]; C [7:8] contain a local memory destination address right justified.

C FIELD:

Specified above.

Using the above B and C addresses, the full word (with tag) is transferred from the W register to local memory.  Odd parity is generated.

MAIN MEMORY READ.

MMR [01001] [001] 1 unit, Group 2.

A FIELD:

Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create the main memory address in the MA registers.  A main memory access is requested after transferring the address to the B 6500 Memory Address register.  When the access is granted the addressed information in main memory is loaded into the W register.  This access of main memory occurs independently of other DCP operations so that the DCP may continue to execute code.

NOTE
Execution of any instruction
accessing main memory or the

W register will be delayed un-
til this instruction is completed.

MAIN MEMORY WRITE UNCONDITIONAL.
MMWU [01001] [100] 1 unit, Group 2.

A FIELD:
Specified above.

B FIELD:
Contains a half-word source address (see 8-bit address).

C FIELD:
Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create
a main memory address in the MA registers.  A main memory write is
requested after transferring MA [19:20] to the B 6500 Memory
Address register and generates odd parity for the word in the W
register.  When the access is granted, the information in the W
register is unconditionally (no check of protect bit for word
currently in main memory) written into the addressed main memory
location.

This write continues independently of other DCP operations so that
the DCP may continue to execute code.

NOTE
Execution of any instruction
accessing main memory or the
W register will be delayed
until this instruction is
completed.

MAIN MEMORY WRITE-READ UNCONDITIONAL.
MMWR [01001] [101] 1 unit, Group 2.

A FIELD:
Specified above.

B FIELD:

Contains a half-word source address (see 8-bit address).

C FIELD:

Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create a main memory address in the MA registers.  A main memory write is requested after transferring MA [19:20] to the B 6500 Memory Address register and generating odd parity for the word in the W register. When the access is granted the information in the W register is unconditionally (no check of protect bit for word currently in main memory) written into the addressed main memory location. After the write, the previous contents of the addressed main memory location are loaded into the W register.

This write-read continues independently of other DCP operations so that the DCP may continue to execute code.

NOTE

Execution of any instruction
accessing main memory or the
W register will be delayed
until this instruction is
completed.

MAIN MEMORY PROTECTED WRITE.
MMWP [01001] [110] 1 unit, Group 2.

A FIELD:
Specified Above.

B FIELD:
Contains a half-word source address (see 8-bit address).

C FIELD:
Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create a main memory address in the MA registers.  A main memory write is requested after transferring MA [19:20] to the B 6500 Memory Address register and generating odd parity for the word in the W register.  When the access is granted, the information in the W register is written into the addressed main memory location only if the protect bit of the word currently in that location is zero (not protected).  If the protect bit is one a fault interrupt will occur (see fault interrupt).

This write continues independently of other DCP operations so that the DCP may continue to execute code.

<div align="center">NOTE</div>

Execution of any instruction accessing main memory or the W register will be delayed until this instruction is completed.

MAIN MEMORY WRITE-READ PROTECTED.
MWRP [01001] [111] 1 unit, Group 2.

A FIELD:
Specified above.

B FIELD:
Contains a half-word source address (see 8-bit address).

C FIELD:
Contains a half-word destination address (see 8-bit address).

This instruction initiates a Move Half-word instruction to create a main memory address in the MA registers.  A main memory write is requested after transferring MA [19:20] to the B 6500 Memory Address register and generates odd parity for the word in the W register.  When the access is granted the information in the W register is written into the addressed main memory location only

if the protect bit of the word currently in that location is zero (not protected). If the protect bit is one, a fault interrupt will occur (see Fault Interrupt). Whether the protect bit is on or not the previous contents of the addressed main memory location will be loaded into the W register. This Write Read continues independently of other DCP operations so that the DCP may continue to execute code.

NOTE

Execution of any instruction
accessing main memory or the
W register will be delayed
until this instruction is
completed.

## GROUP 3 INSTRUCTIONS.

HALF-WORD ADD.

HAD [01100] [000] 3 units, Group 3.

A FIELD:
Specified above.

B FIELD:
Contains a half-word operand address (see 8-bit address).

C FIELD:
Contains a half-word destination address (see 8-bit address).

The half-word selected by the B field address is added to the contents of the MA registers with the sum stored in the half-word designated by the C field address.

The result of the half-word addition controls the CFO and CF1 flip-
flops:

        Carry:    set CFO        Sum zero:     set CF1
   No Carry:    clear CFO     Sum non-zero:    clear CF1

HALF-WORD ADD, LITERAL IN B.
HADB [01101] [000] 3 units, Group 3.

A FIELD:
Specified above.

B FIELD:
Contains a literal operand.

C FIELD:
Contains a half-word destination address (see 8-bit address).

The literal in the B field is added to the contents of the MA reg-
isters and the sum is stored in the half-word designated by the
C field address.

The result of the half-word addition controls the CFO and CF1 flip-
flops:

        Carry:    set CFO        Sum zero:     set CF1
   No Carry:    clear CFO     Sum non-zero:    clear CF1

HALF-WORD SUBTRACT.
HSB [01100] [100] 3 units, Group 3.

A FIELD:
Specified above.

B FIELD:
Contains a half-word operand address (see 8-bit address).

C FIELD:
Contains a half-word destination address (see 8-bit address).

The half-word selected by the B field address is subtracted from the contents of the MA registers, and the remainder is stored in the half-word designated by the C field address.

The result of the half-word subtraction controls the CF0 and CF1 flip-flops:

        Borrow:    set CF0          Remainder zero:      set CF1
    No borrow:    clear CF0     Remainder non-zero:    clear CF1

HALF-WORD SUBTRACT, LITERAL IN B.
HSBB [01101] [100] 3 units, Group 3.

A FIELD:
Specified above.

B FIELD:
Contains a literal operand.

C FIELD:
Contains a half-word destination address (see 8-bit address).

The literal in the B field is subtracted from the contents of the MA register half-word and the remainder is stored in the half-word designated by the C field.

The result of the half-word subtraction controls the CF0 and CF1 flip-flops:

        Borrow:    set CF0              Remainder zero:      set CF1
    No borrow:    clear CF0         Remainder non-zero:    clear CF1

TRANSLATE.
TRAN [01110] 2 units, Group 3.

A FIELD:

| A2 | A1 | AO | Translation |
|----|----|----|-------------|
| O | O | O | EBCDIC to USASCII |
| O | O | 1 | EBCDIC to BCL (Burroughs Common Language) |
| O | 1 | O | (reserved) |
| O | 1 | 1 | (reserved) |
| 1 | O | O | USASCII to EBCDIC |
| 1 | O | 1 | BCL to EBCDIC |
| 1 | 1 | O | (reserved) |
| 1 | 1 | 1 | (reserved) |

B FIELD:

Contains a byte operand address (see 8-bit address).

C FIELD:

Contains a byte destination address (see 8-bit address).

The byte addressed by the B field is routed into the translator specified by the A field with the output byte from the translator sent to the destination specified by the C field.

The translation is always a one-to-one conversion. Addition or deletion of characters must be handled by the program. Other code translations, such as EBCDIC - Baudot, must be performed by a program.

NOTE

The hardware implementation of the translate feature is optional. See Appendix B for more detail.

PARITY.

PARY [01111] [OOV] 1 unit, Group 3.

A FIELD:

A=OOO => check or generate even parity.

A=OO1 => check or generate odd parity.

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a byte destination address (see 8-bit address).

The source operand addressed by the B field is transferred to the
destination specified by the C field.  The bits of the source
operand (including the 9th bit of AI (parity bit) if AI is the
selected source) and the AO bit are exclusively ORed together.  The
CF1 flip-flop is controlled by the outcome of the Exclusive OR:

Even number of one-bits, result zero:  set CF1 (good parity).
Odd number of one-bits, result one:  Clear CF1 (parity error).

The CFO flip-flop is always cleared.

If the AI register is selected as the destination, the 9th bit of
AI (I21) is also controlled by the result of the Exclusive OR:

Odd number of one-bits, result one:  set I21.
Even number of one-bits, result zero:  clear I21.

NOTE
If the AI register is both
the source and the destination,
it is assumed that AI [8:1] = 0 initially.

GROUP 4 INSTRUCTIONS.
ADD.
ADD [11101] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a byte destination address (see 8-bit address).

The operands addressed by the A and B fields and the carry bit
CFO are added with the sum transferred to the destination specified
by the C field.  The operation is commutative.

The output carry signal controls the CFO flip-flop, and the sum
controls the CF1 flip-flop:

                    Carry:      set CFO            Sum zero:        set CF1
          No carry:      clear CFO      Sum non-zero:      clear CF1

ADD, LITERAL IN B.
ADDB [11110] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a literal operand.

C FIELD:
Contains a byte destination address (see 8-bit address).

The operand addressed by the A field, the literal in the B field,
and the carry bit in CFO are added with the sum transferred to the
destination specified by the C field.  Otherwise, the operation is
the same as ADD.

ADD, LITERAL IN C.
ADDC [11111] 1 unit, Group 4.

A FIELD:
Contains an A field destination address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains literal operand.

The operand addressed by the B field, the literal in the C field, and the carry bit in CFO are added with the sum transferred to the register specified by the A field.  Otherwise, the operation is the same as ADD.

SUBTRACT.
SUB [10000] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a byte destination address (see 8-bit address).

The subtrahend addressed by the B field and the borrow bit in CFO are subtracted from the minuend addressed by the A field with the remainder transferred to the destination specified by the C field. The operation is not commutative.

The output borrow signal controls the CFO flip-flop, and the remainder controls the CF1 flip-flop:

    Borrow:    Set CFO        Remainder zero:    Set CF1
 No borrow:  Clear CFO   Remainder non-zero:    Clear CF1

        CF1 = And CFO = 0....A>B
        CFO = 1..............A<B
        CF1 = 0..............A≠B
        CF1 = 0..............A=B
        CFO = 0..............A≥B
        CF1 = 1 OR CFO = 1...A≤B

SUBTRACT, LITERAL IN B.
SUBB [11000] 1 unit, Group 4.

A FIELD:

Contains an A field destination address (see A field address).

B FIELD:

Contains a literal operand.

C FIELD:

Contains a byte operand address (see 8-bit address).

The literal in the B field and the borrow bit in CFO are subtracted from the minuend addressed by the A field with the remainder transferred to the destination specified by the C field.  Otherwise, the operation is the same as SUB.

SUBTRACT, LITERAL IN C.
SUBC [11100] 1 unit, Group 4.

A FIELD:

Contains an A field destination address (see A field address).

B FIELD:

Contains a byte operand address (see 8-bit address).

C FIELD:

Contains a literal operand.

The subtrahend addressed by the B field and the borrow bit in CFO are subtracted from the literal in the C field with the remainder transferred to the destination specified by the A field.  Otherwise, the operation is the same as SUB.

LOGICAL AND.
LAN [10101] 1 unit, Group 4.

A FIELD:

Contains an A field operand address (see A field address).

B FIELD:

Contains a byte operand address (see 8-bit address).

3-32

C FIELD:

Contains a byte destination address (see 8-bit address).

The corresponding bits of the operands addressed by the A and B fields are logically ANDed together individually with the result transferred to the destination specified by the C field.  The AND function has the following relationship:

| Operands | | Result |
|---|---|---|
| (A) [n:1] | (B) [n:1] | (C) [n:1] |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

There is no interaction between different bit positions.  The operation is commutative.

The CF0 flip-flop is cleared, but the result controls the CF1 flip-flop:

All result bits zero:     set CF1
Not all result bits zero:     clear CF1

LOGICAL AND, LITERAL IN B.
LANB [10110] 1 unit, Group 4.

A FIELD:
Contains an A field source address (see A field address).

B FIELD:
Contains a literal operand.

C FIELD:
Contains a byte operand address (see 8-bit address).

The bits of the operand addressed by the A field and the corresponding bits of the literal operand in the B field are logically ANDed

together with the result transferred to the destination specified
by the C field.  Otherwise, the operation is the same as LAN.

LOGICAL AND, LITERAL IN C.
LANC [10111] 1 Unit, Group 4.

A FIELD:
Contains an A field destination address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a literal operand.

The bits of the operand addressed by the B field and the correspon-
ding bits of the literal operand in the C field are logically ANDed
with the result transferred to the register specified by the A field.
Otherwise, the operation is the same as LAN.

LOGICAL AND-OR MA, LITERAL IN C.
LAOM [10100] 1 unit, Group 4.

A FIELD:
Contains an A field destination address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a literal operand.

The bits of the operand addressed by the B field are logically ANDed
individually with the corresponding bits of the literal in the C
field in the same manner as in LANC.  This AND result is ORed with
the contents of the MA byte register which is in the same row as
the destination (A field).  The combined result is transferred to the
destination specified by the A field.

NOTE

    If the destination is an MA
register, the result is ORed
into the selected MA register
without losing any one bits.
This function facilitates pack-
ing information into the MA re-
gisters.

LOGICAL OR.
LOR [10001] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a byte address operand (see 8-bit address).

C FIELD:
Contains a byte destination address (see 8-bit address).

The corresponding bits of the operands addressed by the A and B
fields are logically ORed together individually with the result
transferred to the destination specified by the C field.  The
OR function has the following relationship:

| Operands | | Result |
| --- | --- | --- |
| (A) [N:1] | (B) [n:1] | (C) [n:1] |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

There is no interaction between different bit positions.  The opera-
tion is commutative.

The CF0 flip-flop is cleared, but the result controls the CF1 flip-
flop:

All result bits zero:  set CF1
Not all result bits zero:  clear CF1

NOTE

When only one operand is
addressed, the operator
is in effect a "move
byte" operator.

LOGICAL OR, LITERAL IN B.
LORB [10010] 1 unit, Group 4.

A FIELD:
Contains an A field Operand address (see A field address).

B FIELD:
Contains a literal operand.

C FIELD:
Contains a byte destination address (see 8-bit address).

The bits of the operand addressed by the A field and the correspon-
ding bits of the literal in the B field are logically ORed with
the result transferred to the destination specified by the C field.
Otherwise, the operation is the same as LOR.

When the A field is empty, the operator is in effect a "load byte"
operator.

LOGICAL OR, LITERAL IN C.
LORC [10011] 1 unit, Group 4.

A FIELD:
Contains an A field destination address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a literal operand.

The bits of the operand addressed by the B field and the correspon-
ding bits of the literal in the C field are logically ORed with
the result transferred to the register specified by the A field.
Otherwise, the operation is the same as LOR.

LOGICAL EXCLUSIVE OR.
LEO [11001] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a byte destination address (see 8-bit address).

The corresponding bits of the operands addressed by the A and B
fields are logically exclusively ORed together individually with
the result transferred to the destination specified by the C field.
The Exclusive OR function has the following relationship:

| Operands | | Result |
|---|---|---|
| (A) [n:1] | (B) [n:1] | (C) [n:1] |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

There is no interaction between different bit positions.  The
operation is commutative.

The CF0 flip-flop is cleared, but the result controls the CF1 flip-
flop:

> All result bits zero:    set CF1
> Not all result bits zero:    clear CF1

Exclusive OR may be used for generating longitudinal check characters and negating individual bits.

LOGICAL EXCLUSIVE OR, LITERAL IN B.
LEOB [11010] 1 unit, Group 4.

A FIELD:
Contains an A field operand address (see A field address).

B FIELD:
Contains a literal operand.

C FIELD:
Contains a byte destination address (see 8-bit address).

The bits of the operand addressed by the A field and the corresponding bits of the literal in the B field are logically exclusively ORed with the result transferred to the destination specified by the C field. Otherwise, the operation is the same as LEO.

LOGICAL EXCLUSIVE OR, LITERAL IN C.
LEOC [11011] 1 unit, Group 4.

A FIELD:
Contains an A field destination address (see A field address).

B FIELD:
Contains a byte operand address (see 8-bit address).

C FIELD:
Contains a literal operand.

The bits of the operand addressed by the B field and the corresponding bits of the literal in the C field are logically exclusively ORed with the result transferred to the register specified by the A field. Otherwise, the operation is the same as LEO.

# SECTION 4
## DCP DISPLAY PANEL AND MAINTENANCE AIDS

GENERAL.

DISPLAY PANEL (figure 4-1).

The DCP display panel is mounted in the same cabinet as the DCP. The display panel and DCP are interconnected with six cables. The DCP display panel has various switches and pushbutton indicators which are shared for flip-flop display.

DISPLAY MODE.

The DCP is in a display mode when the RUN flip-flop is off. In the display mode, the Display Select rotary switches select the registers or SM half-words which are displayed. The bits in a selected register or SM half-word can be manually set or reset.

OPERATING MODE.

The DCP is operating when the Run flip-flop is on. In the operating mode, the Display Select switches are not effective and bits cannot be manually set or reset.

INSTRUCTION DISPLAY.

In the display mode, the L half-word of the Instruction register (P-L) is normally connected to the 24-bit instruction display. A P-L flip-flop can be set or reset by pressing the corresponding pushbutton while holding down the BIT SET or BIT RESET button.

> Exception: If the Display Type switch is in the P-R position, the R half-word of the Instruction register (P-R) is displayed in place of the L half-word. The P-R flip-flops cannot be manually set or reset.

In the operating mode, the R half-word of the Instruction register (P-R) is connected to the instruction display. The displayed information may be misleading if the instruction being executed is held for a long time; for example, an Adapter Read When Attention Needed.

OP

A

B

C

47  46  45  44  43  42  41  40  39  38  37  36  35  34  33  32  31  30  29  28  27  26  25  24

TAG

51  50  49  48

ROW 2 BYTE 0 (3)

EX  RUN  FLT

ROW 1 BYTE 1 (4)

CAN  MCYN  MACC  PYPC  PYWC  SCM  SC.C  SC.B  SC.A

ROW 0 BYTE 2 (5)

23  22  21  20  19  18  17  16  15  14  13  12  11  10  09  08  07  06  05  04  03  02  01  00

47  46  45  44  43  42  41  40  39  38  37  36  35  34  33  32  31  30  29  28  27  26  25  24

BIT SET    CM REG

OFF LINE    HOLD P-L    STOP IMMED    STOP    A2    A1    A0

REG          SM-L

P-R          SM-R

MA     3         4    IA

DYX    2

ACI    1         5    W-L

        6    W-R

BIT RESET    CLEAR    ON LINE    STOP ON FAULT    STOP ON FETCH    BRANCH    1 OFF 0    PULSE

0

7

BREAKPOINT

DISPLAY ADDRESS

Figure 4-1.   DCP Display Panel

<u>DATA DISPLAY</u>.

In the display mode, the register column or SM half-word which is designated by the Display Select rotary switches is connected to the 24-bit data display. A bit in the selected register or SM half-word can be manually set or reset by pressing the corresponding pushbutton while holding down the BIT SET or BIT RESET button.

> Exception: If the Display Type switch is in the P-R position, nothing is connected to the data display.

In the operating mode, the D bus is connected to the data display and the pushbuttons are disabled.

<u>TAG DISPLAY</u>.

In the display mode, the tag field of a full word is connected to the four-bit tag display when an R half-word is selected. The bits can be manually set or reset.

> Exception: If the Display Type switch is in the P-R position, the tag field of the Instruction register is displayed but the flip-flops cannot be manually set or reset.

<u>CONTROL DISPLAY</u>.

The state of some of the significant control flip-flops is shown by 12 control display indicators. The Execute flip-flop (EX) is the only flip-flop that can be manually set or reset (display mode only). The other pushbuttons are not connected.

EX     Execute flip-flop. When on, signifies that the L half-word of the Instruction register contains an instruction to be executed.

RUN     Run flip-flop. When on, orders DCP operation.

FLT     Fault flip-flop. Set by a fault condition and, when on, orders fault action.

CAN      Cluster attention needed.  Set by any selected cluster at-
         tention needed signal; cleared when there is no cluster at-
         tention needed signal.

MCYN     Main Memory Cycle Needed flip-flop.   Set when any main mem-
         ory access is needed.

MACC     Main Memory Access flip-flop.   Set when any main memory is
         active.

PYPC     Parity P Register Check flip-flop.   Set when the P register
         is loaded.

PYWC     Parity W Register Check flip-flop.   Set when the W register
         is loaded from a local memory access.

SCM      Local Memory Sequence Control flip-flop.   When on, selects
         the address in the MA registers for a local memory data
         access.

SC.C     Sequence Count "C" flip-flop, ORed.   Indicates the C bit
         of either of two sequence counters, of which only one can
         be non-zero at a time.

SC.B     Sequence Count "B" flip-flop, ORed.   Indicates the B bit
         of any of three sequence counters, of which only one can
         be non-zero at a time.

SC.A     Sequence Count "A" flip-flop, ORed.   Indicates the A bit
         of any of three sequence counters, of which only one can
         be non-zero at a time.

OFF-LINE/ON-LINE SWITCH.
When the switch is in the OFF LINE position, the DCP appears not
ready to any scan-out order from the main system.  The other DCP
operations are not affected.  Main memory access requests are not
inhibited, and the DCP can set the DCP Attention Needed flip-flop
in the multiplexor.

4-4

When the switch is in the ON LINE position, the DCP operates nor-
mally with the main system. For completely normal on-line opera-
tion, all control switches must be off.

BIT SET BUTTON.
When the BIT SET button is held down, the indicator pushbuttons
are enabled to act in the display mode as bit set buttons.

BIT RESET BUTTON
When the BIT RESET button is held down, the indicator pushbuttons
are enabled to act in the display mode as bit reset buttons.

CM REGISTER BUTTON.
When depressed, the CM REG button transfers control of the 16 flip-
flop Cluster Mask register to the BIT SET and BIT RESET buttons.

CLEAR BUTTON.
When depressed, the CLEAR button clears all flip-flops in the DCP.
(The scratchpad memory is not cleared).

HOLD P-L/OFF/STOP ON FAULT SWITCH.
When the switch is in the HOLD P-L position, the instruction in
the L half of the Instruction register is held indefinitely. After
the Execute flip-flop is set, it is not reset. (Exception: a stop
condition in Stop on Breakpoint.) Fetch cycles and IA count-up are
inhibited. A fault condition is ignored.

When the switch is in the STOP ON FAULT position, the RUN flip-
flop is cleared if a fault condition occurs. After a stop, the
Execute (EX) and Fault (FLT) flip-flops are on, and P-L contains
the special fault instruction.

STOP IMMED/OFF/STOP ON FETCH SWITCH.
When the switch is in the STOP IMED position, the RUN flip-flop is
cleared by every clock pulse, and the DCP operates in a single
pulse mode.

Exceptions:    The RUN flip-flop is not cleared during the
following operations:

    a.   Cluster intercommunication in process.
    b.   LM data access in process.
    c.   Branch Relative addition in process.
    d.   Fetch cycle in process.

When the switch is in the STOP ON FETCH position, the RUN flip-flop
is cleared whenever the next instruction in sequence is loaded in
the L half of the Instruction register.  After a stop, the Execute
(EX) flip-flop is on and the new instruction is ready for execution.

BREAKPOINT CONTROL.
STOP/OFF/BRANCH SWITCH.
When the switch is in the STOP on BREAKPOINT position, the RUN
flip-flop and the Execute flip-flop (EX) are cleared by the execu-
tion of a breakpoint instruction in which the A field matches the
setting of the breakpoint code switches.  The next fetch is inhi-
bited and the Breakpoint instruction is left in the Instruction
register.

When the switch is in the BRANCH on BREAKPOINT position, the fol-
lowing "enter subroutine" actions take place after the execution
of a Breakpoint instruction in which the A field matches the set-
ting of the breakpoint code switches:

    a.   The contents of the MA registers are stored in the R
        half-word of SM word 0.

    b.   The next instruction address in the IA registers is trans-
        ferred to the MA registers.

    c.   The branch address in the B:C fields is transferred to the
        IA registers.

If the branch condition is not satisfied, the next instruction in
sequence is fetched.

A2, A1, A0; 1/OFF/0 CODE SWITCHES.

When a switch is in the 1 position, the corresponding bit of the A field of a Breakpoint instruction must be a one to establish coincidence. When a switch is in the 0 position, the corresponding bit of the A field must be a zero to establish coincidence. When a switch is in the OFF position, coincidence with the corresponding bit of the A field is assured. Coincidence in all three positions allows breakpoint action to take place as directed by the Breakpoint Control switch.

PULSE SWITCH.

When the PULSE button is pressed, a single pulse which sets the RUN flip-flop is emitted.

DISPLAY TYPE SWITCH.

The four position DISPLAY TYPE rotary switch selects the type of information which is displayed in the display mode.

P-R POSITION.

The R half of the Instruction register (instead of the L half) is connected to the instruction display. The instruction tag field is connected to the tag display. The data display is vacant.

REG POSITION.

The register column which is designated by the DISPLAY ADDRESS switch is connected to the data display.

SM-L POSITION.

The L half of the SM word which is designated by the DISPLAY ADDRESS switch is connected to the data display.

SM-R POSITION.

The R half of the SM word which is designated by the DISPLAY ADDRESS switch is connected to the data display. The tag field of the word is connected to the TAG display.

DISPLAY ADDRESS SWITCH.

The eight-position DISPLAY ADDRESS switch designates the particular register column or SM word which is displayed in the display

mode.  When the DISPLAY TYPE switch specifies SM-L or SM-R, the
DISPLAY ADDRESS switch positions 0-7 give the SM word address.  The
DISPLAY ADDRESS switch positions select register columns as follows
when the DISPLAY TYPE switch specifies REG.

Table 4-1

Register Display Selection

| Position | Registers displayed |
|----------|---------------------|
| 0 | None |
| 1 | AA, AC, AI |
| 2 | D, Y, X |
| 3 | MA2, MA1, MAO |
| 4 | CF, IA1, IAO |
| 5 | W-L (bytes 0, 1, 2) |
| 6 | W-R (bytes 3, 4, 5, 6) |
| 7 | None |

SECTION 5

ADAPTER CLUSTERS

GENERAL.

The Adapter Cluster provides interface between the Data Communications Processor and the Data Communications lines. Each adapter Cluster services a maximum of 16 lines operating simultaneously with speed ranges from 45.5 to 9600 bits per second.

The basic functions of the Adapter Cluster are:

a. Line termination which includes scanning, clocking, and temporary storage.

b. Character disassembly and assembly.

c. Synchronization; both attainment and maintenance.

d. Timer operation to maintain line discipline.

e. Some character recognition logic (mainly synchronization characters for the various line disciplines).

f. Provide the ability to exchange information with one DCP or two DCP's.

The Adapter Cluster block diagram is shown in figure 5-1.

CHARACTER CODES AND MESSAGE FORMATS.

The Adapter Cluster functions in a manner that makes it transparent to most character codes and all message formats. The Adapter Cluster recognizes only the SYN character in order to attain and retain synchronization when operating in the synchronous mode.

ADAPTER CLUSTER CONTROL OF THE DCP.

The Adapter Cluster is dependent upon the Data Communications Processor to provide control signals for all adapters operating within a cluster. Once an adapter operation is initiated by a DCP program, the adapter begins and continues to operate under control of the
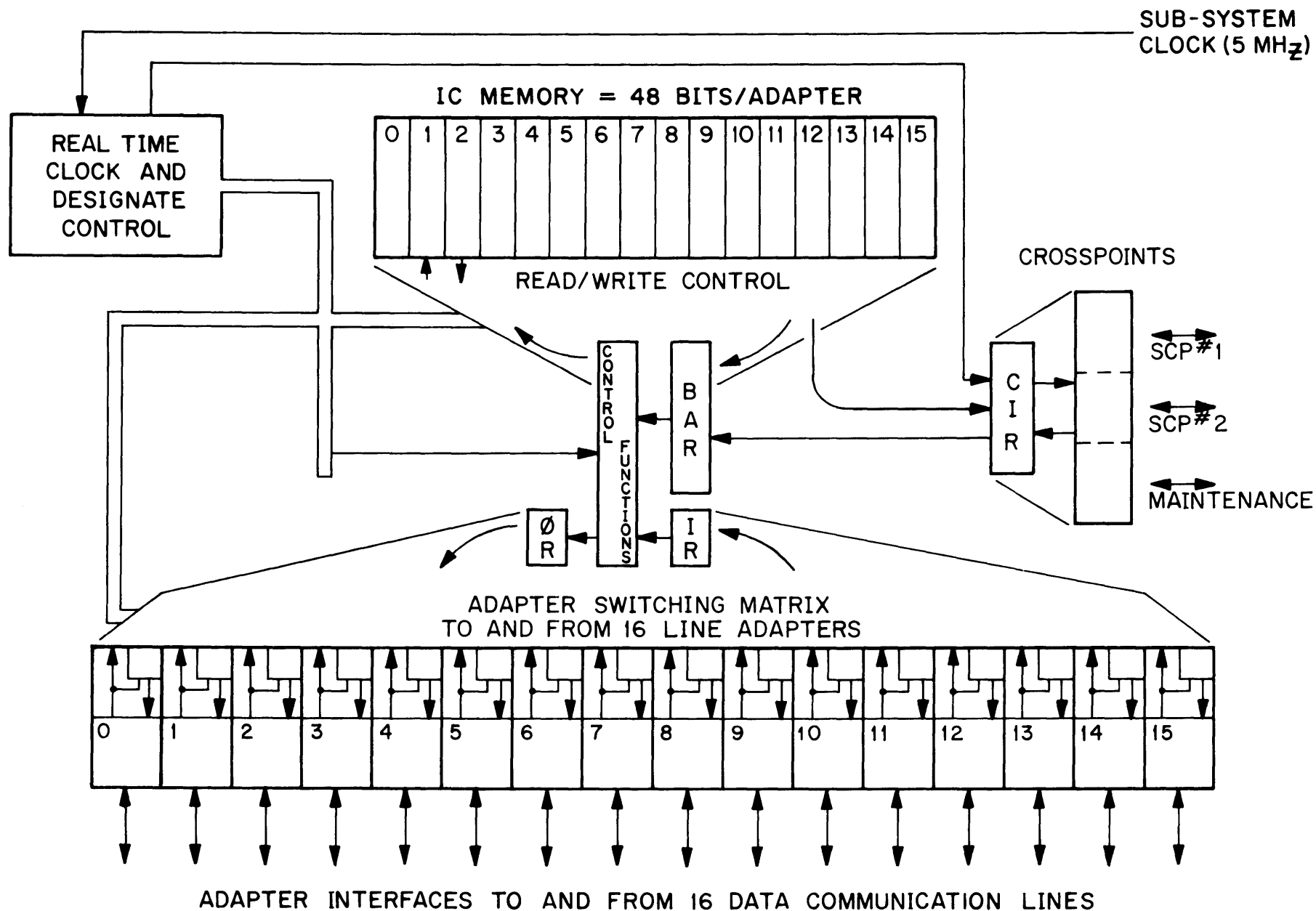
Figure 5-1.  Adapter Cluster Block Diagram

Adapter Cluster until additional control is required from the DCP, in which case an interrupt is sent to the DCP.

Some of the adapter control requirements are:

    a.   Ring Indicator was sensed.

    b.   Data transfer requirement.

    c.   Abandon Call and Retry.

    d.   Loss of Carrier occurred.

    e.   Buffer overflow.

    f.   Time out has occurred.

    g.   Break condition was sensed.

    h.   Data Set went Not Ready.

TEMPORARY STORAGE CAPABILITY OF THE ADAPTER CLUSTER.
The Adapter Cluster provides a two-character minimum storage for each adapter or data line that it services.

The Adapter Cluster also provides for temporary storage of control status information for each adapter.

The Adapter Cluster provides each of the 16 adapters with a 56-bit word for temporary storage of data and control status information.

ORGANIZATION OF THE ADAPTER CLUSTER.
The Adapter Cluster is subdivided into control sections. These sections are associated with individual data lines (adapters) or all data lines (adapters).

Sections associated with individual data lines (unique to one line) are:

    a.   Integrated circuit memory words.

    b.   Adapters.

Sections associated with all data lines (time-shared by all lines) are:

    a.   Cluster interface.

b.  Registers.

c.  Clock and Adapter Designate control.

d.  Bar-Field sensing and control logic.

e.  Read/Write control.

f.  Adapter Switching Matrix.

CLUSTER INTERFACE.

This interface is time-shared by all of the adapters in the Adapter
Cluster.  It interfaces both control and data information between
the Adapter Cluster and either the DCP or the Cluster Display
(figure 5-2).

These control signals are described as follows:

NOTE

Signals that apply to the DCP also
apply to the Cluster Display.

CONTROL SIGNALS FROM DCP TO ADAPTER CLUSTER.

DESIGNATE (DES).  When on, the Designate signal indicates that the
DCP is executing an Adapter Read, an Adapter Write, or an Adapter
Interrogate for the Adapter Cluster addressed by AA [7:4] of the
DCP.

NOTE

There is a separate DES signal
for each Adapter Cluster.

CLUSTER WRITE (CWR).  When on, the Cluster Write signal indicates
that the DCP is executing an Adapter Write or the write phase of the
Adapter Interrogate.  The Cluster Write signal is significant only
when the Designate signal is on also.

CLEAR (CLR).  When on, the Clear signal orders the Adapter Cluster
to clear all of its control flip-flops and go to an intialized idle
state.  The Clear signal is independent of the other signals and may
occur at any time.

CONTROL SIGNALS FROM THE ADAPTER CLUSTERS TO DCP.
CLUSTER ATTENTION NEEDED (CAN).  When on, the Cluster Attention
Needed signal indicates that an Adapter Cluster is requesting an
Adapter Read operation, which will cause the CAN to go off.

ACCESS GRANTED (ACG).  The Access Granted signal is on for one clock
period to signify that a designated Adapter Cluster is either
accepting output information from the DCP or is releasing informa-
tion to the DCP.

HOLD (HLD).  When on, the Hold signal causes the DCP to retain its
Adapter Write or Adapter Interrogate in anticipation of the ACG
signal coming on.  If neither the HLD or ACG signal comes on during
the execution of the above instructions, a fault interrupt occurs in
the DCP signifying "adapter not present".

INFORMATION SIGNALS.
The information signals are transmitted in either direction in a
half-duplex mode.  When a CWR is on, information travels from the
DCP to the designated Adapter Cluster.  When CWR is off, information
travels from the designated Adapter Cluster to the DCP.  There is a
one clock period turn-around time.

ADAPTER ADDRESS (AA0, AA1, AA2, AA3).  The Adapter Address signals
carry the adapter address between AA $[3:4]$ in the DCP and the Adap-
ter Cluster.

ADAPTER CONTROL (AC0, AC1, AC2, AC3, AC4).  The Adapter Control sig-
nals carry control codes specifying the Adapter Cluster operations
between AC $[4:5]$ in the DCP and the designated Adapter Cluster.

ADAPTER INFORMATION (AI0, AI1, AI2, AI3, AI4, AI5, AI6, AI7, AI8).
The Adapter Information signals carry data on control information
depending on the setting of Adapter Control between AI $[8:9]$ in
the DCP and the Adapter Cluster.  AI8 is used to carry the parity
bit if the information is data.

REGISTERS.

ACCESS CONFIRM REGISTER (AC).

This six-bit register contains a four-bit adapter address, an
occupied bit, and a program timer interrupt bit. This register
addresses the adapter for which the DCP or Cluster Display has just
serviced a CAN. The occupied bit is on until BAR clears the in-
terrupt bit in the related cluster memory word. The program timer
interrupt bit is used to differentiate the program timer interrupt
from all the other interrupts.

CLUSTER INTERFACE REGISTER (CIR).

The Cluster Interface register is connected to the DCP and the
Cluster Display via the Cluster Interface. The CIR register pro-
vides for information (control or data) transfer between the Cluster
Memory and the DCP or Cluster Display.

This 20-bit register is divided into four fields (figure 5-2)
which are:

CIR STATE (CS). This two-bit field is sensed and controlled by the
Adapter Cluster and Cluster Interface (table 5-1).

Table 5-1
CIR Current Status

| CS1 | CS0 | CIR State |
|-----|-----|-----------|
| 0 | 0 | Not occupied. |
| 0 | 1 | Occupied with information for the DCP or for the Cluster Display. |
| 1 | 0 | Occupied with information for the Adapter Cluster (BAR). |
| 1 | 1 | Master Clear occurred. |
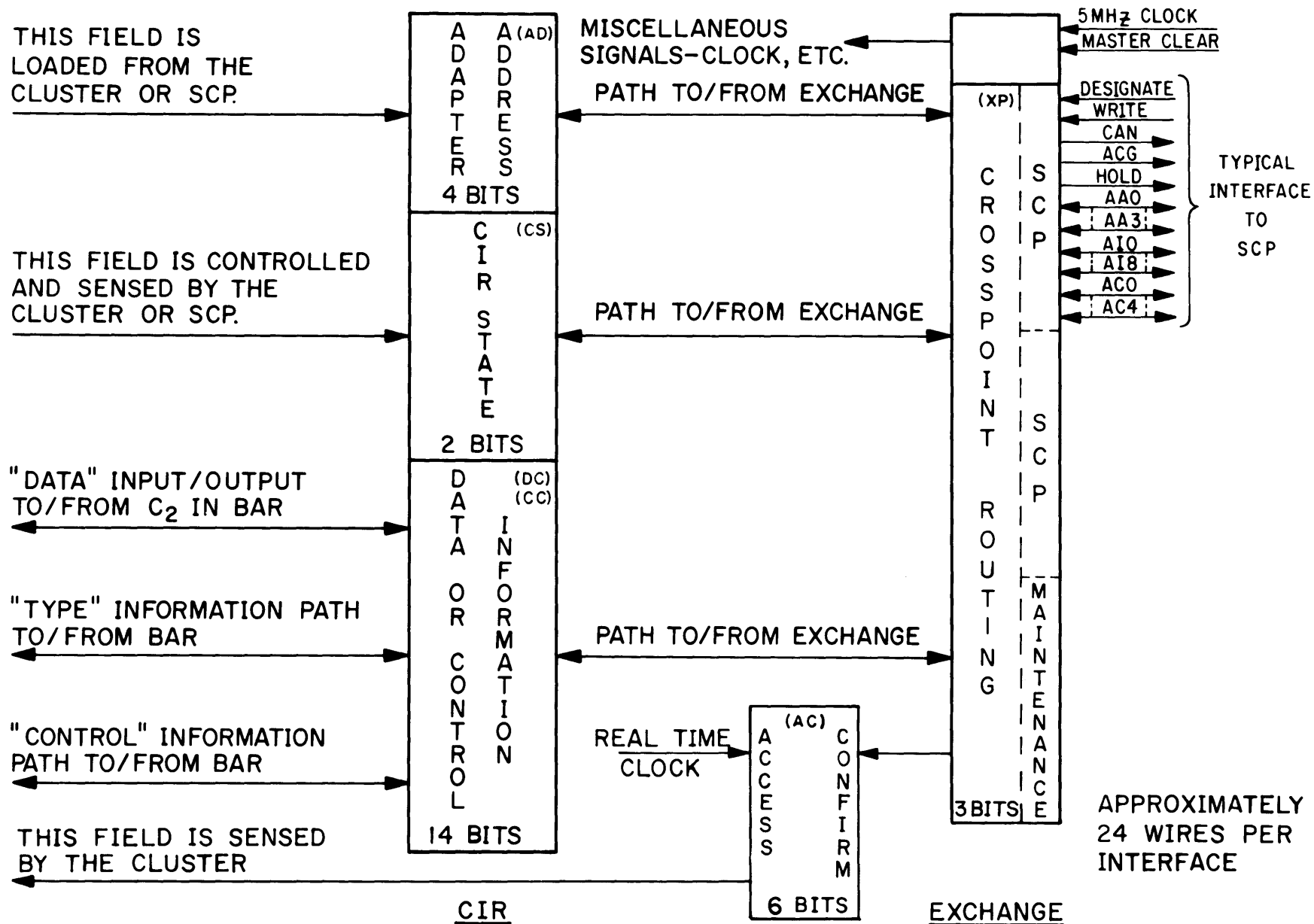
Figure 5-2. Adapter Cluster Interface

ADAPTER ADDRESS (AD).  This four-bit field receives information
from the Cluster Interface and the Adapter Designate Control section
of the Adapter Cluster.  The address of an adapter within the Adap-
ter Cluster is contained in this field.

NOTE

The Cluster Interface selects the information
for AD from AA [3:4] in the DCP or from the
Cluster Display.

CLUSTER CONTROL (CC).  This five-bit field receives information from
the Cluster Interface, Buffer Associative register, and the Cluster
Memory.  Multiple paths exist between this field, BAR and Cluster
Memory.  This field carries only control information.

NOTE

The Cluster Interface selects the
information for CC from AC in the
DCP or from the Cluster Display.

DATA OR CONTROL (DC).  This nine-bit field receives information from
the Cluster Interface and the Cluster Memory.  Multiple paths exist
between this field, BAR, and Cluster Memory.  This field carries
control or data information as determined by CC.

NOTE

The Cluster Interface selects the
information for DC from AI in the
DCP or from the Cluster Display.

CIR PRIORITY.
The lower-numbered adapter positions take priority over the higher
numbered adapters in CIR i.e., a CAN for adapter four overwrites a
CAN for adapter five in CIR.  Also, data has priority over control
information.  And, output to CIR has priority over a CAN from that
CIR.
CROSSPOINT REGISTER (XP).
This three-bit register uses a single bit to designate the unit

communicating with the Cluster Interface. There is one bit each
for DCP 1, DCP 2, or the maintenance display (Cluster Display).
Only one bit will be on at a time.

BUFFERS ASSOCIATIVE REGISTER (BAR).
This 56-bit register is the heart of the Adapter Cluster. All trans-
fer of control information and data between the adapters and the
Cluster Memory is through this register. This register is time-
shared by all adapters continuously. The contents of this register
are changing with every clock time as a result of sensing changes
on paths to CIR, the Adapter Switching Matrix, and Read/Write
control.

BAR is divided into eight fields as shown in figure 5-3. These
fields are:

CHARACTER ONE (C1). This 11-bit field can accept a bit or character
from the Adapter Switching Matrix or send one to it. Various paths
to Cluster Memory are necessary to implement the basic control of
this field. There is a path that shifts the entire field one bit
position. There is a path which shifts the C1 field content to the
Character Two field position within the buffer.

NOTE
This field includes start and
stop bits of the character
when applicable.

CHARACTER TWO (C2). This ten-bit field can accept a character from
the DC field in the CIR register or send one to it. This field
contains eight character bits, one parity bit, and an occupied bit.
Various paths are necessary to implement the basic control of this
field. There is a path which shifts the C2 field contents to the
Character One field position within the buffer.

BIT TIMER (BT). This seven-bit field is used for information
strobing within the Adapter Cluster. It is used for both synchro-
nous and asynchronous adapter operation. During asynchronous

BIT OUTPUT TO
ADAPTER SWITCHING MATRIX

CHAR I/O
WITH ADAPTER
SWITCHING MATRIX

BIT INPUT FROM
ADAPTER SWITCHING MATRIX

CHARACTER INPUT FROM CIR

CHARACTER OUTPUT TO CIR

I BIT SHIFT OF
CI WHEN STORED
INTO BUFFER

$C_1$
11 BITS

$C_2$
10 BITS

VARIOUS PATHS INTO
BUFFER FROM CI & C2
VIA READ/WRITE CONTROL

BIT $^{(BT)}$
TIMER
7 BITS

TYPE INFORMATION PATH
TO/FROM CIR

TYPE $^{(TY)}$
6 BITS

CONTROL SIGNALS TO CIR

STATE $^{(SC)}$
COUNTER $^{(SA)}$
5 BITS

CONTROL INFORMATION TO CIR

COMMAND $^{(BC)}_{(BI)}$
AND
INTERRUPT
7 BITS

CONTROL INPUT FROM CIR

PATHS TO/FROM BUFFER
VIA READ/WRITE CONTROL

CONTROL $^{(CT)}$
TIMER 2 BITS

CONTROL INPUT FROM CIR

PROGRAM $^{(PT)}$
TIMER 8 BITS

Figure 5-3.  Buffers Associative Register (BAR)

operation, this field is basically an extension of the clock counter of the Clock Generation section of the cluster. During synchronous adapter operation, this field senses the clock lines of the Data Sets through the adapter and the Adapter Switching Matrix of the cluster. In either case, this field provides control signals for the adapter and the C1 field.

TYPE (TY). This six-bit field is used for basic control purposes within the Adapter Cluster. It provides for Type information transfer to or from the CIR register DC field. The information within this field defines the type of adapter being serviced with each buffer access. The sixth bit, Force Image Path (FIP), is used to signal BAR that it should not perform any functions on data except to return the current data to the appropriate word in Cluster Memory unchanged.

The Type field designations are given in table 5-2. Tables 5-3, 5-4, 5-5, and 5-6 define the A, B, and C terms found in table 5-2.

Table 5-2

Type Field Designations

| TY4 | TY3 | TY2 | TY1 | TY0 | Adapter Type |
|-----|-----|-----|-----|-----|--------------|
| 0 | 0 | 0 | 0 | 0 | Unused |
| 0 | 0 | 0 | 0 | 1 | Unused |
| 0 | 0 | 0 | 1 | A0 | Touch-Tone |
| 0 | 0 | 1 | 0 | 0 | Reserved |
| 0 | 0 | 1 | 0 | 1 | Voice Response Single Light |
| 0 | 0 | 1 | 1 | 0 | Unused |
| 0 | 0 | 1 | 1 | 1 | Automatic Calling Unit |
| 0 | 1 | B2 | B1 | B0 | Synchronous |
| 1 | C3 | C2 | C1 | C0 | Asynchronous |

Table 5-3

Touch-Tone Attended Light Control

| AO | Description |
|----|-------------|
| 0 | Turns Data Set Attended light off. |
| 1 | Turns Data Set Attended light on. |

Table 5-4

Synchronous Specifications

| B2 | B1 | B0 | Character Size | Sync Pattern |
|----|----|----|----------------|--------------|
| 0 | 0 | D0 | 9 | 011010000 |
| 0 | 1 | D0 | 9 | 010011000 |
| 1 | 0 | D0 | 8 | 01101000 |
| 1 | 1 | D0 | 7 | 1010110 |

Table 5-5

Sync Editing

| DO | Receive Mode | Transmit Mode |
|----|--------------|---------------|
| 0 | Edit all sync characters. | Inhibit Sync Fill interrupt. |
| 1 | Inhibit sync character editing: | Enable Sync Fill interrupt. (See Buffer interrupt.) |
| | a. After first two sync characters. | |
| | b. After Initiate Receive mode. | |

## NOTE
Sync Fill occurs when necessary.  The
Sync Fill interrupt indicates that a
Sync Fill has occurred.

Table 5-6

Asynchronous Specifications

| C3 | C2 | C1 | CO | BPS | Character Size | Unit Size |
|----|----|----|----|-----|----------------|-----------|
| 0 | 0 | 0 | 0 | Unused | - | - |
| 0 | 0 | 0 | 1 | 45.5 | 5 | 7.5 |
| 0 | 0 | 1 | 0 | 56.9 | 5 | 7.5 |
| 0 | 0 | 1 | 1 | 75 | 5 | 7.5 |
| 0 | 1 | 0 | 0 | 110 | 8 | 11 |
| 0 | 1 | 0 | 1 | 134.5 | 7 | 9 |
| 0 | 1 | 1 | 0 | 150 | 8 | 10 |
| 0 | 1 | 1 | 1 | 300 | 8 | 10 |
| 1 | 0 | 0 | 0 | 600 | 8 | 10 |
| 1 | 0 | 0 | 1 | 1200 | 8 | 10 |
| 1 | 0 | 1 | 0 | 1200 | 4 | 6 |
| 1 | 0 | 1 | 1 | 1800 | 8 | 10 |
| 1 | 1 | 0 | 0 | 2400 | 8 | 10 |
| 1 | 1 | 0 | 1 | 3600 | 8 | 10 |
| 1 | 1 | 1 | 0 | 4800 | 8 | 10 |
| 1 | 1 | 1 | 1 | 9600 | 8 | 10 |

STATE COUNTER (SC) AND STATE COUNTER AUXILIARY (SA).  The two-bit
SC field is used for sequence controlling within the Adapter Clus-
ter.  Along with the BC field of BAR, SC is used to define the
existing state of an adapter as it is serviced with each buffer
access.  The three-bit SA field is used to buffer interrupt condi-
tions before they are encoded into the BI field of BAR.

BAR COMMAND (BC).  This three-bit field reflects and specifies the current command to the related adapter.  These commands are specified in table 5-7 and described in the paragraphs that follow the table.

<div align="center">

Table 5-7

BAR Commands

</div>

| BC [2:3] | Operation |
|----------|-----------|
| 0 0 0 | Out of service. |
| 0 0 1 | Look for CE (ring indicator). |
| 0 1 0 | Receive Ready. |
| 0 1 1 | Initiate Receive Ready. |
| 1 0 0 | Transmit Ready. |
| 1 0 1 | Initiate Transmit mode. |
| 1 1 0 | Finish Transmit. |
| 1 1 1 | Break. |

<u>Out Of Service</u>.
Adapter is not ready and all interrupts for that adapter are inhibited.

<u>Look For CE</u>.
The adapter will notify the data set to become ready to answer a call and interrupt the system when the call is answered.

<u>Receive Ready</u>.
The adapter is in a status to receive data.  This is not a software operation.  It is the result of a completed Initiate Receive mode.

<u>Initiate Receive Mode</u>.
This operation initializes the line to Receive then goes to Receive Ready, barring no complications.

Transmit Ready.

The adapter is in a status to transmit data. This is not a software operation. It is the result of a completed Initiate Transmit mode.

Initiate Transmit Mode.

This operation initializes the line to Transmit status and then goes to Transmit Ready, barring no complications.

Finish Transmit.

This operation causes the adapter to leave Transmit mode and enter Initiate Receive mode and continues to Receive mode as in Initiate Receive mode.

Break Transmit.

This operation transmits a Break condition on the related line.

BAR INTERRUPT (BI). This four-bit field, in conjunction with BC, indicates the status of the various interrupts generated by the Adapter Cluster. These interrupts are designated in table 5-8 and described in the paragraphs that follow the table.

NOTE

Normally these interrupts will appear
in the AI register when AC = 00010.
(See instruction codes.) However,
they can be interrogated to determine
the current status of an adapter.

For all of the interrupts with a BI of C or greater, there is a three-second wait (CID BUFF) which gives the program (DCP) time to respond to the condition before the interrupt is generated again.

Idle.

This indicates that the related adapter has no interrupts.

Idle Buffer.

This indicates that the related adapter has had an interrupt of the BI = C, D, E, or F within the last three seconds. If BC/BI is not

Table 5-8

BAR Interrupts

| BC:BT | Asynchronous | Synchronous | Automatic Calling Unit | Teletype | Voice Response |
|-------|--------------|-------------|------------------------|----------|----------------|
| 1:0 | IDLE | IDLE | IDLE | IDLE | IDLE |
| 1:1 | ID BUFF | ID BUFF | ID BUFF | ID BUFF | ID BUFF |
| 1:F | RING IND | RING IND | PROG ERROR | RING IND | PROG ERROR |
| 3:0 | IDLE | IDLE | IDLE | IDLE | IDLE |
| 3:1 | ID BUFF | ID BUFF | ID BUFF | ID BUFF | ID BUFF |
| 3:3 | DATA PRESENT | DATA PRESENT | | DATA PRESENT | |
| 3:6 | REC STATUS | REC STATUS | | REC STATUS | |
| 3:7 | ICTO | 3 SEC NO SYNC | | | |
| 3:8 | STOP BIT ERROR | | | | |
| 3:9 | BUFF OVFLW | BUFF OVFLW | | BUFF OVFLW | |
| 3:A | 30 SEC RNWY | 30 SEC RNWY | PROG ERROR | 30 SEC RNWY | PROG ERROR |
| 3:B | 3 SEC NT RDY | 3 SEC NT RDY | PROG ERROR | 3 SEC NT RDY | PROG ERROR |
| 3:C | BREAK | | | | |
| 3:D | DISCONNECT | | | | |
| 3:E | CF←OFF | CF←OFF | | | |
| 3:F | CC←OFF | CC←OFF | | CC←OFF | |
| 5:0 | IDLE | IDLE | IDLE | IDLE | IDLE |
| 5:1 | ID BUFF | ID BUFF | ID BUFF | ID BUFF | ID BUFF |
| 5:2 | BYTE REQ | BYTE REQ | BYTE REQ | BYTE REQ | BYTE REQ |
| 5:9 | | SYNC FILL | | VR ENABLE | BUFF EMPTY |
| 5:A | BYTE REQ BK UP | BYTE REQ BK UP | BYTE REQ BK UP | BYTE REQ BK UP | BYTE REQ BK UP |

Table 5-8 (cont)

BAR Interrupts

| BC:BT | Asynchronous | Synchronous | Automatic Calling Unit | Teletype | Voice Response |
|-------|--------------|-------------|------------------------|----------|----------------|
| 5:B | 3 SEC NT RDY | 3 SEC NT RDY | 3 SEC NT RDY | 3 SEC NT RDY | 3 SEC NT RDY |
| 5:C | BREAK | SB←ON | ACR←ON | | |
| 5:D | DISCONNECT | | COS+DLO ←OFF | | |
| 5:E | CB←OFF | CB←OFF | COS←ON | | |
| 5:F | CC←OFF | CC←OFF | PWI←OFF | CC←OFF | RDY←OFF |
| 7:0 | IDLE | IDLE | IDLE | IDLE | IDLE |
| 7:1 | ID BUFF | ID BUFF | ID BUFF | ID BUFF | ID BUFF |
| 7:F | BREAK CMPLT | BREAK CMPLT | PROG ERROR | PROG ERROR | PROG ERROR |

serviced within these three seconds since the original interrupt, it will occur again if the interrupt condition still exists.

Three Second Not Ready (Receive).
This indicates that the data set interface has gone into Receive mode or Finish Transmit.

Break.
This indicates line spacing (Break) has been received on the line.

Disconnect.
This indicates that the data set went "On Hook".

CF←Off.
This indicates that the Data Carrier Detector is no longer being received, i.e., carrier from remote data set is lost.

CC←Off.

This indicates that the data set is no longer in the data mode.

Byte Request Back Up.

This indicates that no byte has been received from the DCP for 30 seconds after the AC byte request.

Three Second Not Ready (Transmit).

This indicates that the data set has not gone into the Transmit mode within three seconds after the Initiate Transmit Mode.

CB←Off.

This indicates that the Clear to Send signal is no longer on, i.e., the data set is no longer prepared to send data.

Ring Indicator.

This indicates that the data set associated with the related adapter is ringing.

Data Present.

This indicates that there is a full data character present in C2 for the related adapter. This will normally be seen in the AC register as 00001.

Byte Request.

This indicates that C2 is empty and that another output character is required. This will normally be seen in the AC register as 01001.

Receive Status.

This indicates that the adapter has successfully entered Receive mode after a Finish Transmit request.

Input Character Time Out (ICTO).

This indicates that no new character has been received for three seconds since the last character was received.

Stop Bit Error.

This indicates that there was a Stop bit error, i.e., it did not arrive.

5-18

Buffer Overflow.

This indicates that C2 was filled, C1 was filled and more data is being received, i.e., data is lost.

30 Second Runaway.

This indicates that 30 seconds has passed since the beginning of the first character in Receive mode or the last 30 Second Runaway.

Break Complete.

Break (two spacing characters) has been sent out by the adapter.

Three Seconds No Sync.

This indicates that there have been no initial sync characters within three seconds.

Sync Fill.

This, if enabled, indicates that there has been sync fill in Transmit mode.

SB←On.

This indicates that Supervisory Received Data is on, i.e., the reverse channel is sensing data.

Program Error.

This indicates an undefined operation has been requested.

ACR←On.

This indicates that Abandon Call and Retry has come on, i.e., time out after sending digit because of no response.

COS + DLO←Off.

This indicates that Data Line Occupied went off and the related data set did not go into the data mode.

COS←On.

This indicates that the related data set has gone into the data mode.

PWI←Off.

This indicates that Power is off at the ACU.

<u>VR Enable</u>.

This indicates that the related VRU may be used to transmit.

<u>Buffer Empty</u>.

This indicates that a Finish Transmit request is complete.

<u>RDY←Off</u>.

This indicates the VRU is not ready.

CONTROL TIMER (CT). This two-bit field is used to generate timing for a particular line discipline if it is required, i.e., poll or select time out.

PROGRAM TIMER (PT). This eight-bit field is used to count a certain prespecified amount of time and give an interrupt after the time has elapsed. After the interrupt PT = 11111111. PT $[7:3]$ specifies the period of the clock as shown below. PT $[4:5]$ specifies the number of periods, PT $[7:3]$, required for the desired time delay. If PT $[4:5]$ is zero, the Program Timer interrupt will occur immediately. If PT $[4:5]$ is 31, the timer will be disabled. To minimize the error in the total time, use the period related to the range within which the desired time falls.

Table 5-9

PT Time Periods

| Range (to minimize error) Microseconds | Period (microseconds) | PT $[7:3]$ |
|---|---|---|
| .0064 - .192 | .0064 | 000 |
| .192 - 1.536 | .0512 | 001 |
| 1.536 - 12.228 | .4096 | 010 |
| 12.228 - 98.1 | 3.27 | 011 |
| 98.1 - 786. | 26.2 | 100 |
| 786. 6291. | 209.7 | 101 |
| 6291. - 50310. | 1677. | 110 |
| 50310. - 402630. | 13421. | 111 |

INPUT REGISTER (IR).
This ten-bit register carries the input status and information from the adapter currently being serviced (figure 5-1).

OUTPUT REGISTER (OR).
This six-bit register carries the output status and information to the currently serviced adapter (figure 5-1).

READ/WRITE TIMER REGISTER.
This 26-bit register is the basic five megaHertz clock for the Adapter Cluster and is divided into two fields (See figure 5-4).

SCAN COUNTER. These four bits provide three operations when addressing the adapters:

    a.  Scan each adapter to perform the operation requested
        by the BAR translation of the Cluster Memory word.

    b.  Transfer information from CIR to BAR when these four
        low-order bits in RT match AD in the CIR register.

    c.  Address a particular Cluster Memory word for the
        Read/Write control logic.

REAL TIME COUNTER. These 22 bits are used for timing the necessary clocks during the transmission and reception of asynchronous data for the various adapters via the BT field and for clocking the timer in the CT and PT fields.

MAINTENANCE REGISTER (MR).
To be specified.

CLUSTER MEMORY.
The Adapter Cluster Memory is a 56-bit, 16-word, integrated circuit memory. One word is assigned to each of the 16 adapters. The bit
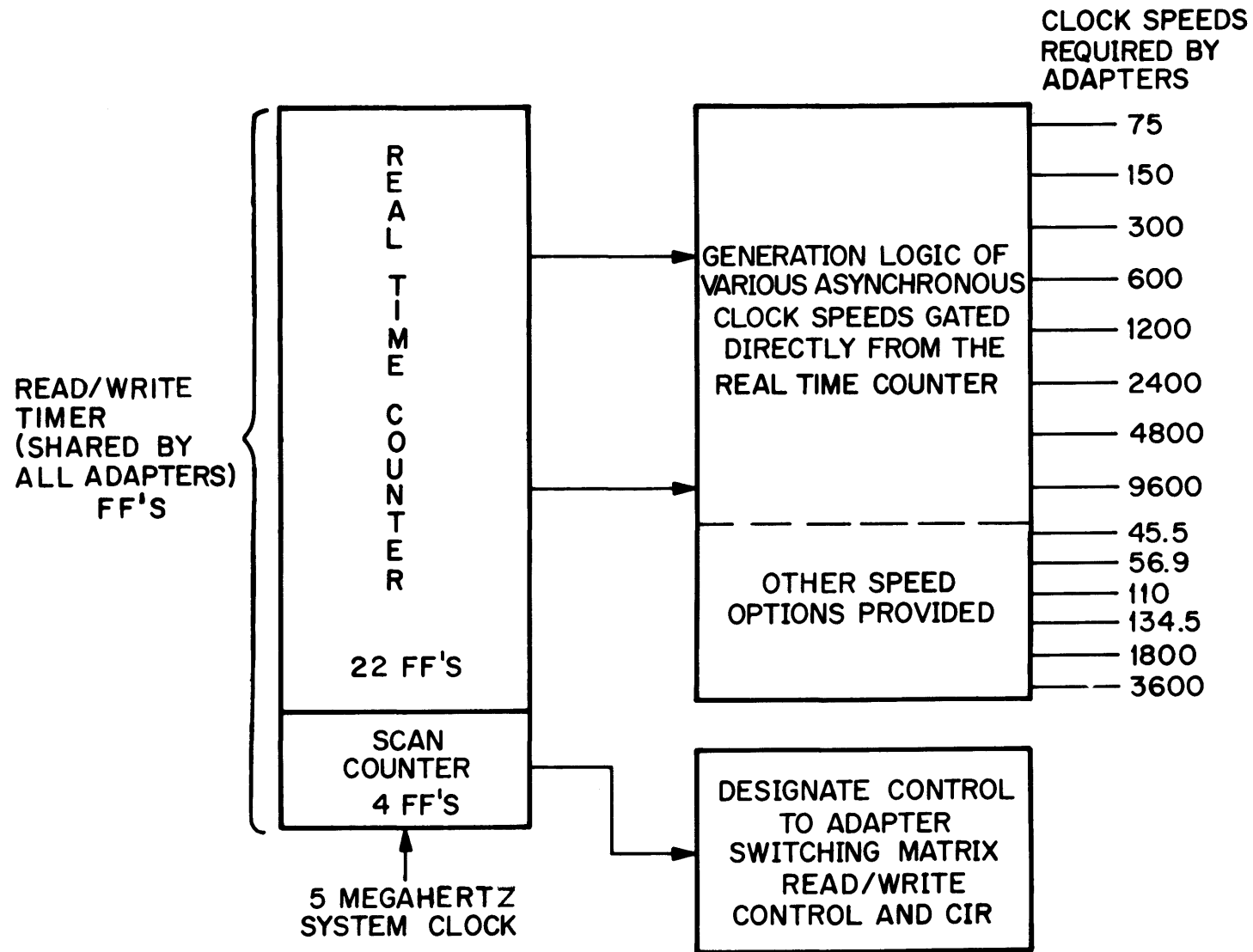
Figure 5-4. Clock Generator Within the Cluster for Asynchronous Operation

configuration within each word is identical to that which is speci-
fied for BAR.  BAR is the source for information stored in the
Cluster Memory and the destination for the information read out of
Cluster Memory.  Basic read and write times are 50 nanoseconds and
are done simultaneously on consecutive words where word zero follows
word 15 i.e., circular operation.

CONTROL FUNCTIONS.
This section contains the logic for driving most of the functions
of the Adapter Cluster.  It uses the contents of the different
fields of BAR to specify the operations required of the data in BAR
and IR.  And finally, the results of these operations are directed
back into the Cluster Memory or OR.

READ/WRITE CONTROL SECTION.
This section contains the control logic for simultaneous read and
write cycles of the Cluster Memory words.  The operation allows
an adapter word to be written into the Cluster Memory as another
adapter word is read from the Cluster Memory.

ADAPTER SWITCHING MATRIX SECTION.
This section contains designate control logic for the individual
adapters.  The designation generated within this section allows
the adapters to time-share common busses for IR and OR.

DATA AND INSTRUCTION FORMATS.
The data and instruction formats are defined in relation to the
Cluster Interface register (CIR) CC and DC fields.

DATA FORMATS.
The following data formats are shown in relation to the DC field
of the CIR register.  Unspecified bits of the DC field are always
equal to zero.  The least significant bit is found in bit position
zero in all cases.  See figures 5-5 through 5-11.

```
8  7  6  5  4  3  2  1  0     Bit designation.
```

| P | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|

[8:1] = Parity bit.
[7:8] = Data bits.

Figure 5-5.   EBCDIC Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.
```

| P |   | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|

[8:1] = Parity bit.
[6:7] = Data bits.

Figure 5-6.   USASCII Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.
```

| P |   |   | B | A | 8 | 4 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|

[8:1] = Parity bit.
[5:6] = Data bits.

Figure 5-7.   BCL Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.
```

|   |   |   | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|

[5:1] = Shift bit.
[4:5] = Data bits.

Figure 5-8.   BAUDOT Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.

┌──┬──┬──┬──┬──┬──┬──┬──┬──┐
│C │  │  │B │A │8 │4 │2 │1 │
└──┴──┴──┴──┴──┴──┴──┴──┴──┘
```

[8:1] = Parity bit.
[5:6] = Data bits.

Figure 5-9.  PTT CODE/6 Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.

┌──┬──┬──┬──┬──┬──┬──┬──┬──┐
│P │  │  │6 │5 │4 │3 │2 │1 │
└──┴──┴──┴──┴──┴──┴──┴──┴──┘
```

[8:1] = Parity bit.
[5:6] = Data bits.

Figure 5-10.  XS-3 Coded Data (Character)

```
8  7  6  5  4  3  2  1  0     Bit designation.

┌──┬──┬──┬──┬──┬──┬──┬──┬──┐
│  │  │  │  │  │8 │4 │2 │1 │
└──┴──┴──┴──┴──┴──┴──┴──┴──┘
```

[3:4] = Data bits.

Figure 5-11.  Burroughs Numeric Coded
              Data (Character)

INSTRUCTION CODES.

There are three groups of instruction codes that may be used on
the Adapter Cluster.  They are the Write, Interrogate, and Inter-
rupt instruction codes.  These are used to construct and monitor
the various fields in BAR and IR.

WRITE INSTRUCTIONS.

These instructions are used to fill or initialize the different
fields of BAR, IR, and MR.  Normally, only TY, C2, BC, and PT will

be used by the software. The other writes are provided for hard-
ware maintenance and testing. The codes and fields used to write
data are listed in table 5-10. CWR = 1 for these instructions.

Table 5-10
Write Codes and Formats

| Write | AI [8:9] | AC [4:5] |
|-------|----------|----------|
| TY | 000 FIP TY [4:5] | 00000 |
| C2 (Data) | C2 [8:9] | 0a001 |
| BC/BI | 00 BC [2:3] BI [3:4] | 00010 |
| SC/SA | 00 C110 C100 SC [1:2] SA [2:3] | 00011 |
| CT/BT | CT [1:2] BT [6:7] | 00100 |
| C1 | C1 [9:9] | 00101 |
| IR | IR [8:9] | 0b110 |
| MR | 00 MR [6:7] | 00111 |
| BC* | 00 BC [2:3] 0000 | 01010 |
| PT | 0 PT [7:8] | 01011 |

a = C2 occupied bit     b = IR9

* This write is the same as the BC/BI write except that
  the times in BAR are initialized, TY is left unchanged,
  BC is set as specified, and the other fields are zeroed.

INTERROGATE INSTRUCTIONS.
These instructions are used in conjunction with AWRR in the DCP to
interrogate the different fields of BAR, IR, and MR. Before the
interrogate, AC is specified as in table 5-11. This causes the
Adapter Cluster to read the desired field and transfer the infor-
mation to the AI register in the DCP for the responsive read por-
tion of the AWRR. CWR = 1 for the write portion of the interrogate;
CWR = 0 and CAN = 1 for the responsive read.

Table 5-11

Interrogate Codes and Formats

| Interrogate | AI [8:9] | AC [4:5] |
|---|---|---|
| TY | 000 FIP TY [4:5] | 10000 |
| C2 | C2 [8:9] | 1a001 |
| BC/BI | 00 BC [2:3] BI [3:4] | 10010 |
| SC/SA | 00 C110 C100 SC [1:2] SA [2:3] | 10011 |
| CT/BT | CT [1:2] BT [6:7] | 10100 |
| C1 | C1 [9:9] | 10101 |
| IR | IR [8:9] | 1b110 |
| MR | 00 MR [6:7] | 10111 |
| C2 clear** | C2 [8:9] | 11001 |
| BC/BI control* | 00 BC [2:2] 1 BI [3:4] | 11010 |
| OR | 000 OR [5:6] | 11111 |

a = C2 occupied (zero before interrogate)

b = IR9 (zero before interrogate)

* This interrogate is identical to the BC/BI interrogate except that BCO is set to one forcing a BC/BI interrupt condition for the software.

INTERRUPT INSTRUCTIONS.

These instructions are not programmatic and are the result of specific hardware conditions in the Adapter Cluster. When any of these interrupts occur, CWR = 0 and CAN = 1.

DATA PRESENT. This indicates that C2 has a complete character ready to be read in by the DCP. This requires a reading or clearing of C2.

BC/BI. This indicates that there was a BAR interrupt. BI will

have to be analyzed to determine the cause of the interrupt (see BAR interrupt).

Table 5-12
Interrupt Codes and Formats

| Interrupt | AI [8:9] | AC [4:5] |
|-----------|----------|----------|
| Data Present | C2 [8:9] | 00001 |
| BC/BI | 00 BC [2:2] 1 BI [3:4] | 00010 |
| Byte Request | 0 0 0 0 0 0 0 0 0 | 01001 |
| PT | 0 0 0 0 0 0 0 0 0 | 01011 |

BYTE REQUEST. This indicates that C2 is empty and that it has been shifted to C1 where it is being transmitted. Therefore, another character is required for transmission.

PT. This indicates that the desired total time period as originally specified in PT has elapsed (see program timer).

CHARACTER SYNCHRONIZATION.
The Adapter Cluster provides for both asynchronous and synchronous transmission of characters over the communication lines. Asynchronous transmission makes use of start-stop synchronization to identify the bits on the line. Synchronous transmission makes use of bit or character patterns to attain or retain synchronization on the line. The specified pattern (sync pattern) is dependent upon the line discipline being used on a line. The sync pattern precedes the transmission of a message and may be interspersed with the transmission of a message.

SERIALIZATION AND DESERIALIZATION OF DATA.
The Adapter Cluster provides for serial-by-bit or parallel-by-bit transmission of characters over communications lines. Within the

cluster, characters are transferred parallel-by-bit, adding or deleting bits as required for the various line disciplines.

CHARACTER FORMATS ON COMMUNICATION LINES.

The Adapter Cluster handles the following character formats on the communication lines as shown in figures 5-12 through 5-19.

SERIAL-BY-BIT CHARACTER FORMATS.

START-STOP SIX UNIT CHARACTER. Transmission speed is 1200 bits per second.



Figure 5-12. Six Unit Character

START-STOP 7.5 UNIT CHARACTER. Transmission speeds of 45.5, 56.9 and 75 bits per second.
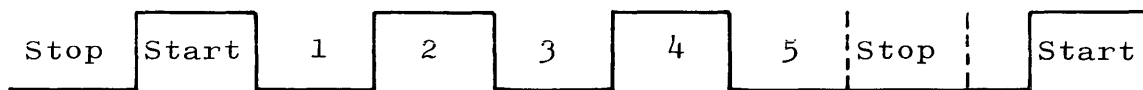


Figure 5-13. 7.5 Unit Character

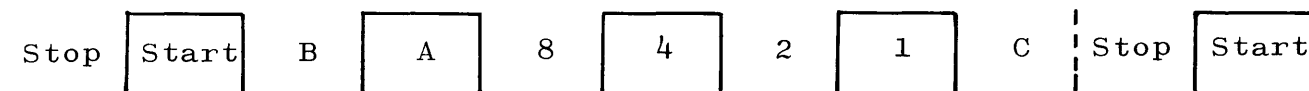START-STOP NINE UNIT CHARACTER. Transmission speed of 134.5 bits per second.



Figure 5-14. Nine Unit Character

START-STOP TEN UNIT CHARACTER. Transmission speeds of 150, 300, 600, 1200, 1800, 2400, 3600, 4800, and 9600 bits per second.
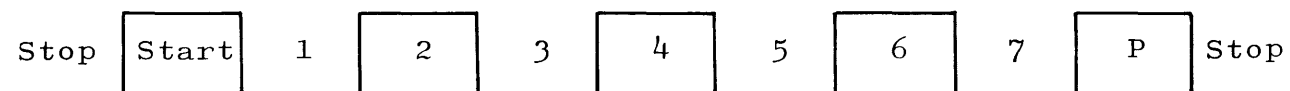
Figure 5-15.   Ten Unit Character

START-STOP 11 UNIT CHARACTER.   Transmission speeds of 110 bits per second.



Figure 5-16.   11 Unit Character

SYNC PATTERN NINE-BIT CHARACTER.   Transmission speeds of 2000, 2400, 4800, and 9600 bits per second.
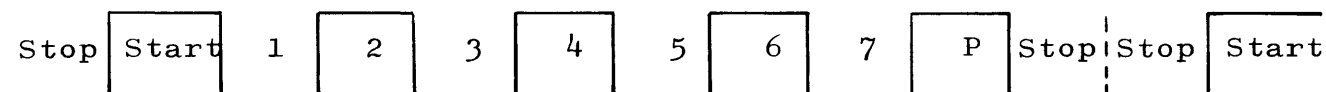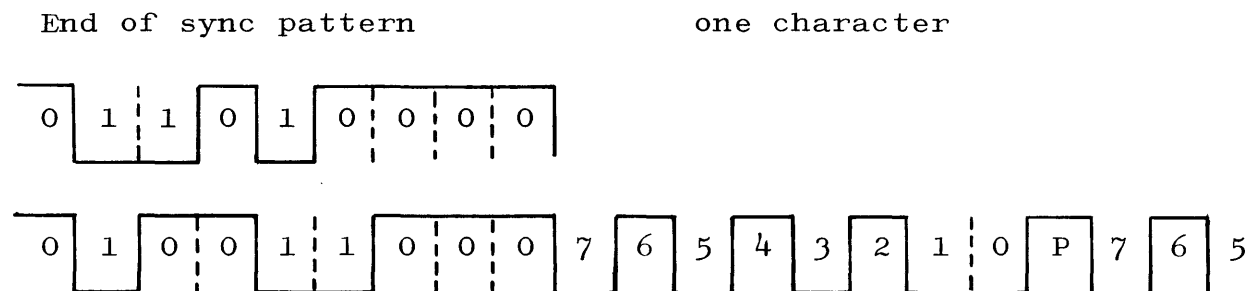


Figure 5-17.   Nine-Bit Character Sync Pattern

SYNC PATTERN EIGHT-BIT CHARACTERS.   Transmission speeds of 2000, 2400, 4800, and 9600 bits per second.
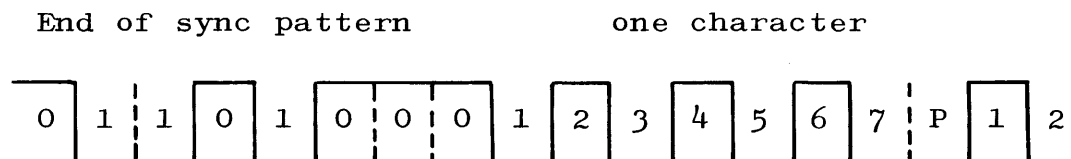


Figure 5-18.   Eight-Bit Character Sync Pattern

SYNC PATTERN SEVEN-BIT CHARACTER.   Transmission speeds of 2000, 2400, 4800, and 9600 bits per second.

```
 ___    ___    ___    ___  ___    ___    ___ ___
| 1 |  | 1 |  | 1 |1 | 1 ||   | 2 | 4 || 8 | A | B |P |   | 2 | 4 |
    | 0 |  | 0 |  |   |   | 0 | 1 |   |   |   |   |   | 1 |   |
```
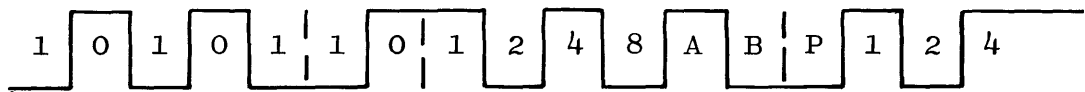
Figure 5-19.   Seven-Bit Character Sync Pattern

PARALLEL—BY—BIT CHARACTER FORMATS.

DIALING DIGIT BITS (OUTPUT).   Transmission speed is controlled by the Automatic Calling Unit.

        Number Bit 1    -----
        Number Bit 2    -----
        Number Bit 4    -----
        Number Bit 8    -----

TOUCH TONE CHARACTER 4 BITS (INPUT).   Transmission speed is a maximum of 14 input characters per second.

        -----    Bit 1
        -----    Bit 2
        -----    Bit 3
        -----    Bit 4

SECTION 6

CLUSTER DISPLAY PANEL AND MAINTENANCE AIDS

GENERAL.

The Cluster Display Panel is independent of the data communications system.  It may be temporarily connected to groups of up to 16 Adapter Clusters for maintenance purposes.  The Cluster Display has various switches and push-button indicators which are used to provide full control capabilities of the cluster interface in the same way a DCP controls the cluster interface (figure 6-1).

OPERATION.

This display may be used interactively while the Adapter Cluster is also being used by the data communications system (DCP).  The Cluster Display contends for the Adapter Cluster just as does the DCP causing no timing conflicts.

The Cluster Display provides the ability to set up the A register, as does the DCP, and performs the same reads, writes, or interrogates and then looks at the results in the A register again.

SWITCHES AND INDICATORS.

SWITCHES AA7F THROUGH AA4F.

These two-position switches designate the adapter cluster address (as does the AA [7:4] register in the DCP) selected by the Cluster Display Panel.

SWITCH-INDICATORS AA3F THROUGH AAOF.

These push-button switches designate the particular adapter (as does the AA [3:4] register in the DCP) selected by the Cluster Display Panel.

SWITCH-INDICATORS AC4F THROUGH ACOF.

These push-button indicators designate the specific operation (as in the five low-order bits of the AC register in the DCP) requested by the Cluster Display Panel.

SWITCH-INDICATORS AI8F THROUGH AIOF.

These push-button indicators designate the information required as specified by the AC field (as in the AI register in the DCP) for the Cluster Display Panel.

MASTER CLEAR SWITCH.

This switch clears all of the push-button indicators to zero.

BIT RESET SWITCH.

This push-button switch, when used in conjunction with a push-button indicator, clears it to zero.

ACTION SWITCH.

This three-position switch specifies the action to be taken by the Cluster Display Panel.

NEUTRAL.  No operations are initiated from the Cluster Display Panel.

RUN.  The designated operation recycles until the switch is returned to NEUTRAL.

STEP.  This spring-loaded position returns to NEUTRAL after permitting a single operation of the specified operation.

OPERATION SWITCH.

This five-position rotary switch provides the following functions:

INTERROGATE.  A Write operation occurs using the information in the Cluster Display Panel.  The Cluster Display Panel waits until a response is returned to the display push-button indicators.

WRITE.  A Write operation occurs using the information in the Cluster Display Panel without waiting for a response.

CAN.  Any Cluster Attention Needed signal (CAN) is serviced by the Cluster Display Panel when the switch is in this position.

EXCHANGE. This position allows monitoring the various control signals from the cluster interface by displaying them in the AI bits of the Cluster Display Panel.

LOCAL. No operations occur when the switch is in this position.

COUNT ENABLE SWITCH.
This two-position switch counts up AA3F-AAOF by one after each Cluster Display Panel operation when in the ON position. Otherwise, these bits remain unchanged.
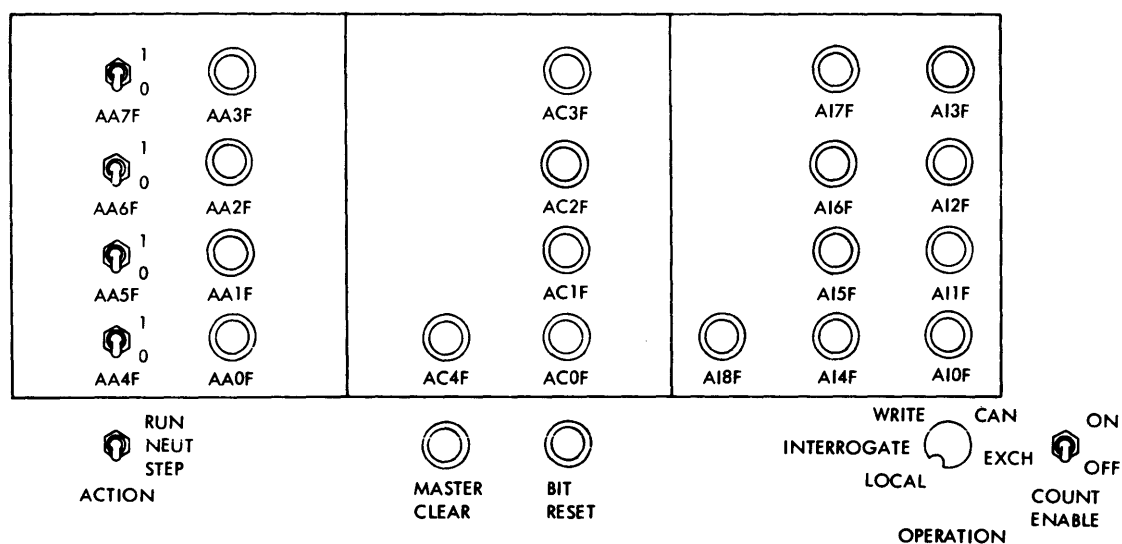


Figure 6-1. Cluster Display Panel

SECTION 7

ADAPTERS

GENERAL.

A maximum of 16 adapters can be housed in one Adapter Cluster. Each
adapter exchanges bits or characters between the Adapter Cluster
memory and the data lines. The temporary storage capability of the
adapter ranges from one bit to two characters depending upon the
type of adapter. An adapter is physically mounted on one card and
included with the adapter is an etched card connector which attaches
to a cable. The cable connects directly to the data set or to the
remote device in the case of direct connection. The various types
of adapters that are provided allow the DCP to interface with data
sets, the Voice Response system and by direct connection to remote
devices. The various adapters are defined below:

MODEM CONNECT ADAPTERS.

This adapter has the following characteristics:

    a.   Interfaces directly with the Adapter Switching Matrix
        of the Adapter Cluster.

    b.   Interfaces directly data sets using the RS 232 C
        defined interface.

    c.   Accommodates either synchronous or asynchronous data
        transmissions.

    d.   Operates in a serial-by-bit mode.

    e.   Information transfer rates accommodated by this adapter
        range from 75 to 9600 bits per second.

        1)  B 6650-1.  Up to 600 BPS.
        2)  B 6650-2.  Up to 1800 BPS.
        3)  B 6650-3.  Up to 2400 BPS.
        4)  B 6650-4.  Up to 4800 BPS.
        5)  B 6650-5.  Up to 9600 BPS.

f.  Transmission to and from this adapter is in the half-
    duplex mode.

g.  The control timer of the Adapter Cluster is used in
    conjunction with the operation of this adapter.

DIRECT CONNECT ADAPTERS.

This adapter has the following characteristics:

a.  Interfaces directly with the Adapter Switching Matrix of
    the Adapter Cluster.

b.  Interfaces directly with the remote terminal using either
    a two-wire current or voltage interface.

c.  Accommodates asynchronous data transmission.

d.  Operates in a serial-by-bit mode.

e.  Information transfer rates range from 45.5 to 1800
    bits per second.

    1)  B 6650-1.  Up to 600 BPS.
    2)  B 6650-2.  Up to 1800 BPS.

f.  Transmission to and from this adapter is in the half-
    duplex mode.

g.  The control timer of the Adapter Cluster is used in con-
    junction with the operation  of this adapter.

AUTOMATIC CALLING UNIT ADAPTER.

This adapter (B 6650-8) has the following characteristics:

a.  Interfaces directly with the Adapter Switching Matrix
    of the Adapter Cluster.

b.  Interfaces directly with the 801 Automatic Calling Unit
    (ACU) types A or C.

c. Operates in a synchronous mode.

d. Outputs one dialing digit (character) at a time.

e. The information transfer rate is controlled by the
Automatic Calling Unit (ACU).

f. Transmission from the adapter is always in a simplex mode.

g. The Automatic Calling Unit specified for use with this
adapter has a provision which allows the associated data
set to disconnect the line.

TOUCH-TONE ADAPTER.

This adapter (B 6650-6) has the following characteristics:

a. Interfaces directly with the Adapter Switching Matrix of
the Adapter Cluster.

b. Interfaces directly with the 403E3/403D5 data set using
an EIA voltage interface.

c. Operates in the synchronous mode on input.

d. Receives four-bit digits (characters) from the data set
in BCD code.

e. The 14 character-per-second maximum information transfer
rate is controlled by the remote station and the data set.
The remote station is a touch-tone telephone pad containing
12 buttons.

f. Transmission to the adapter is normally in a simplex mode.

g. Two leads couple a voice output through the adapter to
a telephone receiver.

h. The control timer of the Adapter Cluster is used in con-
junction with the input operation  of this adapter.

i.  Houses the driving elements that control the tone control leads of the data set.

AUDIO RESPONSE ADAPTER.

This adapter (B 6650-7) has the following characteristics:

a.  Interfaces the Adapter Cluster by a special interface.

b.  Interfaces the companion adapters (B 6650-6) by the two leads which couple the voice output to the 403 Data Set.

c.  Interfaces with the Voice Response Unit using a voltage interface.

d.  Operates in synchronization with the Voice Response Unit on output.

e.  The eight-bit address sent from an adapter to the Voice Response Unit has the following format:
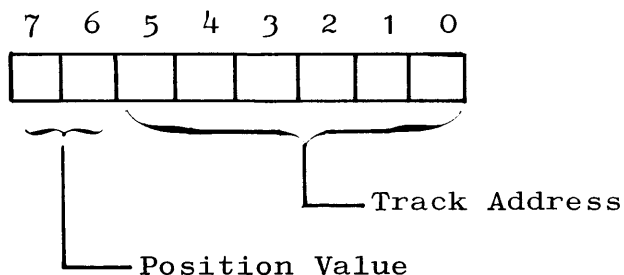


| Position Value | Description |
|---|---|
| 0 0 | Selects next position. |
| 0 1 | Selects position 1. |
| 1 0 | Selects position 2. |
| 1 1 | Selects position 3. |

Figure 7-1.  Voice Response Unit Address Format

f.  The Voice Response Unit initiates an address transfer
    cycle with a word time pulse occurring every .5 seconds.

g.  Address bits are sent from the adapter with the high-
    order bit first.

h.  The adapter generates shift pulses for the serial transfer
    of the address.

| EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII | EBCDIC | USASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | 00 NUL | 10 | 10 DLE | 20 | 1 | 30 | 1 | 40 | 20 SP | 50 | 26 & | 60 | 2D − | 70 | 1 |
| 01 | 01 SOH | 11 | 11 DC1 | 21 | 1 | 31 | 1 | 41 | 1 | 51 | 1 | 61 | 2F / | 71 | 1 |
| 02 | 02 STX | 12 | 12 DC2 | 22 | 1 | 32 | 16 SYN | 42 | 1 | 52 | 1 | 62 | 1 | 72 | 1 |
| 03 | 03 ETX | 13 | 13 DC3 | 23 | 1 | 33 | 1 | 43 | 1 | 53 | 1 | 63 | 1 | 73 | 1 |
| 04 | 1 | 14 | 1 | 24 | 1 | 34 | 1 | 44 | 1 | 54 | 1 | 64 | 1 | 74 | 1 |
| 05 | 09 HT | 15 | 1 | 25 | 0A LF | 35 | 1 | 45 | 1 | 55 | 1 | 65 | 1 | 75 | 1 |
| 06 | 1 | 16 | 08 BS | 26 | 17 ETB | 36 | 1 | 46 | 1 | 56 | 1 | 66 | 1 | 76 | 1 |
| 07 | 7F DEL | 17 | 1 | 27 | 1B ESC | 37 | 04 EOT | 47 | 1 | 57 | 1 | 67 | 1 | 77 | 1 |
| 08 | 1 | 18 | 18 CAN | 28 | 1 | 38 | 1 | 48 | 1 | 58 | 1 | 68 | 1 | 78 | 1 |
| 09 | 1 | 19 | 19 EM | 29 | 1 | 39 | 1 | 49 | 1 | 59 | 1 | 69 | 1 | 79 | 60 \ |
| 0A | 1 | 1A | 1 | 2A | 1 | 3A | 1 | 4A | 5B [ | 5A | 5D ] | 6A | 7C \| | 7A | 3A : |
| 0B | 0B VT | 1B | 1 | 2B | 1 | 3B | 1 | 4B | 2E . | 5B | 24 $ | 6B | 2C , | 7B | 23 # |
| 0C | 0C FF | 1C | 1C FS | 2C | 1 | 3C | 14 DC4 | 4C | 3C < | 5C | 2A * | 6C | 25 % | 7C | 40 @ |
| 0D | 0D CR | 1D | 1D GS | 2D | 05 ENQ | 3D | 15 NAK | 4D | 28 ( | 5D | 29 ) | 6D | 5F _ | 7D | 27 ' |
| 0E | 0E SO | 1E | 1E RS | 2E | 06 ACK | 3E | 1 | 4E | 2B + | 5E | 3B ; | 6E | 3E > | 7E | 3D = |
| 0F | 0F SI | 1F | 1F US | 2F | 07 BEL | 3F | 1A SUB | 4F | 21 ! | 5F | 5E ¬ | 6F | 3F ? | 7F | 22 " |
| 80 | 1 | 90 | 1 | A0 | 1 | B0 | 1 | C0 | 7B { | D0 | 7D } | E0 | 5C \ | F0 | 30 0 |
| 81 | 61 a | 91 | 6A j | A1 | 7E ~ | B1 | 1 | C1 | 41 A | D1 | 4A J | E1 | 1 | F1 | 31 1 |
| 82 | 62 b | 92 | 6B k | A2 | 73 s | B2 | 1 | C2 | 42 B | D2 | 4B K | E2 | 53 S | F2 | 32 2 |
| 83 | 63 c | 93 | 6C l | A3 | 74 t | B3 | 1 | C3 | 43 C | D3 | 4C L | E3 | 54 T | F3 | 33 3 |
| 84 | 64 d | 94 | 6D m | A4 | 75 u | B4 | 1 | C4 | 44 D | D4 | 4D M | E4 | 55 U | F4 | 34 4 |
| 85 | 65 e | 95 | 6E n | A5 | 76 v | B5 | 1 | C5 | 45 E | D5 | 4E N | E5 | 56 V | F5 | 35 5 |
| 86 | 66 f | 96 | 6F o | A6 | 77 w | B6 | 1 | C6 | 46 F | D6 | 4F O | E6 | 57 W | F6 | 36 6 |
| 87 | 67 g | 97 | 70 p | A7 | 78 x | B7 | 1 | C7 | 47 G | D7 | 50 P | E7 | 58 X | F7 | 37 7 |

1 EBCDIC input codes without correspondence in USASCII are translated to USASCII 00 NULL

| EBCDIC | USASCII | | EBCDIC | USASCII | | EBCDIC | USASCII | | EBCDIC | USASCII | EBCDIC | USASCII | | EBCDIC | USASCII | | EBCDIC | USASCII | | EBCDIC | USASCII | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 88 | 68 | h | 98 | 71 | q | A8 | 79 | y | B8 | 1 | C8 | 48 | H | D8 | 51 | Q | E8 | 59 | Y | F8 | 38 | 8 |
| 89 | 69 | i | 99 | 72 | r | A9 | 7A | z | B9 | 1 | C9 | 49 | I | D9 | 52 | R | E9 | 5A | Z | F9 | 39 | 9 |
| 8A | 1 | | 9A | 1 | | AA | 1 | | BA | 1 | CA | 1 | | DA | 1 | | EA | 1 | | FA | 1 | |
| 8B | 1 | | 9B | 1 | | AB | 1 | | BB | 1 | CB | 1 | | DB | 1 | | EB | 1 | | FB | 1 | |
| 8C | 1 | | 9C | 1 | | AC | 1 | | BC | 1 | CC | 1 | | DC | 1 | | EC | 1 | | FC | 1 | |
| 8D | 1 | | 9D | 1 | | AD | 1 | | BD | 1 | CD | 1 | | DD | 1 | | ED | 1 | | FD | 1 | |
| 8E | 1 | | 9E | 1 | | AE | 1 | | BE | 1 | CE | 1 | | DE | 1 | | EE | 1 | | FE | 1 | |
| 8F | 1 | | 9F | 1 | | AF | 1 | | BF | 1 | CF | 1 | | DF | 1 | | EF | 1 | | FF | 1 | |

1   EBCDIC input codes without correspondence in USASCII are translated to USASCII 00 NULL

APPENDIX A (cont)

Translation Chart EBCDIC to USASCII

| EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL | EBCDIC | BCL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | 1 | 10 | 1 | 20 | 1 | 30 | 1 | 40 | 10 SP | 50 | 30 & | 60 | 20 - | 70 | 1 |
| 01 | 1 | 11 | 1 | 21 | 1 | 31 | 1 | 41 | 1 | 51 | 1 | 61 | 11 / | 71 | 1 |
| 02 | 1 | 12 | 1 | 22 | 1 | 32 | 1 | 42 | 1 | 52 | 1 | 62 | 1 | 72 | 1 |
| 03 | 1 | 13 | 1 | 23 | 1 | 33 | 1 | 43 | 1 | 53 | 1 | 63 | 1 | 73 | 1 |
| 04 | 1 | 14 | 1 | 24 | 1 | 34 | 1 | 44 | 1 | 54 | 1 | 64 | 1 | 74 | 1 |
| 05 | 1 | 15 | 1 | 25 | 1 | 35 | 1 | 45 | 1 | 55 | 1 | 65 | 1 | 75 | 1 |
| 06 | 1 | 16 | 1 | 26 | 1 | 36 | 1 | 46 | 1 | 56 | 1 | 66 | 1 | 76 | 1 |
| 07 | 1 | 17 | 1 | 27 | 1 | 37 | 1 | 47 | 1 | 57 | 1 | 67 | 1 | 77 | 1 |
| 08 | 1 | 18 | 1 | 28 | 1 | 38 | 1 | 48 | 1 | 58 | 1 | 68 | 1 | 78 | 1 |
| 09 | 1 | 19 | 1 | 29 | 1 | 39 | 1 | 49 | 1 | 59 | 1 | 69 | 1 | 79 | 1 |
| 0A | 1 | 1A | 1 | 2A | 1 | 3A | 1 | 4A | 3C [ | 5A | 1E ] | 6A | 1 | 7A | 0D : |
| 0B | 1 | 1B | 1 | 2B | 1 | 3B | 1 | 4B | 3B . | 5B | 2B $ | 6B | 1B , | 7B | 0B # |
| 0C | 1 | 1C | 1 | 2C | 1 | 3C | 1 | 4C | 3E < | 5C | 2C * | 6C | 1C % | 7C | 0C @ |
| 0D | 1 | 1D | 1 | 2D | 1 | 3D | 1 | 4D | 3D ( | 5D | 2D ) | 6D | 1 | 7D | 1 |
| 0E | 1 | 1E | 1 | 2E | 1 | 3E | 1 | 4E | 3A + | 5E | 2E ; | 6E | 0E > | 7E | 1D = |
| 0F | 1 | 1F | 1 | 2F | 1 | 3F | 1 | 4F | 1 | 5F | 1 | 6F | 00 ? | 7F | 1F " |
| 80 | 1 | 90 | 1 | A0 | 1 | B0 | 1 | C0 | 3A + | D0 | 2A x | E0 | 1 | F0 | 0A 0 |
| 81 | 1 | 91 | 1 | A1 | 1 | B1 | 1 | C1 | 31 A | D1 | 21 J | E1 | 1 | F1 | 01 1 |
| 82 | 1 | 92 | 1 | A2 | 1 | B2 | 1 | C2 | 32 B | D2 | 22 K | E2 | 12 S | F2 | 02 2 |
| 83 | 1 | 93 | 1 | A3 | 1 | B3 | 1 | C3 | 33 C | D3 | 23 L | E3 | 13 T | F3 | 03 3 |
| 84 | 1 | 94 | 1 | A4 | 1 | B4 | 1 | C4 | 34 D | D4 | 24 M | E4 | 14 U | F4 | 04 4 |
| 85 | 1 | 95 | 1 | A5 | 1 | B5 | 1 | C5 | 35 E | D5 | 25 N | E5 | 15 V | F5 | 05 5 |
| 86 | 1 | 96 | 1 | A6 | 1 | B6 | 1 | C6 | 36 F | D6 | 26 O | E6 | 16 W | F6 | 06 6 |

1  EBCDIC input codes without correspondence in BCL are translated to BCL 00 ? (question mark)

| EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | | EBCDIC | BCL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 87 | 1 | | 97 | 1 | | A7 | 1 | | B7 | 1 | | C7 | 37 | G | D7 | 27 | P | E7 | 17 | X | F7 | 07 | 7 |
| 88 | 1 | | 98 | 1 | | A8 | 1 | | B8 | 1 | | C8 | 38 | H | D8 | 28 | Q | E8 | 18 | Y | F8 | 08 | 8 |
| 89 | 1 | | 99 | 1 | | A9 | 1 | | B9 | 1 | | C9 | 39 | I | D9 | 29 | R | E9 | 19 | Z | F9 | 09 | 9 |
| 8A | 1 | | 9A | 1 | | AA | 1 | | BA | 1 | | CA | 1 | | DA | 1 | | EA | 1 | | FA | 1 | |
| 8B | 1 | | 9B | 1 | | AB | 1 | | BB | 1 | | CB | 1 | | DB | 1 | | EB | 1 | | FB | 1 | |
| 8C | 1 | | 9C | 1 | | AC | 1 | | BC | 1 | | CC | 1 | | DC | 1 | | EC | 1 | | FC | 1 | |
| 8D | 1 | | 9D | 1 | | AD | 1 | | BD | 1 | | CD | 1 | | DD | 1 | | ED | 1 | | FD | 1 | |
| 8E | 1 | | 9E | 1 | | AE | 1 | | BE | 1 | | CE | 1 | | DE | 1 | | EE | 1 | | FE | 1 | |
| 8F | 1 | | 9F | 1 | | AF | 1 | | BF | 1 | | CF | 1 | | DF | 1 | | EF | 1 | | FF | 1 | |

1  EBCDIC input codes without correspondence in BCL are translated to BCL  00 ?

(question mark)

| USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | | USASCII | EBCDIC | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | 00 | NUL | 10 | 10 | DLE | 20 | 40 | SP | 30 | F0 | 0 | 40 | 7C | @ | 50 | D7 | P | 60 | 79 | \ | 70 | 97 | p |
| 01 | 01 | SOH | 11 | 11 | DC1 | 21 | 4F | ! | 31 | F1 | 1 | 41 | C1 | A | 51 | D8 | Q | 61 | 81 | a | 71 | 98 | q |
| 02 | 02 | STX | 12 | 12 | DC2 | 22 | 7F | " | 32 | F2 | 2 | 42 | C2 | B | 52 | D9 | R | 62 | 82 | b | 72 | 99 | r |
| 03 | 03 | ETX | 13 | 13 | DC3 | 23 | 7B | # | 33 | F3 | 3 | 43 | C3 | C | 53 | E2 | S | 63 | 83 | c | 73 | A2 | s |
| 04 | 37 | ECT | 14 | 3C | DC4 | 24 | 5B | $ | 34 | F4 | 4 | 44 | C4 | D | 54 | E3 | T | 64 | 84 | d | 74 | A3 | t |
| 05 | 2D | ENQ | 15 | 3D | NAK | 25 | 6C | % | 35 | F5 | 5 | 45 | C5 | E | 55 | E4 | U | 65 | 85 | e | 75 | A4 | u |
| 06 | 2E | ACK | 16 | 32 | SYN | 26 | 50 | & | 36 | F6 | 6 | 46 | C6 | F | 56 | E5 | V | 66 | 86 | f | 76 | A5 | v |
| 07 | 2F | BEL | 17 | 26 | ETB | 27 | 7D | ' | 37 | F7 | 7 | 47 | C7 | G | 57 | E6 | W | 67 | 87 | g | 77 | A6 | w |
| 08 | 16 | BS | 18 | 18 | CAN | 28 | 4D | ( | 38 | F8 | 8 | 48 | C8 | H | 58 | E7 | X | 68 | 88 | h | 78 | A7 | x |
| 09 | 05 | HT | 19 | 19 | EM | 29 | 5D | ) | 39 | F9 | 9 | 49 | C9 | I | 59 | E8 | Y | 69 | 89 | i | 79 | A8 | y |
| 0A | 25 | LF | 1A | 3F | SUB | 2A | 5C | * | 3A | 7A | : | 4A | D1 | J | 5A | E9 | Z | 6A | 91 | j | 7A | A9 | z |
| 0B | 0B | VT | 1B | 27 | ESC | 2B | 4E | + | 3B | 5E | ; | 4B | D2 | K | 5B | 4A | [ | 6B | 92 | k | 7B | C0 | { |
| 0C | 0C | FF | 1C | 1C | FS | 2C | 6B | , | 3C | 4C | < | 4C | D3 | L | 5C | E0 | \ | 6C | 93 | l | 7C | 6A | ¦ |
| 0D | 0D | CR | 1D | 1D | GS | 2D | 60 | - | 3D | 7E | = | 4D | D4 | M | 5D | 5A | ] | 6D | 94 | m | 7D | D0 | } |
| 0E | 0E | SO | 1E | 1E | RS | 2E | 4B | . | 3E | 6E | > | 4E | D5 | N | 5E | 5F | ∧ | 6E | 95 | n | 7E | A1 | ~ |
| 0F | 0F | SI | 1F | 1F | US | 2F | 61 | / | 3F | 6F | ? | 4F | D6 | O | 5F | 6D | _ | 6F | 96 | o | 7F | 07 | DEL |

NOTE:  All other input codes (80 through FF) are translated to EBCDIC 00 NULL.

| B C L | EBCDIC | | B C L | EBCDIC | | B C L | EBCDIC | | B C L | EBCDIC | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | 6F | ? | 10 | 40 | SP | 20 | 60 | - | 30 | 50 | & |
| 01 | F1 | 1 | 11 | 61 | / | 21 | D1 | J | 31 | C1 | A |
| 02 | F2 | 2 | 12 | E2 | S | 22 | D2 | K | 32 | C2 | B |
| 03 | F3 | 3 | 13 | E3 | T | 23 | D3 | L | 33 | C3 | C |
| 04 | F4 | 4 | 14 | E4 | U | 24 | D4 | M | 34 | C4 | D |
| 05 | F5 | 5 | 15 | E5 | V | 25 | D5 | N | 35 | C5 | E |
| 06 | F6 | 6 | 16 | E6 | W | 26 | D6 | O | 36 | C6 | F |
| 07 | F7 | 7 | 17 | E7 | X | 27 | D7 | P | 37 | C7 | G |
| 08 | F8 | 8 | 18 | E8 | Y | 28 | D8 | Q | 38 | C8 | H |
| 09 | F9 | 9 | 19 | E9 | Z | 29 | D9 | R | 39 | C9 | I |
| 0A | F0 | 0 | 1A | 6F① | ? | 2A | D0② | } | 3A | 4E | + |
| 0B | 7B | # | 1B | 6B | , | 2B | 5B | $ | 3B | 4B | . |
| 0C | 7C | @ | 1C | 6C | % | 2C | 5C | * | 3C | 4A | [ |
| 0D | 7A | : | 1D | 7E | = | 2D | 5D | ) | 3D | 4D | ( |
| 0E | 6E | > | 1E | 5A | ] | 2E | 5E | ; | 3E | 4C | < |
| 0F | 6F① | ? | 1F | 7F | " | 2F | 6F① | ? | 3F | 6F① | ? |

NOTE: All other input codes (40 through FF) are translated to EBCDIC 00 NUL

① BCL input codes without correspondence in EBCDIC are translated to EBCDIC 6F ? (question mark)

② Denotes "minus zero"

APPENDIX D
Translation Chart BCL to EBCDIC

# APPENDIX E
## Instruction List

| Group | Op Code | A Field | MNE | Operation |
|-------|---------|---------|-----|-----------|
| 0 | 00000 | 000 | IDLE | Idle |
|   | 00010 | OVV | AWI | Adapter Write |
|   | 00010 | 1VV | AWRR | Adapter Interrogate |
|   | 00001 | VVV | BKP | Stop/Branch on Breakpoint |
|   | 00011 | 100 | SHFT | Shift MA Registers Right |
|   | 00011 | 101 | HEYU | Set System Interrupt |
|   | 00011 | 000 | SCMR | Set Cluster Mask Register |
|   | 00011 | 001 | RCMR | Read Cluster Mask Register |
| 1 | 00100 | 000 | ARWN | Adapter Read When Needed |
|   | 00100 | 001 | ARIN | Adapter Read If Needed |
|   | 00100 | 010 | DBYZ | Decrement Branch Y Is Zero |
|   | 00100 | 011 | DBYN | Decrement Branch Y Is Non-Zero |
|   | 00100 | 100 | BRAN | Branch If System Attention Needed |
|   | 00101 | VVV | GOTO | Branch |
|   | 00110 | VVV | GOX | Branch Relative with X |
|   | 00111 | VVV | GOI | Branch Relative Indirect |
| 2 | 01000 | 000 | MOVE | Move Halfword |
|   | 01000 | 001 | LMRI | Local Memory Read Indirect |
|   | 01000 | 010 | SMRD | Scratchpad Memory Read Direct |
|   | 01000 | 011 | LMRD | Local Memory Read Direct |
|   | 01000 | 101 | LMWI | Local Memory Write Indirect |
|   | 01000 | 110 | SMWD | Scratchpad Memory Write Direct |
|   | 01000 | 111 | LMWD | Local Memory Write Direct |
|   | 01001 | 001 | MMR | Main Memory Read |
|   | 01001 | 100 | MMWU | Main Memory Write Unconditional |
|   | 01001 | 101 | MMWR | Main Memory Write Unconditional Retain Readout |
|   | 01001 | 110 | MMPW | Main Memory Protected Write |
|   | 01001 | 111 | MMPR | Main Memory Protected Write Retain Readout |

| Group | Op Code | A Field | MNE | Operation |
|-------|---------|---------|-----|-----------|
| 3 | 01100 | 000 | HAD | Halfword Add |
|  | 01101 | 000 | HADB | Halfword Add B Literal |
|  | 01100 | 100 | HSB | Halfword Subtract |
|  | 01101 | 100 | HSUB | Halfword Subtract B Literal |
|  | 01110 | VVV | TRAN | Translate |
|  | 01111 | 00V | PARY | Parity |
|  |  |  |  |  |
| 4 | 11101 | AAA | ADD | Add |
|  | 11110 | AAA | ADDB | Add B Literal |
|  | 11111 | AAA | ADDC | Add C Literal |
|  | 10000 | AAA | SUB | Subtract |
|  | 11000 | AAA | SUBB | Subtract B Literal |
|  | 11100 | AAA | SUBC | Subtract C Literal |
|  | 10101 | AAA | LAN | Logical AND |
|  | 10110 | AAA | LANB | Logical AND B Literal |
|  | 10111 | AAA | LANC | Logical AND C Literal |
|  | 10100 | AAA | LAOM | Logical AND-OR MA |
|  | 10001 | AAA | LOR | Logical OR |
|  | 10010 | AAA | LORB | Logical OR B Literal |
|  | 10011 | AAA | LORC | Logical OR C Literal |
|  | 11001 | AAA | LEO | Logical Exclusive OR |
|  | 11010 | AAA | LEOB | Logical Exclusive OR B Literal |
|  | 11011 | AAA | LEOC | Logical Exclusive OR C Literal |

# APPENDIX F

## Instruction Description Chart

IDLE      Idle until SCAN-OUT INITIALIZE

AWI      $A1=1 \Rightarrow AC \leftarrow B$; $AO=1 \Rightarrow AI \leftarrow C$; $CLIN \leftarrow AA.AC.AI$;

AWRR      $A1=1 \Rightarrow AC \leftarrow B$; $AO=1 \Rightarrow AI \leftarrow C$; $CLIN \leftarrow AA.AC.AI$; $AA.AC.AI \leftarrow CLIN$;

BKP      MATCH $\Rightarrow$ $IA \leftarrow B.C$ OR HALT ELSE NO-OP;

SHFT      $(MA[N:1] \leftarrow MA[N+1:1])*(B[7:2]=01 \Rightarrow Y[2:3]+1$ ELSE $B[2:3]+1)$;

HEYU      SYSTEM DATACOM INTERRUPT $\leftarrow$ TRUE;

MSKW      $CM \leftarrow AC.AI$;

MSKR      $AC.AI \leftarrow CM$;

 

ARWN      CAN $\Rightarrow$ $AA.AC.AI \leftarrow CLIN$; SAN $\Rightarrow$ $IA \leftarrow B.C$;

ARIN      CAN $\Rightarrow$ $AA.AC.AI \leftarrow CLIN$; $IA \leftarrow B.C$;

DBYZ      $Y=0 \Rightarrow IA \leftarrow B.C$ ELSE $Y \leftarrow Y-1$;

DBYN      $Y \neq 0 \Rightarrow IA \leftarrow B.C$, $Y \leftarrow Y-1$;

BRAN      SAN $\Rightarrow$ $IA \leftarrow B.C$;

GOTO      TRUE $\Rightarrow$ $IA \leftarrow B.C$;

GOX      TRUE $\Rightarrow$ $IA \leftarrow B.C+X$;

GOI      TRUE $\Rightarrow$ $IA \leftarrow B.C+(Y)$;

 

MOVE      $(C) \leftarrow MA$; $MA \leftarrow (B)$;

LMRD      $W \leftarrow LM(B.C)$;

LMRI      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $W \leftarrow LM(MA)$;

LMWD      $LM(B.C) \leftarrow W$;

LMWI      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $LM(MA) \leftarrow W$;

SMRD      $W \leftarrow SM(B)$ or $RG(B)$; $(C) \leftarrow MA$;

SMWD      $MA \leftarrow (B)$; $SM(C)$ or $RG(C) \leftarrow W$;

MMR      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $W \leftarrow MM(MA)$;

MMWU      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $MM(MA) \leftarrow W$; UNCONDITIONAL;

MMWR      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $MM(MA) \leftarrow W$; $W \leftarrow MM(MA)$ PREVIOUS CONTENTS;
             UNCONDITIONAL;

MMWP      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $MM(MA) \leftarrow W$; PROTECTED;

MWRP      $(C) \leftarrow MA$; $MA \leftarrow (B)$; $MM(MA) \leftarrow W$; $W \leftarrow MM(MA)$ PREVIOUS CONTENTS;
             PROTECTED;

| | |
|---|---|
| HAD | $(C) \leftarrow (B)+MA;$ |
| HADB | $(C) \leftarrow B+MA;$ |
| HSB | $(C) \leftarrow MA-(B);$ |
| HSBB | $(C) \leftarrow MA-B;$ |
| TRAN | $(C) \leftarrow TRAN(B);$ |
| PARY | $(C) \leftarrow PARY(B);$ |
| | |
| ADD | $(C) \leftarrow (A)+(B)+CFO;$ |
| ADDB | $(C) \leftarrow (A)+B+CFO;$ |
| ADDC | $(A) \leftarrow (B)+C+CFO;$ |
| SUB | $(C) \leftarrow (A)-(B)-CFO;$ |
| SUBB | $(C) \leftarrow (A)-B-CFO;$ |
| SUBC | $(A) \leftarrow C-(B)-CFO;$ |
| LAN | $(C) \leftarrow (A) \text{ AND } (B);$ |
| LANB | $(C) \leftarrow (A) \text{ AND } B;$ |
| LANC | $(A) \leftarrow (B) \text{ AND } C;$ |
| LAOM | $(A) \leftarrow [(B) \text{ AND } C] \text{ OR } MA(B);$ |
| LOR | $(C) \leftarrow (A) \text{ OR } (B);$ |
| LORB | $(C) \leftarrow (A) \text{ OR } B;$ |
| LORC | $(A) \leftarrow (B) \text{ OR } C;$ |
| LEO | $(C) \leftarrow (A) \text{ XOR } (B);$ |
| LEOB | $(C) \leftarrow (A) \text{ XOR } B;$ |
| LEOC | $(A) \leftarrow (B) \text{ XOR } C;$ |

# APPENDIX G
## Operand Selection Charts

| C7 B7 | C6 B6 | C5 B5 | C4 B4 | C3 B3 | C2 B2 | C1 B1 | C0 B0 |
|---|---|---|---|---|---|---|---|
| O | O | O | O | O | ZERO | ZERO | ZERO |
| O | O | O | O | 1 | AA | AC | AI |
| O | O | O | 1 | O | D | Y | X |
| O | O | O | 1 | 1 | MA-2 | MA-1 | MA-0 |
| O | O | 1 | O | O | CF | IA-1 | IA-0 |
| O | O | 1 | O | 1 | W-0 | W-1 | W-2 |
| O | O | 1 | 1 | O | W-3 | W-4 | W-5 |
| O | O | 1 | 1 | 1 | RSVD | RSVD | RSVD |
| O | 1 | IF B THEN B ← Y ELSE C ← D; | | | | | |
| 1 | O | SCRATCHPAD MEMORY WORD | | | HALF WORD | O | O |
| 1 | O | SCRATCHPAD MEMORY WORD | | | BYTE NMBR + ONE | | |
| 1 | 1 | 1 | 1 | 1 | BYTE NMBR + ONE (W) | | |

| A field | | | Instruction is not "literal in C field" | Instruction is "literal in C field" |
|---|---|---|---|---|
| A2 | A1 | A0 | Register selected as source | Register selected as destination |
| O | O | O | None (data = zeros) | None (no data stored) |
| O | O | 1 | AA | Indirect (The B field selects both the source and the destination) |
| O | 1 | O | X | X |
| O | 1 | 1 | Y | Y |
| 1 | O | O | D | D |
| 1 | O | 1 | MA0 | MA0 |
| 1 | 1 | O | MA1 | MA1 |
| 1 | 1 | 1 | MA2 | MA2 |

# APPENDIX H
## Instruction Matrix Chart

| OPCODE | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|
| 00000 | IDLE | | | | | | | |
| 00001 | BKP BREAKPOINT CODE - - - - - - - - - - - - - - - - - - - | | | | | | | |
| 00010 | AWI | AI←C | AC←B | AI←C AC←B | AWRR | AI←C | AC←B | AI←C AC←B |
| 00011 | MSKW | MSKR | | | SHFT | HEYU | | |
| 00100 | ARWN | ARIN | DBYZ | DBYN | BRAN | | | |
| 00101 | GOTO NOOP | CF1=0 CFO≐0 | CFO=1 | CF1=0 | CF1=1 | CFO=0 | CF1=1 CFO≠1 | UNCON |
| 00110 | GOX NOOP | CF1=0 CFO≐0 | CFO=1 | CF1=Q | CF1=1 | CFO=0 | CF1=1 CFO≠1 | UNCON |
| 00111 | GOI NOOP | CF1=0 CFO≐0 | CFO=1 | CF1=0 | CF1=1 | CFÒ=0 | CF1=1 CFO≠1 | UNCON |
| 01000 | MOVE | LMRI | SMRD | LMRD | | LMWI | SMWD | LMWD |
| 01001 | | MMR | | | MMWU | MMWR | MMWP | MWRP |
| 01010 | and 01011 are not used | | | | | | | |
| 01100 | HAD | | | | HSB | | | |
| 01101 | HADB | | | | HSBB | | | |
| 01110 | TRAN US←EB | BC←EB | | | | EB←US | EB←BC | |
| 01111 | PARY EVEN | ODD | | | | | | |
| 10000 | SUB A FIELD OPERAND - - - - - - - - - - - - - - - - - - - | | | | | | | |
| 10001 | LOR A FIELD OPERAND - - - - - - - - - - - - - - - - - - - | | | | | | | |
| 10010 | LORB A FIELD OPERAND - - - - - - - - - - - - - - - - - - - | | | | | | | |

## Instruction Matrix Chart

| OPCODE | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| 10011 | LORC<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 10100 | LAOM<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 10101 | LAN<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 10110 | LANB<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 10111 | LANC<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11000 | SUBB<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11001 | LEO<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11010 | LEOB<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11011 | LEOC<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11100 | SUBC<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11101 | ADD<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11110 | ADDB<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |
| 11111 | ADDC<br>A FIELD OPERAND - - - - - - - - - - - - - - - - - - - - - |

# APPENDIX I
## Carry Flip-Flop Chart

| | CFO=0 | CFO=1 | CF1=0 | CF1=1 |
|---|---|---|---|---|
| IDLE | N/A | N/A | N/A | N/A |
| AWI | N/A | N/A | N/A | N/A |
| AWRR | N/A | N/A | N/A | N/A |
| BKP | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| SHFT | CLEARED | CLEARED | CLEARED | CLEARED |
| HEYU | N/A | N/A | N/A | N/A |
| MSKW | N/A | N/A | N/A | N/A |
| MSKR | N/A | N/A | N/A | N/A |
| | | | | |
| ARWN | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| ARIN | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| DBYZ | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| DBYN | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| BRAN | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| GOTO | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| GOX | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| GOI | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED | BRANCH CLEARED |
| | | | | |
| MOVE | N/A | N/A | N/A | N/A |
| LMRD | N/A | N/A | N/A | N/A |
| LMRI | N/A | N/A | N/A | N/A |
| LMWD | N/A | N/A | N/A | N/A |
| LMWI | N/A | N/A | N/A | N/A |
| SMRD | N/A | N/A | N/A | N/A |
| SMWD | N/A | N/A | N/A | N/A |
| MMR | N/A | N/A | N/A | N/A |
| MMWU | N/A | N/A | N/A | N/A |
| MMWR | N/A | N/A | N/A | N/A |
| MMWP | N/A | N/A | N/A | N/A |
| MWRP | N/A | N/A | N/A | N/A |

## Carry Flip-Flop Chart

|        | CF0=0      | CF0=1   | CF1=0          | CF1=1        |
|--------|------------|---------|----------------|--------------|
| HAD    | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| HADB   | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| HSB    | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| HSBB   | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| TRAN   | N/A        | N/A     | N/A            | N/A          |
| PARY   | CLEARED    | CLEARED | PARITY ERROR   | GOOD PARITY  |
| ADD    | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| ADDB   | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| ADDC   | NO CARRY   | CARRY   | NONZERO RESULT | ZERO RESULT  |
| SUB    | NO BORROW  | BORROW  | NONZERO RESULT | ZERO RESULT  |
| SUBB   | NO BORROW  | BORROW  | NONZERO RESULT | ZERO RESULT  |
| SUBC   | NO BORROW  | BORROW  | NONZERO RESULT | ZERO RESULT  |
| LAN    | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LANB   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LANC   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LAOM   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LOR    | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LORB   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LORC   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LEO    | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LEOB   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |
| LEOC   | CLEARED    | CLEARED | NONZERO RESULT | ZERO RESULT  |

# INDEX

# BURROUGHS CORPORATION
## DATA PROCESSING PUBLICATIONS
### REMARKS FORM

TITLE: _____          FORM: _____
_____                 DATE: _____
_____

CHECK TYPE OF SUGGESTION:

☐ ADDITION        ☐ DELETION        ☐ REVISION        ☐ ERROR

GENERAL COMMENTS AND/OR SUGGESTIONS FOR IMPROVEMENT OF PUBLICATION:

FROM:   NAME      _____        DATE _____
        TITLE     _____
        COMPANY   _____
        ADDRESS   _____
                  _____

cut along dotted line

STAPLE

FOLD DOWN                    SECOND                    FOLD DOWN

Postage
Will Be Paid
by
Addressee

No
Postage Stamp
Necessary
If Mailed in the
United States
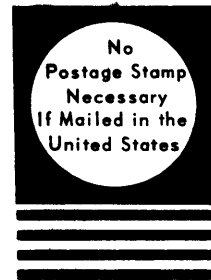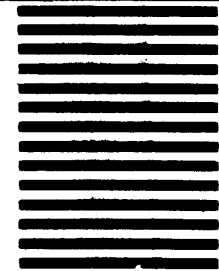
BUSINESS REPLY MAIL
First Class Permit No. 817, Detroit, Mich. 48232

Burroughs Corporation
6071 Second Avenue
Detroit, Michigan  48232

attn:  Sales Technical Services
        Systems Documentation

FOLD UP                        FIRST                      FOLD UP