IBM Field Engineering

Theory of Operation

IBM Confidential

2020 Processing Unit

System/360 Model 20

(Machines with serial no. 50,000 and above)

Z33-1021-0

Preface

This manual describes the IBM 2020 Central Processing Unit for the IBM System/360 Model 20 (Machines with serial no. 50,000 and above). It also defines the relationship of the 2020 Processing Unit to the System/360 Model 20 and to the input/output devices to which it may be connected.

Field Engineering Maintenance Diagrams manual, 2020 Processing Unit, System/360 Model 20 (Machines with serial no. 50,000 and above), Form Z33-1024-0 is complementary to this manual and is referenced in the text as "FEMDM".

First Edition

This manual contains preliminary information and is subject to change without further notice.

A form is provided at the back of this publication for reader's comments. If the form has been removed, comments may be addressed to IBM Corporation, 112 East Post Road, White Plains, N.Y. 10601, for the attention of Publishing Operations.

Illustrations

AC	Address Check
ALC	Automatic Length Count
ALU	Arithmetic Logic Unit
ASCII	American Standard Code for Information Interchange
BSCA	Binary Synchronous Communications Adapter
CC	Condition Code
CIO	Control Input/Output
CPL	Control Program Load
cpm	cards per minute
CPU	Central Processing Unit
CS	Cycle Steal
CY	Cycle
DA	Device Address
DR	Data Register
EBCDIC	Extended Binary-Coded-Decimal Interchange Code
E-phase	Execution-phase
FDR	From-Data Register
FL	Flip Latch
FS	Function Specification
Hz	Hertz
IAR	Instruction Address Register
ICPL	Initial Control Program Load
INH	Inhibit
INT	Internal, Integrator
I/O	Input/Output
IOC	Input/Output Channel
I-phase	Instruction Phase
LC	Length Count (latches)
LS	Local Store
LSA	Local Store Addressing Check, Line Sense Amplifier
MAR	Modify Address-Register
MANOP	(Circuit Controlled) Manual Operation
MFCM	Multi-Function Card Machine
NSI	Next Sequential Instruction
OCU	Operation Control Unit
Op code	Operation code
OP REG	Operation Register
PL	Program Level
PSW	Program Status Word
SA	Sense Amplifier
SAR	Storage Address Register
SDR	Storage Data Register
SIOC	Serial Input/Output Channel
SU	Shift Unit
TDR	To-Data Register
TIOB	Test Input/Output and Branch
XIO	Transfer Input/Output

CHAPTER 1 INTRODUCTION

IBM SYSTEM /360 MODEL 20

The IBM System /360 Model 20 Data Processing System (Figure 1-1) consists of the IBM 2020 Processing Unit, the IBM 1403 Printer or IBM 2203 Printer, the IBM 1442 Card Punch Model 5, the IBM 2501 Card Reader and either the IBM 2560 Multi-Function Card Machine (MFCM) or the IBM 2520 Card Read Punch. The system can be equipped with a Serial I/O Channel (SIOC) which accommodates I/O units, such as the IBM 1419 Magnetic Ink Character Reader or the IBM 1259 Magnetic Ink Character Reader, that use 1400 serial languages. The system may also be equipped with an I/O channel which serves I/O devices such as the IBM 2415 Magnetic Tape Unit and Control, or with a storage control feature to control the IBM 2311 Disk Storage Drive. A Binary Synchronous Communications Adapter (BSCA) allows transmitting and receiving of data from another source via telephone lines.

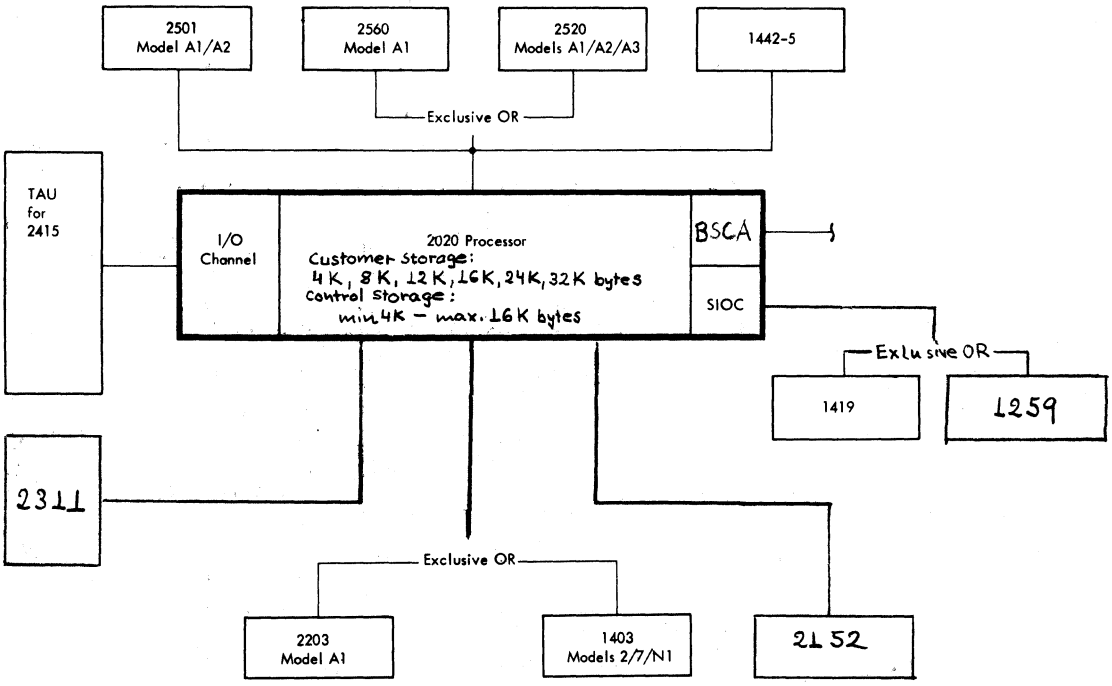


Figure 1-1. System/360 Model 20 Layout
(Maximum System Configuration)

The system configuration depends on the wishes of the customer, with four exceptions:

1. Either the 2560 Multi-Function Card Machine or the 2520 Card Read Punch or Card Punch can be used, but not both.
2. Either but not both of the two printers (IBM 1403 and IBM 2203) can be used.
3. Only one of the 2520 machines can be used on the system.
4. Either but not both of the two magnetic ink character readers (IBM 1419 or IBM 1259) can be used on the system.

The CPU housing contains the logic circuits of the CPU, as well as the logic circuits of the I/O attachments, channels and control features. The CPU has three console panels: one for the operator, another (which is normally covered) for the customer engineer and a third for operating the BSCA.

Data and machine instructions (customer) are entered into the system by means of a card reader (2501, 2560 or 2520), and placed into the customer s area of the main storage (core storage) as logical data.

The basic format of the logical data is a byte. One byte consists of a parity bit and eight data bits.

The addressing system allows selection of any byte or group of bytes within the available customer area of the main storage.

The Model 20 can perform 36 different machine operations (instructions). Everyone of the 36 machine instructions (Fig. 1-2) may be used to build the machine (customer's) program.

The machine instructions have different formats; that is, they are either two, four or six bytes long. The first byte contains the operation code (op code) which specifies the operation that is to be performed (add, subtract, compare, and so on). The following bytes contain the addresses of one or two operands or data itself (immediate data). The instructions which use operands that are greater than two bytes also contain a field length code that specifies the size of the operand.

NAME	MNEMONIC	FORMAT	OP CODE
BRANCH ON CONDITION	BCR	RR	07
BRANCH AND STORE	BASR	RR	0D
ADD	AR	RR	1A
SUBTRACT	SR	RR	1B
STORE HALFWORD	STH	RX	40
BRANCH ON CONDITION	BC	RX	47
LOAD HALFWORD	LH	RX	48
COMPARE HALFWORD	CH	RX	49
ADD HALFWORD	AH	RX	4A
SUBTRACT HALFWORD	SH	RX	4B
BRANCH AND STORE	BAS	RX	4D
SET PSW	SPSW	SI	81
TEST UNDER MASK	TM	SI	91
MOVE	MVI	SI	92
AND	NI	SI	94
COMPARE	CLI	SI	95
OR	OI	SI	96
HALT AND PROCEED	HPR	SI	99
TEST I/O AND BRANCH	TIOB	SI	9A
CONTROL I/O	CIO	SI	9B
TRANSFER I/O	XIO	SS	D0
MOVE NUMERICAL	MVN	SS	D1
MOVE CHARACTERS	MVC	SS	D2
MOVE ZONE	MVZ	SS	D3
COMPARE	CLC	SS	D5
TRANSLATE	TR	SS	DC
EDIT	ED	SS	DE
MOVE WITH OFFSET	MVO	SS	F1
PACK	PACK	SS	F2
UNPACK	UNPK	SS	F3
ZERO AND ADD	ZAP	SS	F8
COMPARE DECIMAL	CP	SS	F9
ADD DECIMAL	AP	SS	FA
SUBTRACT DECIMAL	SP	SS	FB
MULTIPLY DECIMAL	MP	SS	FC
DIVIDE DECIMAL	DP	SS	FD

Figure 1-2 Machine Instructions

Instructions are normally stored in consecutive main storage locations and are executed sequentially. However, the sequence of operations may be altered at any point in the program by conditional branch instructions. Conditional branch instructions make logical decisions by performing tests on indicators set by the machine as a result of other operations (such as a comparison of two operands or a forms skip after print).

The Model 20 uses the System /360 machine language that is able to process data of fixed length (fixed-point data) as well as a string of data the number of which is defined by a field length (variable-length data).

The CPU controls the functions of all I/O devices attached to it. However, I/O operations are initiated, halted, or tested by program instructions which select the desired unit and determine the operation it must perform (read, punch, and so on).

The CPU is available for processing during each I/O operation, even though several I/O devices may be functioning simultaneously. This overlap of I/O operations and CPU processing is called time sharing or read/write compute overlap. These modes are accomplished by allowing the CPU to continue with the machine program after data transmission to or from the specified I/O unit, while this unit is still completing the mechanical aspects of reading, punching , printing and so on.

In the Model 20, the machine program is automatically interrupted when all required data of an I/O operation are transferred. This interrupt causes a branch in the machine program. The branch address is defined by the programmer and may start a program routine which performs operations necessary when a data transmission to or from an I/O device is terminated. However, this interrupt can be disabled under program control.

The execution of the 36 machine instruction is the task of the Control Program (microprogram). The control program is a CPU internal sub-program, to which the customer has no access. The control program is stored in the control area of the main storage. The capacity of the control area depends upon the Storage capacity rented by the customer.

The control program consists of micro-instructions which are combined to execute the functions indicated by a machine instruction. The micro-instructions directly control the circuits in the CPU and in the attachments. 44 different micro-instructions are available.

INPUT/OUTPUT DEVICES

2501 Card Reader, Models A1, A2

The 2501 Card Reader provides punched-card input to the Model 20. Cards are read

at a maximum rate of 600 cpm in the Model A1 and up to 1,000 cpm in the Model A2.

Card reading is accomplished by solar cells and the validity of the data is verified as each column is read. Each card is also checked for off-register punching and for incorrect positioning in the read station.

The 2501 functions under control of the Model 20 program. Lights and switches provide the operator with information on operating conditions and with control of start, stop and card run-out.

2560 Multi-Function Card Machine

The 2560 Multi-Function Card Machine affords the Model 20 full card-file maintenance abilities plus two, four or six lines of card printing. The 2560 consists of two hoppers, a solar-cell read station, a common punching station, an optional printing station and five selective stackers.

Cards from both the primary and secondary hoppers can be read, punched and selected into any one of the five stackers, regardless of the hopper of origin. The unit record functions of reproducing, gang-punching, summary punching, collating, decollating and sorting can be performed on the MFCM under control of the Model 20 program. With the optional document printing feature, the MFCM functions include those of an interpreter and a punching, comparing, card document printer.

2520 Card Read Punch

The 2520 Card Read Punch provides punched-card input/output for the Model 20.

Cards are read, read and punched, or punched at the rate of 300 or 500 cards per minute (cpm) according to the model.

Model	Speed	Operation
A1	500 cpm	Read only, punch only, or read and punch simultaneously, as directed by the Model 20 program.
A2	500 cpm	Punch
A3	300 cpm	Punch

Card feeding is in parallel from the hopper, serially through the read station and in parallel through the punch station. Cards are ejected into radial stackers. The Model 20 program can select which of the two stackers receives each card. If no selection is made, the card goes into stacker 1.

The 2520 has lights and switches to provide the operator with information on operating conditions, and control of start, stop and card run-out. The 2520 is under control of the Model 20 program during all phases of reading, punching, stacker selection, and data checking.

1442 Card Punch, Model 5

The 1442 Card Punch Model 5 provides punched-card output for the Model 20. The 1442, Model 5 has a card hopper, a

serial punch station and a radial stacker. Card punching is done serially at a maximum rate of 160 columns per second.

Punching accuracy in the 1442, Model 5 is verified by comparing a signal, generated as each hole is punched, with data in the CPU core storage. Control of the 1442 is by the CPU stored program.

2203 Printer, Model A1

The 2203 Printer provides output for the Model 20 at up to 750 lines per minute. Interchangeable type bars allow the operator to select a type style and character set for a specific printing job.
The printing speed for any one application depends on the total number of lines printed, the amount of processing required for each printed line and the character set used.

Single, double and triple spacing of lines, plus skipping to a predetermined point, are performed by the tape-controlled carriage, directed by the CPU. The sequence and arrangement of data printed are also controlled by the stored program; a line to be printed is assembled in core storage in exactly the same sequence in which it is to appear as output. To ensure accuracy of output, each character is checked with the corresponding position in core storage before being printed.

The dual-feed carriage special feature permits independent and simultaneous control of two sets of forms.

1403 Printer, Models 2, 7, N1

The 1403 Printer provides output for the Model 20 at a rate of 600 lines per minute. The Model 2 has a capacity of 132 printing positions and the Model 7 can print 120 positions. The Model N1 prints 1,100 lines per minute on 132 positions.

Single, double and triple spacing of lines, plus skipping to a predetermined point, are performed by the tape-controlled carriage, under control of the CPU stored program. The Model 2 has a dual-speed carriage that permits high-speed skipping.

Each printing position can print 48 different characters and the printing format is controlled by the stored program of the system.

As each character is printed, checking circuits are set up to ensure that the character printed is correct. Checks are also made to ensure that only valid characters are printed and that overprinting does not occur. If an error is detected, the machine stops and the associated check light comes on.

2152 Printer - Keyboard

The 2152 Printer-Keyboard consists of a table mounted typewriter that is cable-

connected to its attachment in the CPU. The 2152 is used mainly as an inquiry station which allows the operator to retrieve information (e.g., from a disk file), and to print this informations. Inquiries via the Printer-Keyboard are especially advantageous when jabs are being run sequentially and there is a need to know the current working status. The Printer-Keyboard can also be used to change information or data in a just running program.

Beside the above mentioned functions, the 2152 may be used as a secondary, low speed printer, so that two separate reports may be produced by the same program.

The 2152 has a 125-character print line. The machine prints at 15.5 characters per second, producing one original print and a maximum of four copies. Printable are 88 characters of the standard System /360 layout.
Vertical spacing of three or six lines per inch can be manually selected by the operator. Selectable spacing of four or eight lines is optional.

2415 Magnetic Tape Units

The 2415 Magnetic Tape Unit is used as mass storage. The machine functions as an input and output device entering data into the system, and recording data generated by the system.

Six models of the 2415 Magnetic Tape Unit are available. In all models, the basic unit is a cabinet containing two tape drives, the tape control unit and the power supply. The additional tape drives required for the bigger models are housed in cabinets bolted to the basic unit.
All models are equipped with two-gap read/write heads which permit immediate read-back after writing, for error-detection purposes. The tape speed is 18.75 " (476,25 mm) per second in each case, and all models can read backwards.

The 2415 is attached to the CPU via the Input/Output Channel (IOC). The IOC regulates the flow of data to and from the system. Input and output are controlled by the stored program, which initiates operations by passing instructions to the channel. These commands are accepted by the tape control in the 2415 operating the tape drives.

2311 Disk Storage Drive

The models 11 or 12 can be attached to the system. A significant increase in data storage capacity for the System/360 Mod 20 is provided by the IBM 2311 Disk Storage Drive . The amount of information that can be stored is virtually unlimited since the disk pack holding the data is easily removed and stored on a shelf. Thus, an entire library can be kept off-line.
The 2311 is connected to the CPU via the Storage Control feature.

The Storage Control feature accommodates two 2311's of the same model (models 11 and 12 cannot be intermixed on the same system). Since both models use the same disk pack, the data format is the same, that is, a fixed length of 270 bytes.

Binary Synchronous Communications Adapter (BSCA)

The BSCA is a fully program-controlled communications adapter offering teleprocessing at accelerated speeds to the System /360 Model 20. The BSCA can communicate in point-to-point or in multipoint fashion. In the first case, data exchange is between only two stations at a time; in the second case, a master station consisting of a larger model of the System /360 can select or poll one of several Model 20 slave stations interconnected on a leased or private line.

The BSCA can transmit and receive all characters of the EBCDI or ASCII Code, whichever code is specified, since these are used directly as line codes. Eleven control characters provide for specific data link functions, which the program can use to control the data exchange in many ways. Thus, more operational flexibility is introduced, since almost all controlling power is given to the program.

1419 Magnetic Character Reader

The 1419 Magnetic Character Reader can be attached to a Model 20 through the serial input/output channel device. The 1419 reads into the system the magnetically inscribed information on checks and other banking documents, at speeds as high as 1,600 documents per minute. Documents can be sorted into as many as 13 classifications as they are read. The SIOC and, in turn, the 1419 operate under the control of the Model 20 program.

1259 Magnetic Character Reader

The 1259 is a medium-speed machine capable of reading and sorting magnetic-ink-inscribed banking documents of intermixed sizes and paper weights at speeds up to 600 documents per minute. The machine is connected to the Serial I/O Channel (SIOC) to operate under control of the machine program in the CPU, or it can operate off-line as a sorter under the control of its own circuitry and the operator's panel on the 1259.

PROGRAMMING

Data Formats

The basic data format is the byte (Fig. 1-3). The byte consists of eight data bits

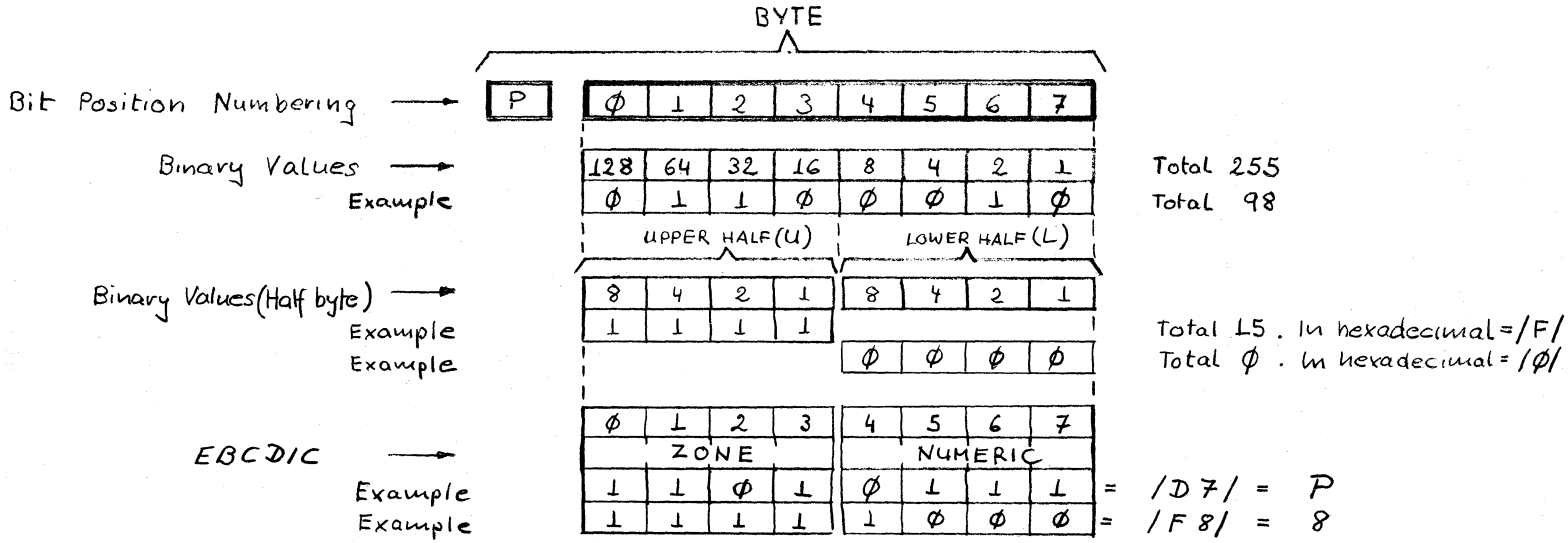


Figure 1-3 Byte Layout

and one parity bit. The parity bit is used for data checking purposes. In relation to the eight data bits the parity bit is turned on or off to provide always an odd number of turned on bits within a byte (Odd Parity). Thus, an even number of turned on bits is an indication that bit has been lost or turned on during the byte movement in the logical circuits.

The eight data bits provide 256 different bit combinations (0000 0000 to 1111 1111). The highest binary number expressed by a byte is 255 (all data bits on).

The eight data bits are coded in the Extended Binary-Coded-Decimal Interchange Code (EBCDIC) to represent numerical, alphabetical and special characters. 256 different bit combinations are possible (Figure 1-4).

For easier byte reading (e.g., display), the byte is divided into an upper and lower half (U and L). Each half is now able to represent the binary value 15 (all four bits on). Adding a one to this binary number results in a carry into the next higher four-bit (half) position. The number system in which no carry occurs until

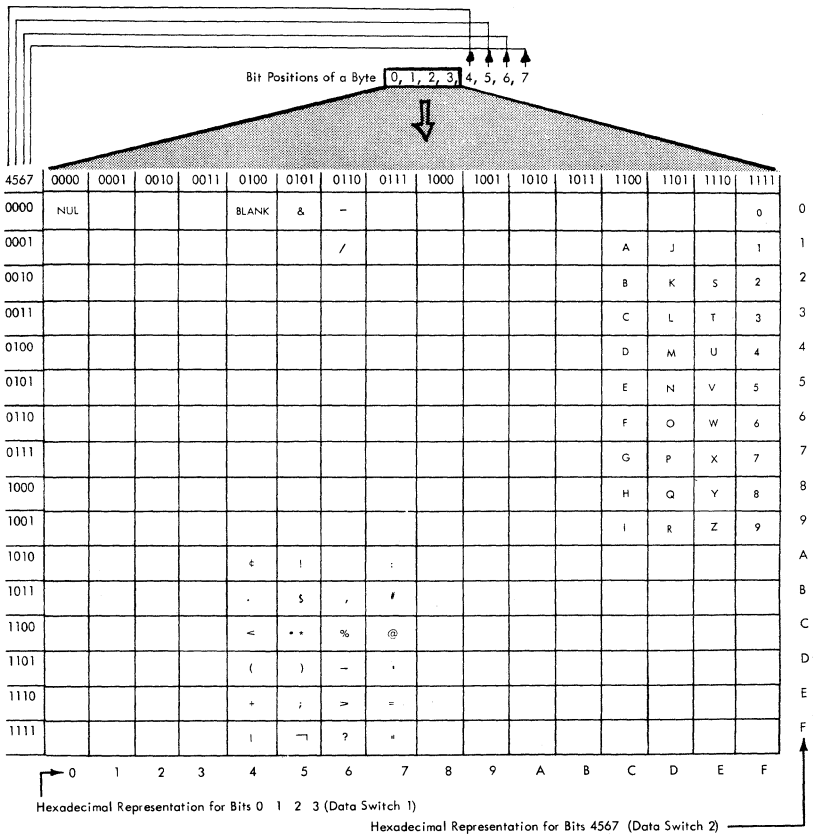


Figure 1-4 Extended Binary-Coded-Decimal Interchange Code (EBCDIC)

after 15 is called Hexadecimal. Because the available decimal system only provides the digits 0 to 9, an additional representation for the hexadecimal values corresponding to 10 through 15 is necessary. These values are represented by the alphabetical characters A (for 10) to F (for 15). To distinguish from the letters A to F, the hexadecimal values (Figure 1-5) are set in slashes (/A/ to /F/).

Fixed Point Data

Binary numbers have a fixed field length of two bytes. They are used in fixed point arithmetic because their point is assumed to be always to the right of the low-order bit (Figure 1-6).

Although this field contains 16 data bits (the parity bits are of no importance), only the 15 bits of the integer field represent the actual binary number. The high-order bit represents the sign. Negative numbers are expressed in the two's complement form.

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Figure 1-5 Hexadecimal Values

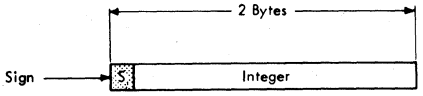


Figure 1-6 Binary Numbers

Thus, when the sign bit is zero, the integer is positive. When the sign bit is one, the integer is negative. In arithmetic operations, the sign bit is treated in the same way as any other bit. As every bit has a base value of two, the largest number that can be represented is $2^{15} = 32,000$.

Decimal Data

Decimal numbers have a variable field length from one to 16 bytes. They may appear in two different formats, the packed format or the zoned format.

In the zoned format, the byte contains only one decimal digit in the lower part of the byte while the upper part is filled with a zone (Figure 1-7).

The zone is either /F/ (15) in the EBCDIC or 5 in the American Standard Code for Information Interchange (ASCII) mode of operation. Either zone is automatically supplied by the CPU when a data field is changed from packed to zoned format. The zoned format is not used for arithmetic operations but it is needed for character-set sensitive I/O

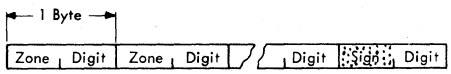


Figure 1-7 Zoned Format

devices such as printers. Negative decimal numbers are represented as true numbers. Only the sign indicates the difference. (See "Sign Standardization".)

Decimal digits range from zero to nine and are represented by four binary bits. Thus, a byte may contain two decimal digits; this is called the packed format (Figure 1-8). Decimal numbers carry the sign in the low-order position. The sign is represented by the bit configurations /A/ - /F/ (10-15).

Logical Data

Logical Data may have fixed as well as variable field length. Fixed length is one byte ; the variable length ranges from one to 256 bytes. Logical data is subject to such operations as translating, editing, comparing, bit testing and bit setting. Logical data consists of alphabetic or numerical character codes and is used for communication with character-set sensitive I/O devices. There is no sign (Figure 1-9).

Sign Standardization

In the System/360 language, all values from /A/ to /F/ are valid signs. During processing of decimal data, the sign is standardized by selecting either a plus or a minus sign from this group. The selection depends on the mode in which the CPU operates. The normal mode is EBCDIC but it can be changed to ASCII by the customer. The standardized sign is set into the sign position (four low-order bits) of the packed decimal data field (Figure 1-10).

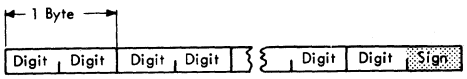


Figure 1-8. Packed Format

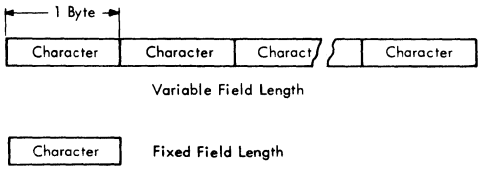


Figure 1-9. Logical Data

Machine Instruction Formats

- Machine Instructions are the commands that tell the CPU what to do with its data.
- Machine Instructions differ in length and they process different data formats.
- The four formats are: RR, RX, SI and SS.
- The RR instruction is two bytes long.
- The RX and the SI instructions are four bytes long.
- The SS instruction is six bytes long.

Valid Signs	EBCDIC	ASCII
/A/ = +		Standard +
/B/ = -		Standard -
/C/ = +	Standard +	
/D/ = -	Standard -	
/E/ = +		
/F/ = +		

Figure 1-10. Sign Standardization

Machine Instructions are the building blocks of the customer's program. Every machine instruction accomplishes a certain task because it tells the CPU what to do and which data to use. Data (for example, the amount of a bank account) is stored in the main storage in a number of consecutive bytes to form a field. Such a field may be one of the operands which a machine instruction acts upon. In most cases, another operand is required to which the first one is added (or subtracted, multiplied, and so on). To accomplish their tasks, the machine instructions carry the following information:

1. The operation code which specifies what is to be done.
2. The address of one or two operands.
3. One or two length codes when variable length data is processed.
4. Device Address (DA) and Function Specification (FS) when dealing with I/O devices.

Many operations produce an arithmetic result that usually replaces the first operand. They also produce a condition code that reflects the nature of the result (zero, less than zero or greater than zero).

RR Format

The RR abbreviation refers to register operations. These registers are areas in a part of the main storage which is called protected area. To distinguish between these and other registers, the registers in the protected area are named 'general registers'. There are eight general registers, each two bytes in length.

R1 is the address of the general register that contains operand 1. R2, similarly, specifies the general register in which operand 2 is stored (Figure 1-11).

In the branch-on-condition instruction, the R1 field is named M1. The M1 field bits represent, from left to right, the condition codes 0, 1, 2, 3. The M1 field (the mask) is compared with the condition code in the Program Status Word (PSW) and, if the comparison is equal, the branch is executed. If the mask is empty (no-1 bits) or the comparison is unequal, no branch is possible. The branch address is in the register specified by R2. The condition code is set to different values during operations, depending on the result (such as equal or unequal, plus or minus) of the operation. The program status word is employed as an internal control of the operation in the CPU and is located in auxiliary storage.

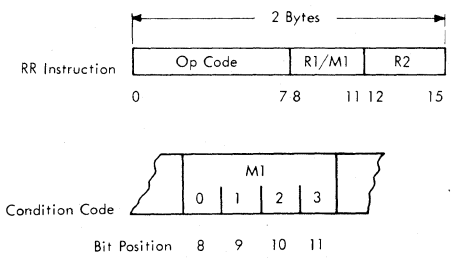


Figure 1-11 RR Format

Op Code in Hex	Name	Mnemonic	Type of Data Processed	Condition Code Influenced?
1A	Add	AR	Binary	yes
1B	Subtract	SR	Binary	yes
0D	Branch and Store	BASR	Binary	no
1D	Branch on Condition	BCR	- -	no

Figure 1-12 RR-Type Instructions

The RR-type instructions are listed in Figure 1-12.

RX Format

The RX abbreviation refers to operations from general register to main storage (or vice versa). Main storage is involved because one of the operands is located there. R1 is the address of a general register in which operand 1 is located (Figure 1-13). The bit field 12-15 is always zero in an RX instruction. B2 is the address of a general register in which some data (not operand 2) are stored. D2 is an address field.

The actual operand 2 address is obtained via two different methods. Normally, the entire B2/D2 field is used to address a main storage location. The other method is called indexing, in which the B2 address is used to read out the content of a general register. This content is then added to the D2 field and the result is the actual operand 2 address. The operand 2 address

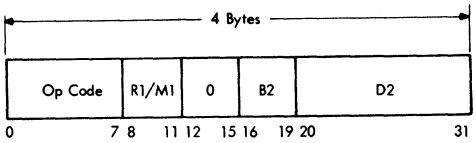


Figure 1-13. RX Instruction

Op Code in Hex	Name	Mnemonic	Type of Data Processed	Condition Code Influenced?
4A	Add Halfword	AH	Binary	yes
4B	Subtract Halfword	SH	Binary	yes
49	Compare Halfword	CH	Binary	yes
48	Load Halfword	LH	Binary	no
40	Store Halfword	STH	Binary	no
4D	Branch and Store	BAS	Binary	no
47	Branch on Condition	BC	Binary	no

Figure 1-14. RX-Type Instructions

is needed to fetch the operand 2 from the main storage. In the add and subtract operation, the result replaces operand 1.

The RX-type instructions are listed in Figure 1-14.

SI Format

The SI abbreviation refers to operations that involve data in the main storage and immediate data. Immediate data is an operand which is located in the instruction itself. The I2 field is the operand 2, the B1 field is the address of a general register, and the D1 field is an address field (Figure 1-15). The operand 1 address is obtained either by using all bits from 16 to 31 (B1/D1) directly as the operand 1 address or by indexing. In indexing, the content of the general register that is addressed by B1 is added to the D1 field, and the result is the operand 1 address. The SI-type instructions process logical (alphamerical) data.

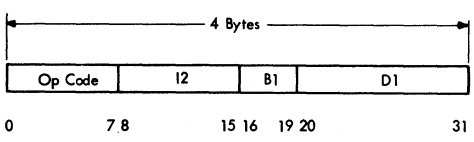


Figure 1-15. SI Instruction

Two of the three I/O instructions have the SI format. Their internal structure differs somewhat from that of the other SI instructions because they must be able to perform various operations. The Test I/O and Branch (TIOB) instruction (Figure 1-16) causes a branch in the machine program if a certain condition exists in an I/O device. The DA field is the device address that designates the I/O device in which a certain condition is to be tested. The FS field is the function specification that defines exactly what is to be tested in this particular I/O unit. The B1/D1 field is the address to which the machine program branches when the tested condition is found to be 'on'. This makes branching on an error condition possible. Because B1 is also the address of a general register, the branch address may be altered via indexing.

Another I/O instruction in the SI format is the Control I/O (CIO) instruction (Figure 1-17). The device address specifies in which I/O device the control operation is to be performed and the function specification defines the particular component of this machine (for example, carriage of the printer) in which something is to be controlled. The B1/D1 field is a detailed function specification that defines exactly what is to be done (for example, an immediate space). The detailed function specification

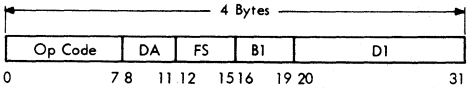


Figure 1-16. Test I/O and Branch Instruction

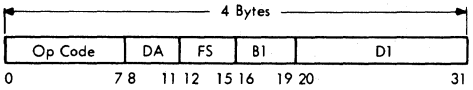


Figure 1-17. Control I/O Instruction

may be altered via indexing. Figure 1-18 shows the SI-type instructions.

SS Format

The SS abbreviation refers to storage-to-storage operations. The main storage is involved because both operands are data fields which are located in the main storage. The operands that contain decimal data may have a size of one to 16 bytes each. The address of the leftmost byte of operand 1 can be taken directly from the B1/D1 field. It can also be computed via indexing. In indexing, the B1 part is used to address a general register and the content of this register is added to the D1 part to generate the operand 1 address. The same is true of the B2/D2 field which is the address of

Op Code in Hex	Name	Mnemonic	Type of Data Processed	Condition Code Influenced?
92	Move	MVI	Logical	no
95	Compare Logical	CLI	Logical	yes
94	AND	NI	Logical	yes
96	OR	OI	Logical	yes
91	Test under Mask	TM	Logical	yes
99	Halt and Proceed	HPR	Logical	no
81	Set PSW	SPSW	Logical	yes
9A	Test I/O and Branch	TIOB	- -	no
9B	Control I/O	CIO	- -	yes
83	Diagnose	- -	- -	- -

Figure 1-18. SI-Type Instructions

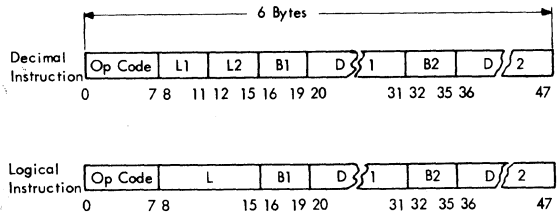


Figure 1-19. SS Format

the leftmost byte of operand 2 (Figure 1-19).

The length of operand 1 is specified by the L1 field in the instruction. The length of operand 2 is given in the L2 field. The L1/L2 fields have four bits each and the maximum number that can be represented is 15. The number in these fields is actually a field length code because it specifies the number of bytes that extend to the left of the units byte of an operand. Thus, the true length of every operand is always one byte longer than indicated by the field length code.

This field length specification is used to obtain a carry from a length counter when the units byte is reached. Also, it is possible to obtain the units byte address by simply adding the length code to the address of the leftmost byte of an operand.

The SS format instructions that handle logical data have only one length of code field, eight bits in size. This length code refers to operand 1 only.

The length code may specify operand 1 sizes from 0 to 255 bytes to the left of the units byte, equaling operand sizes from 1 to 256 bytes.

Among the SS instructions is the third I/O instruction. This instruction has a somewhat different internal structure as shown in Figure 1-20. The Transfer I/O (XIO) instruction causes a data transfer from an I/O device to the main storage, or vice versa. The DA field specifies the particular I/O device (printer, card reader, and so on) and the FS field specifies the particular function in the addressed device (for example, punch secondary). The B1/D1 field is the address of the leftmost byte of the input or output field in the main storage. This address may be altered via indexing. The B2/D2 field, which may also be indexed, is true field length. Although this field length contains 16 bits, the field length is limited to 4,095 bytes for other reasons. Also , the field

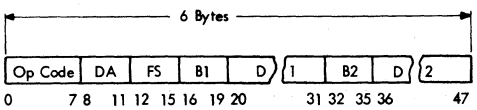


Figure 1-20. Transfer Instruction

Op Code in Hex	Name	Mnemonic	Type of Data Processed	Condition Code Influenced?
F1	Move with Offset	MVO	Any	no
F2	Pack	PACK	Any	no
F3	Unpack	UNPK	Any	no
F8	Zero and Add	ZAP	Decimal	yes
F9	Compare Decimal	CP	Decimal	yes
FA	Add Decimal	AP	Decimal	yes
FB	Subtract Decimal	SP	Decimal	yes
FC	Multiply Decimal	MP	Decimal	no
FD	Divide Decimal	DP	Decimal	no
D1	Move Numerics	MVN	Any	no
D2	Move Characters	MVC	Any	no
D3	Move Zones	MVZ	Any	no
D5	Compare Logical	CLC	Any	no
DE	Edit	ED	Dec/Logic	yes
DC	Translate	TR	Any	no
D0	Transfer I/O	XIO	Any	no

Figure 1-21. SS-Type Instructions

length must not be zero or else an error will occur.

Figure 1-21 shows the SS-type instructions.

Indexing

- Indexing is a method of address generation.
- The B field and the D field of an instruction are used to generate a new address.
- The B field addresses one of the eight general registers in the protected area to read out the base address.
- The base address is added to the D field (displacement) and the result is the new address.

All instructions that process data located in main storage have an address field for each operand, to be able to fetch the operand data from main storage (Figure 1-22). The address field is divided into a B-part and a D-part. Normally, both parts are used as one solid field, that is, the total binary value of the 14 low-order B/D bits represents the operand address. This is called direct addressing. It is possible to generate another address, however, from the same B/D field without changing anything in the instruction.

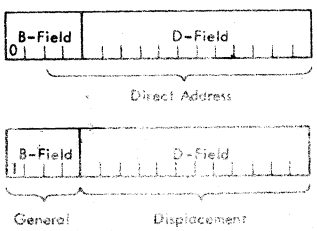


Figure 1-22. Indexing

The B field is not only part of the direct address, but may also become the address of a general register. The eight general registers in the protected area have binary addresses from 8 to 15. As the B field consists of four bits, it can address such a general register as soon as it contains eight or more. Thus, the B field reads out the content of a general register (the base address) and this content is then added to the D field (the displacement) of the instruction. The result of this add operation represents the new address. This is called effective addressing or indexing.

Indexing is performed by the micro-program which automatically checks the 8 bit in the B field to find out whether indexing is required. If an 8 bit is found, the indexing routine is performed. Indexing is ineffective when the addressed general register contains zero. It is also ineffective when the D field is zero.

DATA TRANSFER PRINCIPLES CPU - I/O DEVICES

- Input data is read into CPU via I/O devices.

- Input data is being processed in the CPU and the resultant data is subsequently transferred to an I/O device that produces the output.

- Data transfer between CPU and I/O devices (or vice versa) is a very substantial part of all processing operations.

- Data transfer can be performed in either burst mode, time sharing mode, or overlap mode.

Burst Mode

- During burst mode operations, the total amount of data specified in the I/O instruction (field length) is transferred in one single continuous flow.

- The flow stops when all specified data units have been transferred.

- During burst mode operations, the customer program is completely inactive (no new instruction issued).

- The termination of burst mode operations is indicated by condition code settings (Tape, Disk, etc.) or by requests for interrupt (card I/O devices not operating in time sharing mode).

Time Sharing Mode

- Time sharing mode is active when the Time Sharing switch on the customer console is in position on.

- Time Sharing allows the issue of a new customer instruction as soon as the previous one has been accepted.

- The continued customer program is interrupted by I/O requests which indicate that a data unit has to be transferred.

- Several I/O devices can operate at the same time.

- The I/O requests of the running I/O devices are satisfied depending upon a fixed priority scheme if more than one request is active at the same time.

- Operating in time sharing mode, the CPU uses the time between subsequential I/O requests (e.g., between two card columns to be read) to perform any other operations including data transfer from or to I/O devices.

Overlap Mode

- Overlap Mode can be considered as time sharing for I/O devices with a high data transmission rate (Disk, Tape, BSCA).

- The request of these high speed I/O devices are of highest priority.

- Contrary to time sharing operations, which require a control program routine (service phase) to transfer a data unit, the overlap mode data transfer is performed within one process (CPU) cycle only.

- Overlap operations can transfer two data units (bytes) at a time.

IBM 2020 CENTRAL PROCESSING UNIT (CPU)

- The CPU is main control unit of the whole system.
- The CPU processes data and controls input/output (I/O) operations.
- For I/O operations the CPU activates the control units (attachments, channels, control features) of the I/O devices.

OPERATING PRINCIPLES

- Machine language instructions are executed by micro-program.
- The micro-program consists of micro-instructions linked in a logical sequence
- The principles are shown in figure 1-23.

In order to solve a customer's problem, machine language instructions are linked to build a program (Figure 1-23). This machine program is loaded and stored in main storage. Machine language instructions are performed in series.

Machine language instructions are investigated (instruction phase, I-phase) and their specified operations are executed (execute phase, E-phase) by a sub-program, called Micro-Program. This micro-program consists of micro-instructions arranged in logical sequence necessary to execute the operation called for by the machine language instruction.

The micro-program is also stored in main storage, but in an area not accessible to the customer. This restricted area is called control area and is physically a part of the core storage array which also contains the customer's area.
The control storage area is specified by addresses higher than the highest possible customer's address (depends upon the customer's storage size)

Micro-Instruction Operating Principles

- Micro-operations are defined by the instruction Op code.
- Micro-operations require one to four processing periods.
- Only one processing period is executed when operating in CE Single Cycle mode.
- A processing period exists of delta cycle and cycle.
- The timing pulses 'delta cycle and cycle' are provided by the Long Time Clock.
- During a processing period 13 steps are performed, which are defined by the T-pulses of the Short Time Clock.

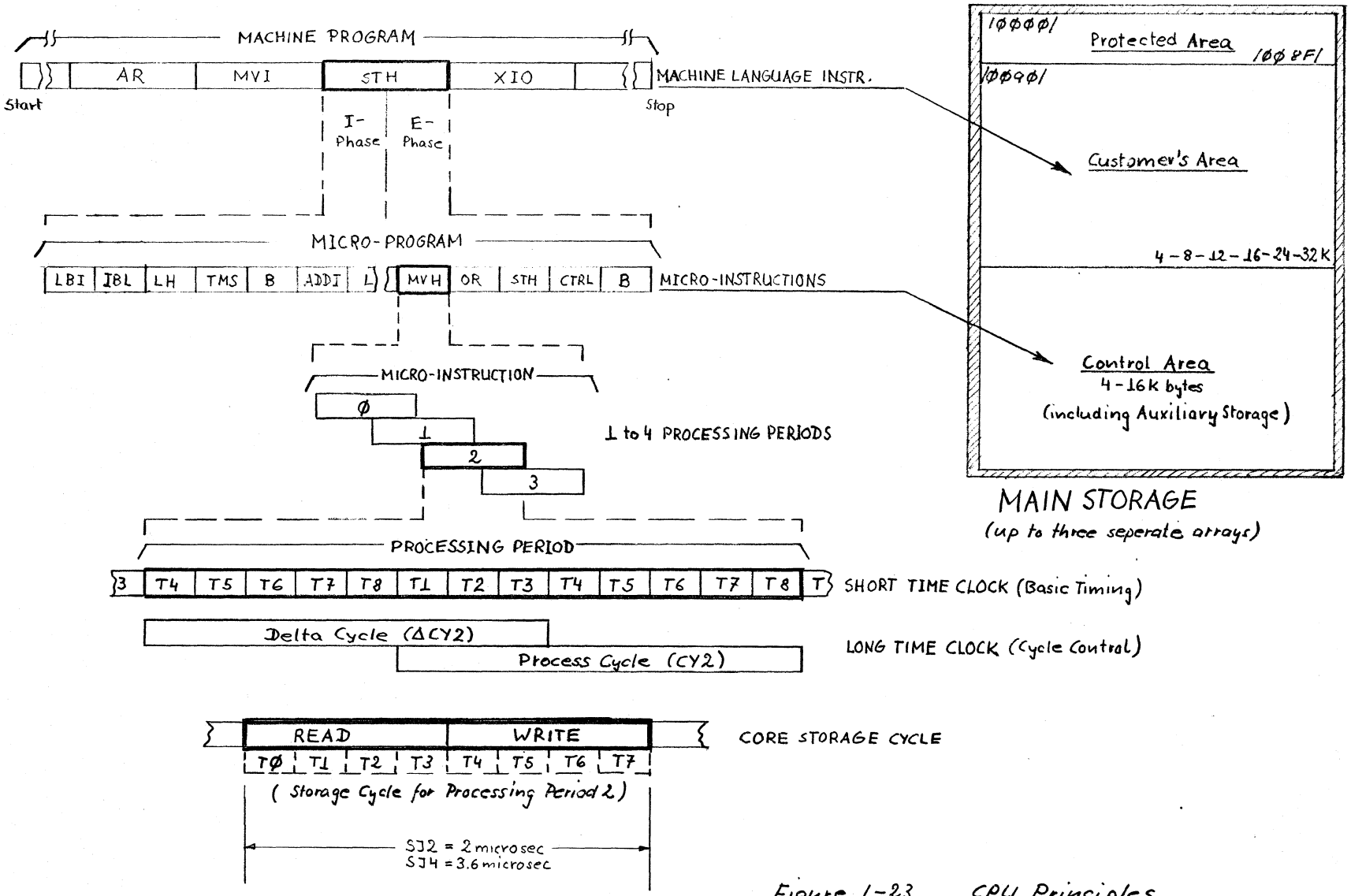


Figure 1-23 CPU Principles

- The Short Time Clock generates the pulses T1 to T8 continuously after power on.

Micro-Instructions are always one halfword (two bytes) in length. Their functions are specified by an operation code (Op code) in the highorder part of the instruction. The loworder part provides immediate data and/or register addresses.

For execution, the micro-instructions are read out from the main storage and set into the operation register (OP REG). Reading of micro-instructions and placing them into OP REG is controlled by circuits functioning during instruction start only.
The bit pattern of the micro-instruction in OP REG activates decode circuits which generate control signals. These signals directly control the CPU circuits to perform the specified micro-operation (Op code).

One micro-operation requires up to four processing periods. During every processing period only one main storage access is possible. Thus, the number of processing periods during a micro-operation depends upon the number of required storage accesses.

A processing period is defined by a delta cycle and process cycle. Delta cycle and process cycle are timing pulses provided by a latch ring. This latch ring is advanced by specified basic timing pulses generated in the Short Time Clock. The latch ring, defined as Long Time Clock, consists of two delta cycle latches and two cycle latches. The delta cycle latches are advanced at each start of delta cycle, while the delta cycle information is accepted by the cycle latches at start of process cycle.
The two delta cycle and cycle latches switch binary and provide delta cycles and cycles 0 to 3 (processing periods 0 to 3).

Each of the four processing periods is assigned to a fixed purpose which depends upon the one possible main storage access:

Processing period 0: Read out micro-instruction.

Processing period 1: Read out second (From) operand.

Processing period 2: Read out first (To) operand.

Processing period 3: Store result.

For example, a micro-instruction with both operands in main storage,requires all four processing periods in sequence when the operation result has to be stored. However, a micro-instruction with both operands in CPU registers, outside main storage only, requires processing period 0 (read out micro-instruction, one storage access).

Micro-instructions with either the from or to operand in main storage require besides processing period 0 , either processing period 1 or processing periods 2 and 3.
To execute only the required processing periods, the Long Time Clock can be advanced out of sequence by control signals. These control signals are generated in a circuit, called Cycle Control. Their generation depends on the micro-instruction

Op code and information where the operands are located (main storage or CPU register).

A processing period is divided in 13 steps. These steps are defined by timing pulses (T-pulses) of the CPU basic timing. The T-pulses are generated by a latch ring called Short Time Clock. Pulses A and B, generated by a crystal oscillator, advance the Short Time Clock. T-pulses are numbered from 1 to 8. Eight T-pulses define the process cycle period (T1 to T8). The required 13 steps of a processing period are achieved by overlapping processing periods in which the delta cycle consists of T4 through T8 plus T1 through T3.

A micro-operation is performed in functional steps such as moving data from one register into another. The functional steps to be executed are requested either unconditionally by circuits or by the micro-instruction Op code and are timed by processing periods (delta cycles and cycles) and T-pulses.

Example : The micro-instruction address has to be moved into the main storage address register (SAR) during processing period 0 (delta CY 0) at T5. This functional step is requested by circuits.

Functional steps can be performed simultaneously, if they use separate CPU circuits.

CPU BASIC OPERATIONS

- Basic operations are : Micro-operations, circuit controlled manual operations (MANOP's), and cycle steal operations.
- CPU basic operations directly control CPU circuits.
- The available logical CPU circuits determine the number of basic operations which can be performed.

Micro-Operations

Micro-operations are specified by micro-instructions. The micro-instruction bit pattern in OP REG directly controls the Logical CPU circuits by generating main control signals. These main control signals are timed by delta cycles and cycles, and by pulses T1 to T8.
The 43 available micro-instructions are given in FEMDM, 2020 CPU, figure 5-0. Micro-operations process data available in main storage or in a work storage device called Local Store (LS).

Local Store

The LS contains 64 halfword registers. For addressing a single LS register, the LS is divided into eight zones (0 to 7). Each zone contains eight registers (0 to 7).
Each zone is assigned to one of eight possible program levels (PL's). PL's 1 to 7 are caused by I/O requests. PL 7 indicates the highest priority I/O request. PL 0 is the CPU operating level which is active when no I/O request has to be served.
The registers 0 to 6 in a zone are general purpose registers. They are addressed by the To- or From-Reg fields in the micro-

instructions.

Register 7 of each zone is used as Instruction Address Register (IAR). During micro-operations the IAR contents are incremented by 2 to obtain the address of the next sequential micro-instruction (NSI). For branching the IAR contents are changed. Not that changing the PL by a I/O request (addressing another zone) causes a branch to the address in IAR of the new zone. The IAR (LS register 7) is only addressable in store-type micro-instructions. Load-type instructions, addressing the IAR, are treated as No Operations to prevent erroneously changing of the instruction sequence.

For FF or I/O format instructions the contents of the register specified in the To- or From-Reg fields are used as data if the highorder bit of the field is off (direct addressing) or as main storage operand addresses if the bit is on (indirect addressing).

For FF and I/O format instructions the operand address is automatically updated (incremented or decremented) according to the processed data format (byte or halfword).

Micro-Instruction Formats

The three basic instruction formats are:

1. Immediate instruction format (RI format)
   Note: I/O instructions are a special case of the RI format. (See "I/O Micro-Instruction Formats".)

2. Instruction format with displacement addressing (RD format)

3. Register to Register instruction format (FF format).

The RI format allows data transfer and logical operations between a register and an immediate data byte in the instruction. The data transfer or operation takes place between the specified To-REG and the immediate data byte. The result is stored in the To-REG.

The RD format provides a displacement address one byte in length. This displacement address is used as the loworder byte of a halfword address. The highorder byte of this address is the highorder byte of the IAR (block address).

The To-Reg field specifies the LS register, the content of which is tested to decide whether branching is performed or not.

Instruction bit 15 on (indirect addressing) indicates that the combined address is used to read out the final branch address.

Exceptions:

LH. With this instruction the specified LS register is loaded with the contents of the addressed halfword.

STH. With this instruction the contents of the specified LS register is stored in the addressed halfword.

For both instruction indirect addressing is performed if bit 15 is on.

The FF format allows data transfer and logical or arithmetic operations between two registers or data fields in core ad-

dressed by the contents of these registers. There exist four different types:

1. Both registers are used as data registers (DD type)

2. One register is used as data register and the other is used as address register specifying a 'FROM ADDRESS'. (DX type)

3. One register is used as data register and the other is used as address register specifying a 'FROM ADDRESS'. (DX type)

4. Both registers are used as address registers specifying a 'FROM ADDRESS' and a 'TO ADDRESS' (XX type).

The operand address of the most FF format instruction are checked when the AC bit is on.

Address checking involves:

1. Checking of halfword boundaries (error: odd address, effective only for halfword operations).

2. Checking of available customer's storage (error: exceeds highest possible customer's address).

3. Checking for protected area (error: address within the first 144 bytes)

An address check causes PL 2 by request if the CPU is working in PL 0. In higher PL's, the check is not effective, but the addresses can be checked by means of a BAC-instruction. For address checks in PL's 1 to 7, the instruction is executed with the deviations indicated in Figure 1-24. For odd halfword addresses, the next lower even address is used.

Automatic length count (ALC) is available for several FF format instructions of the xx-type. This feature allows a string of halfwords/bytes to be processed by one instruction. The automatic length count is activated when LS Register 3 is used as To-operand, and LS Register 5 as From-operand. For operations which need two length - codes (L1/L2), the LS registers are used. Operations with one length count (L) use the contents of LS Register 1. The content of a length count register indicates the number of bytes to be processed minus one (or two for halfword operations). The ALC operation ends when the field lengths are decreased below zero (registers contain /FFFF/ for byte operations or /FFFE/ for halfword operations.
Note: Besides the FF format instructions, ALC is also available for I/O instructions when the To-Reg field address LS register 7 and indirect addressing is specified.

The setting of condition code for the binary and logical instructions is programmable (bit 6 of the instruction). The decimal instructions will always set a condition code.

For the binary instructions AH, SH, AHSC and SHSC and for the logical instructions, AND, OR, EOR and CLC when used in

Program Level	Address Checking specified	Customer Storage Fetch Operation	Customer Storage Store Operation	Control Storage Fetch Operation	Control Storage Store Operation	Not existent St. Fetch Operation	Not existent St. Store Oper.
0	yes	X	X	XT	RT	OT	NT
$\overline{0}$	yes	X	X	X	R	O	N
0, $\overline{0}$	no	X	X	X	X	O	N

X: Data are fetched/stored
R: Storagedata are regenerated, i. e. storage data are not destroyed.
0: Zeros are used instead of storage data.
N: Data are not stored but lost.
T: PL switching is performed.

Figure 1-24 Address Checking

XX format with auto length count and setting of condition code specified (bit 6 of the instruction), these instructions should be preceeded by a CTRL-instruction, which sets the condition code to zero (1000).

The decimal instructions, AP, SP, ZAP and PPC should always be preceeded by a CTRL instruction.

For AP and ZAP this CTRL instruction sets the condition code to zero (1000) and the carry latch off.

For SP and PPC this CTRL instruction sets the condition code to zero (1000) and the carry latch on.

For all these instructions with auto-length count the setting of the condition code is inhibited after a setting of the condition code other than zero is detected.

The overflow latch of the condition code latches can also be forced on if:

1. With an AP instruction in XX format the carry latch is on at the end of the operation.

2. By all decimal instruction, AP, SP, ZAP, PPC in XX format with auto-length count when L1 L2 and the exceeding positions of the second operand are not zero,

Binary instructions: AH, AHSC, SH, SHSC

Condition Code:

1000	Result is zero
0100	Result is less than zero
0010	Result is greater than zero
xxx1	overflow

Logical instructions:	AND, OR, EOR	
Condition Code:	1000	Result is zero
	0100	Result not zero
Logical instruction:	CLC	
Condition Code:	1000	Operand 1 equals Operand 2
	0100	Operand 1 is smaller than Operand 2
	0010	Operand 1 is greater than Operand 2
Decimal instructions:	AP, SP, ZAP, PPC	
Condition Code:	1000	Result is zero
	0100	Result is not zero
	xx01	Overflow

During arithmetical operations a carry out of the loworder byte (byte operations) or out of the highorder byte (halfword operations) turns on the Carry Latch.

In addition, the carry latch can be set, together with the condition code, by a CTRL-instruction. The contents of the specified register of this CTRL-instruction specifies the condition code (bits 12-15) and the carry latch (bit 11).

In all PL's other than PL 0, where the microprogram can set/reset condition code or carry latch, the value of the condition code and carry latch must be saved by a SENS-instruction. Before leaving the Program Level, the original value of the condition code and carry latch should be restored by means of a CTRL-Instruction.

I/O Micro-Instructions

The two I/O micro-instructions are Sense I/O and Control I/O. The Sense I/O instruction fetches data or information from the attachments, while the Control I/O instruction sends data or control information from the CPU to the attachments.
The attachment point to be sensed or controlled is specified by the device address field in the instruction. Since this field is one byte in length, 256 different attachment points can be specified.
Note: CPU is considered as an additional attachment. Normally the device address is expressed by two hexadecimal digits.
The To-Reg field in the instruction defines any LS register from 0 to 6 and the high-order bit of the To-Reg field decides whether the specified LS register is used as data register (direct addressing) or as address register (indirect addressing).

Both instructions are executed with ALC when the To-Reg field contains 7 and indirect addressing is specified.
LS register 1 contains the field length of the ALC operation. LS register 6 contains the address, addressing the storage location where I/O data is to be stored for SENS or where the I/O data is available for CTRL.
The length count is decremented by 1 for each transferred byte.

The operation ends when the LS register 1 is decremented below zero (contains /FFFF/).

The contents of address register 6 is incremented by 1 for each transferred byte.

Circuit Controlled Manual Operations (MANOP's)

Generally, manual operations are selected by the mode switch on the customer console. The selectable manual operations can be divided into two groups:

1. Manual operations performed by micro-program. These operations are Instruction Step, Address Stop, Display Register, and Alter Register (see micro-program flowcharts FEMDM, 2020 CPU, Appendix B).

2. Circuit controlled manual operations. These manual operations are:
   - Storage Alter
   - Storage Fill
   - Storage Display
   - Storage Scan
   - Storage Test
   - LS Register Alter
   - LS Register Display
   - Initial Control Program Load (ICPL)
   - CPU LOG in

With the exception that the Op code is simulated by switch settings or key operation, these circuit controlled manual operations (MANOP's) can be considered as basic operations similar to micro-instructions.

MANOP's are also performed in processing periods. Within the processing periods functional steps are performed as during micro-operations. The main signals, necessary to control the CPU circuits, are generated depending upon turned on MANOP control latches and switch positions.

MANOP's, processing strings of data, operate in a way similar to ALC micro-instruction.

Storage Alter/Fill

The byte, set up in the two data switches, is moved into an addressed main storage position. For Storage Alter, only the main storage address set up by the four address switches is affected. Storage Fill can be considered as Storage Alter operating in ALC mode. The start address, provided by the address switches, is incremented during every byte movement. After the highest possible halfword address is exceeded (/FFFF/ +1), storage Fill continues with address /0000/.
Storage Fill ends when the Stop key is operated.

Storage Display/Scan

An addressed byte is displayed in the data register (DR-) U-L indication. The basic operation performed is similar to Alter/

Fill with the exception that the byte set up in the data switches is not moved into main storage. Storage Scan can be considered as Storage Display operating in ALC mode. Scan investigates the main storage contents for correct parity. Incorrect parity is recognized by the Shift Unit Check.

Storage Test

Storage Test can be considered as a modified Storage Fill operation (mode switch to Storage Fill and CE Storage Test switch on). Storage Test is performed by four different runs. A run covers the main storage addresses /0000/ to /FFFF/. During runs 1 and 2 each core storage halfword position is loaded with its address and the address is read out again for compare. Runs 3 and 4 perform the similar functions but loading and comparing is performed inverted.

LS Register Alter/Display

The mode switch is set to Alter or Display Register and the CE mode switch has to be turned on.
The LS register, defined by data switch 2 of the LS zone specified by data switch 1, is altered by the halfword set up in the CE Select switches and displayed in DR - P, I, U, L. For LS register Display altering is suppressed.

Note: For LS register alter, the just loaded LS halfword is read out again and compared against the origin halfword in the CE Select switches. If the compare is unequal Process Check occurs without any other CPU check indication.

ICPL

ICPL can be considered as a complete card read XIO instruction (machine language) including service phases. ICPL controls the CPU circuits to read the control program loader card into main storage. After the loader card has been read, CPU operates under control of the micro-instructions previously read in from the loader card.

CPU LOG in

CPU LOG operates like a four cycle micro-instruction. This operation stores four halfwords of CPU status informations into a defined main storage area. CPU LOG uses separate circuits so that the LOG information can be saved even if the normal CPU circuits fail to operate.
CPU LOG in is always performed before System Reset.
The System Reset routine (micro-program) is performed for Power-on, System Reset key operated (after Check), and for Load key operated.

Cycle Steal Operations

Cycle Steal Operations can be considered as circuit controlled service phases during which I/O data is directly transferred from main storage to the requesting attachment or vice versa.

Cycle Steal operations, requested by the Cycle Steal devices, interrupt any current micro-instruction for one processing period (cycle). Four different Cycle Steal devices can be attached. The cycle steal requests from these four devices are served depending upon a priority sequence.

I/O INTERFERENCE

- There are two types of I/O interference:
  I/O Service Phases
  Cycle Stealing

I/O Service Phases

Service Phases are micro-program routines, which handle the I/O interference caused by I/O devices with a relative low data transfer rate. An I/O service phase is a part of the execution phase of a machine language I/O instruction and is initiated by I/O service request generated in the attachments. I/O Service Phases interrupt the execution of machine level instructions.

Because there are I/O devices covering a wide range of data rates, some priority has to be established between Service Phases in order to avoid Data Overrun when different devices request service simultaneously. Also, some Service Phases should be interruptable for the benefit of others of a higher priority.

This is accomplished by grouping Service Phases into Program Levels. Service Phases in the same Program Level cannot interrupt each other, those in different Program Levels can.

Program Levels

The CPU proper can conceptually be considered to consist of eight sub-processors which share time and circuits among them in a fixed priority scheme. These sub-processors are called Program Levels (PL). They are numbered from 0 through 7. The higher number indicates the higher priority. To each PL, a local storage zone is assigned. Thus, each PL is controlled by its own IAR. PL 0 (LS zone 0) is assigned to the CPU in its processing state, (CPU, executing machine level instructions). I/O Service Phases are assigned to PL 1 through to 7 (I/O levels). In general, several Service Phases will share one PL.

More than one PL may be on at any time but only one may be active, i.e., executing micro-instructions by using its associated zone of Local Storage. The PLs being on but not active are called pending PL's.

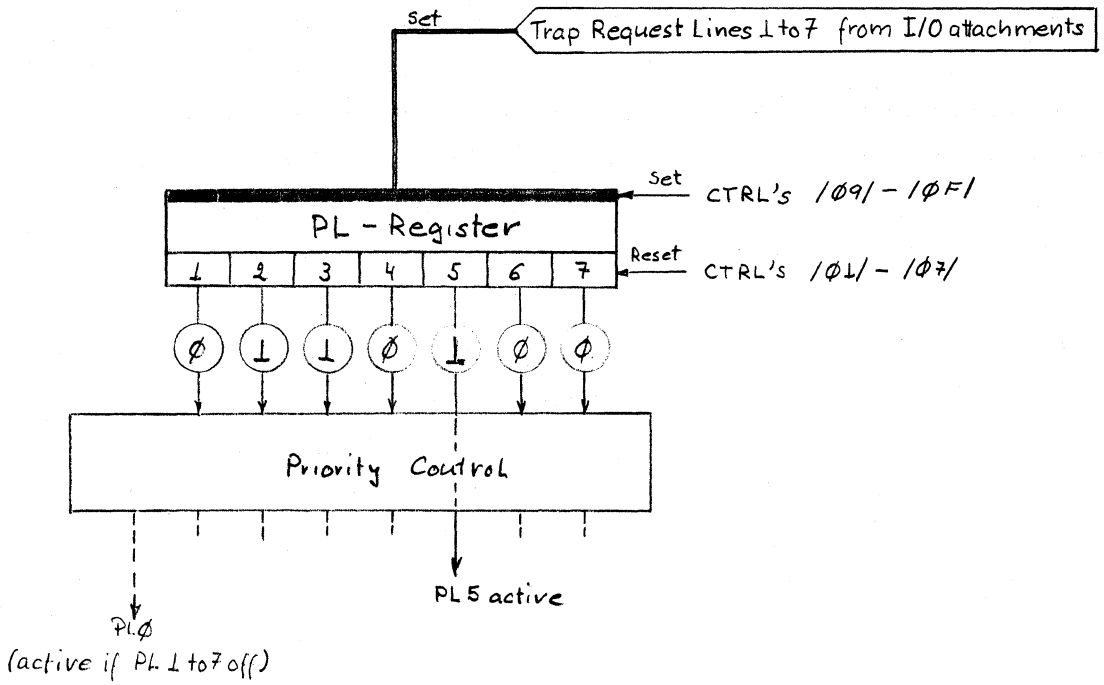


Figure 1-25 Program Level Control

A pending PL will become active if all higher levels are switched off. Otherwise, a presently active PL may become pending (that is processing in this level stops) when a higher priority level is switched on.

To control the Program Levels, the CPU contains a seven position PL register. A priority circuit selects the highest active level out of all which may be on at a given time (figure 1-25).
PL 0 (CPU level) is active when all PL register positions are off.

Program Level Switching

Any PL is switched on by a trap request (I/O request). There are seven trap request lines from the attachments to the CPU, one for each PL (1 to 7.) There is no trap request line for PL 0.

If coincidence occurs, a trap request overrides an attempt to reset a PL by a PL-CTRL. Note that a PL can never be reset by trap-requests.

Any PL is switched off by a reset-PL instruction, which is a CTRL micro-instruction with a device address from /01/ to /07/. The device address indicates the PL to be reset.
Any PL is switched on by a CTRL with a device address from /09/ to /0F/. The three loworder bits of the device address define the PL to be switched on.

Program Trapping

If as a consequence of PL switching another PL becomes active, the CPU selects another LS zone. This is called Program Trapping.
The trap may be upward(higher PL is switched on) or downward in priority.
When all service phases for the currently active PL have been served, the corresponding trap request latch in the attachment is turned off by suitable CTRL-instruction.
The PL itself is reset by a reset PL-CTRL.
Trapping takes place at end of the current micro-instruction or at the end of the current 'loop-cycle' of an ALC instruction but may be delayed by intervening Cycle Steal requests. Thus, excepting Cycle Stealing, a Trap is never delayed by more than four cycles after PL-switching occurs.
LS registers 0 to 7 of any zone, de-selected by trapping, are left unchanged. The IAR points to the instruction which would have been executed next if no Trap occurs. For Traps after branch-type instructions the IAR will point to the branch-address. For ALC instructions interrupted by a Trap, the LAR points to the same instruction. These instructions are re-entrant points after each loop-cycle.

Cycle Stealing

Cycle Stealing handles I/O devices with high data transfer rates (above 5 K bytes/sec) such as magnetic tapes and magnetic disk drives. Cycle Stealing consists in sharing the core memory and part of the local storage between the I/O device and CPU control program. Two LS registers (data address and field length) are used by a device at a time. One memory cycle is executed at the next possible time. Thus, Cycle Steal requests have the highest priority for memory utilization and may even interrupt the cycle sequence of micro-instructions.

CPU DATA FLOW

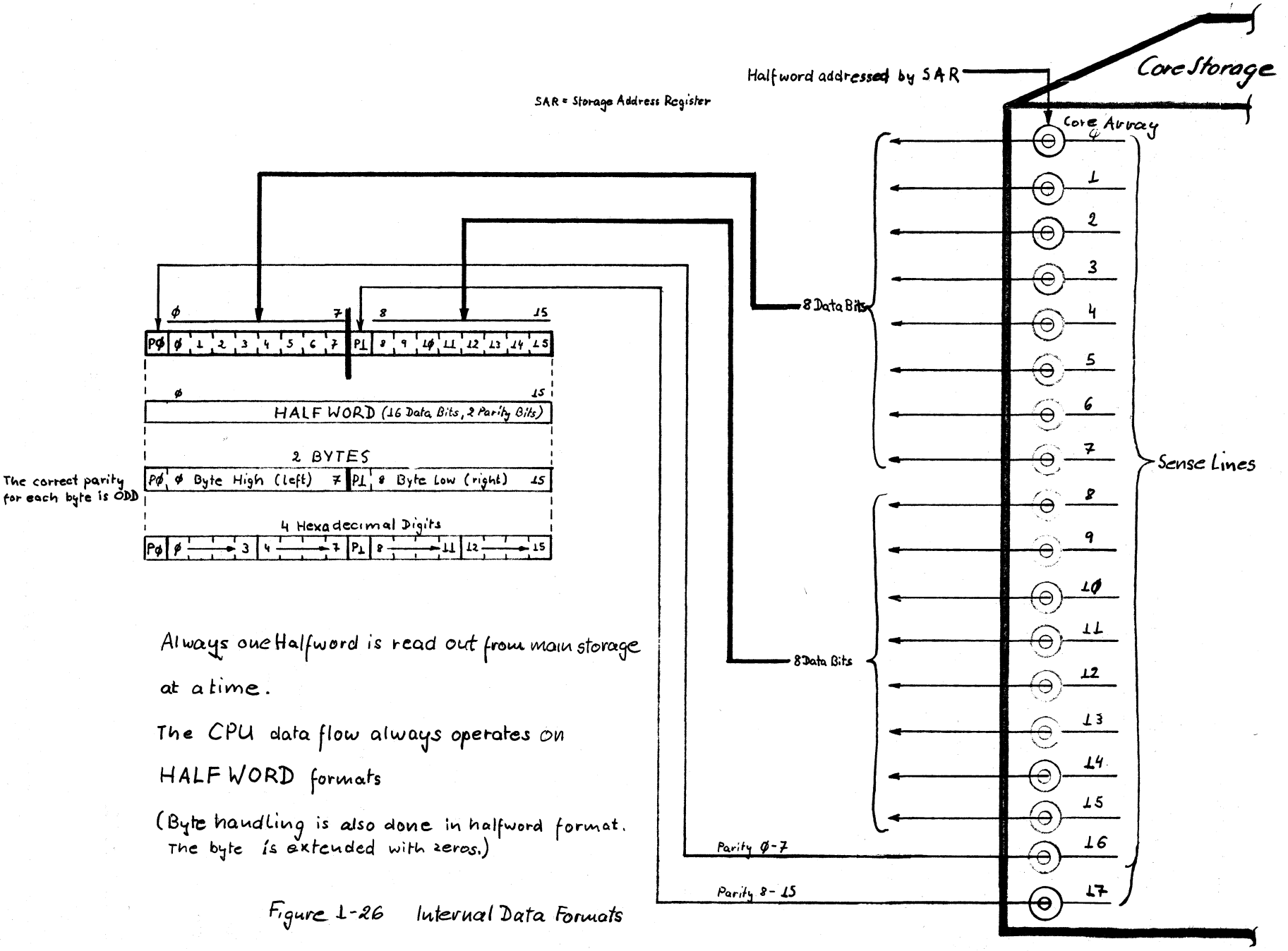
- The CPU data flow is shown in FEMDM, 2020 CPU, figure 3-2.
- The CPU data flow handles halfword data.
- Byte handling is performed by completing the data byte to a halfword either by zeros or by a data byte from another source.

Internal Data Formats

- The internal data formats are shown in figure 1-26.

Main Storage

- The main storage contains the customer's area as well as the control area.



Always one Halfword is read out from main storage at a time.

The CPU data flow always operates on HALF WORD formats

(Byte handling is also done in halfword format. The byte is extended with zeros.)

Figure 1-26 Internal Data Formats

- The available storage types are SJ 4 and SJ 2.
- During each core storage cycle one read and one write operation is performed (read or write cannot be performed separately).
- A core storage cycle is divided into eight T-pulses. Four T-pulses for read and four for write.
- A core storage cycle lasts 3.6 micro-sec (SJ 4) or 2 micro sec (SJ 2).
- The core storage cycle time defines the CPU cycle time, however, both cycles are not aligned. (See Figure 1-23).
- Each addressable storage position contains one halfword.
- SJ 4 and SJ 2 are available as 4 K and 8 K halfword arrays (8 K or 16 K bytes).
- The maximum storage size is 24 K halfwords (48 K bytes customer's and control area ; three arrays on three boards).
- The third board is defined as extension (additional circuits for sensed data and inhibit).

Storage Address Register (SAR)

- The SAR is a halfword register, which saves main storage addresses to be available for read/write time.
- The addresses in SAR define bytes or halfwords.
- For byte handling the unit bit (SAR bit 15) defines the required byte in the halfword read out or to be stored.
- The SAR is parity checked.
- The auxiliary storage areas (512 bytes for each 4 K halfwords) are part of the control area.
- Aux storage addresses are specified by the SAR bits 0 and 1 on (byte addresses above 48 K).
- For main storages smaller than 48 K bytes, there is an addressing gap between the highest storage address and the first Aux storage address (/C000/).
- Addressing of not available storage positions (gap positions or not available Aux storage positions up to 64 K cause 'read out of blanks' .
- For 'reading of blanks' the parity bits are forced on (in SDR) to provide a valid halfwords containing zeros.

- For CPU LOG in, main storage addresses are forced by circuits.
- SAR can be displayed on the CE console.

Storage Data Register (SDR)

- The SDR is a halfword register set by the main storage sense bits 0 to 17.
- Sense bits 0 to 15 are the data bits, while bits 16 and 17 are the parity bits P0 (highorder byte) and P1 (low-order byte), respectively.
- The sense bits occur during read time of the core storage cycle.
- The SDR provides data to the CPU data flow, to the Cycle Steal data bus (bytes or halfwords), and to the Inhibit Switch (bytes or halfwords for regeneration during write time).
- SDR can be displayed on the CE console.

Inhibit Switch

- The Inhibit switch selects sources which provide data (bytes or halfwords) to be stored.
- Data to be stored is provided by the SDR (regeneration), ALU, or by the Cycle Steal data bus.
- The halfword to be stored is parity checked.
- During CPU LOG in, the parity check circuit is used to test the LOG informations at read time.

Operation Register (OP REG)

- During processing period 0, the OP REG accepts the SDR halfword (micro-instruction).
- The micro-instruction in OP REG remains during its execution until the next instruction is read in.
- For I/O micro-instructions, the low-order byte containing the device address activates the I/O Address Bus.
- The OP REG output feeds the Operation Control Unit and the LS addressing circuits (register selection, X-addresses).
- The OP REG can be displayed on the CE console.

Operation Control Unit (OCU)

- The OCU is an imaginary unit considered to generate the CPU control signals and to provide all necessary timings.

- Some control conditions can be displayed on the CE console (OCU 1 and 2).

Program Level Control (PL Control)

- The PL Control consists of PL Reg, Priority Decision, and Current PL Reg.
- The PL control is activated by Trap request from the I/O attachments or by CTRL micro-instructions.
- The PL control defines the current work zone in the Local Store (Y-addresses).
- The PL Reg can be displayed on the CE console (OCU 2).
- The Local Store Address Check (LSA check) tests that always one X- and Y- address is active at a time.

Local Store (LS)

- The LS consists of 64 halfword registers separately addressable.
- Reading of a LS register does not destroy its content.
- The LS registers are arranged in eight zones (0 to 7) specified by the PL control (Y-addresses 0 to 7).
- Each zone contains eight registers (0 to 7) defined by X-address.
- LS register 7 in each zone contains the Instruction Address Register (IAR) which provides the address of the micro-instruction performed as the next.
- Writing into LS is controlled by the LS Write signal, while reading is controlled by the sense amplifier gate (SA gate).
- LS writing and reading is performed within one T-pulse period.

Modify Address - Register (MAR)

- The MAR is a halfword register which is set by LS data only.
- The LS data may be addresses or field lengths which have to be updated by the Modifier.
- The MAR can be displayed on the CE console.
- The MAR output feeds the Modifier as well as test circuits which investigates the MAR contents for being zero, minus, plus, or being an invalid address (Address Check, outside customer's area).

Modifier

- The Modifier is capable of incrementing or decrementing halfwords by one or two.
- Apart from being incremented or decremented, data can flow through the Modifier unchanged.
- The Modifier output is parity checked.
- The output parity is predicted depending upon the input parity (MAR) and the operation to be performed.
- The Modifier result can be placed back into LS only.

Logical Unit

- The Logical Unit is a triple combination consisting of Shift Unit (SU), Invert Switch and Arithmetic-Logic Unit (ALU).
- The ALU connects two halfword operands, provided by SU and Invert Switch either arithmetically (ADD) or logically (AND, OR, OE).
- To-Data Register (TDR) and From-Data Register (FDR) save the input operands to be available until the ALU result is stored.
- TDR, FDR, and the ALU result can be displayed on the CE console.

Shift Unit (SU)

- The SU is divided into three functional groups able to modify the To-operand:
  1. Eight Shift (shift left/right by 8, byte shift)
  2. Shift Unit (shift left/right by 2 or 4)
  3. Suppress Unit
- TDR halfwords are fed through the Eight Shift either direct or cross-shifted (byte exchange).
- Both Eight Shift circuit operations are controlled by signals.
- The Eight Shift sets a TDR byte to zero (parity is corrected) when no control signal for the corresponding byte is generated.
- The resulting Eight Shift halfword (INTernal SU Bits) are parity checked (SU Check).
- The SU check tests the data read out from main storage (SDR-TDR-Eight Shift).
- The Loworder byte after Eight Shift is set onto the I/O Data Bus to be used as control data for CTRL micro-instructions (CPU data to attachments).

- The Shift Unit handles the INT SU halfword either for shifting left/right by 2 or 4 bits or for no shifting.
- No shift left/right and not no shift control forces the affected byte to zero.
- The Shift Unit contains circuits for normalizing packed decimal signs according to ASCII or EBCDI code (as selected).
- The Shift Unit provides test circuits, which investigate packed decimal data for being valid.
- The Suppress Unit suppresses the SU halfword partially (in defined bit groups) or completely.
- The output of the Suppress Unit, which provides the modified TDR halfword to the ALU, is displayed in DR's P, I, U, L on customer console.

Invert Switch

- The Invert Switch transmits the FDR halfword either true or inverted
- The two bytes can be controlled separately.
- The true and invert functions are controlled by signals.
- Simultaneous true and invert control forces zeros.
- Simultaneous no true and no invert control forces ones.
- The halfword at Invert Switch out is displayed in DR's - E, S, T, R on customer console.

Arithmetic-Logic Unit (ALU)

- SU and Invert Switch halfwords are used as ALU input operands
- The ALU connects the input operands either by ADD, AND, OR, or Exclusive OR (OE).
- ADD, AND, and OR functions are controlled by ALU gates, while the OE function is always performed when no ALU gate is active.
- The ALU provides Six Correction circuits used to generate valid packed decimal digits (ALU loworder byte only).
- A carry latch recognizes the carry out of the highorder ALU byte (halfword operations) or carry out of the loworder byte (byte operations).

- The ALU result condition sets four Condition Code latches respectively (micro-program condition code).
- The ALU output is parity checked (ALU Check).
- The ALU parity (highorder and loworder byte) is predicted depending upon the parity of the input operands and the performed ALU function.
- The ALU result can be saved in LS, used for addressing (branch), or stored into main storage.
- Either the complete result halfword, or the loworder byte is stored.
- For storing a byte only, the Inhibit Switch halfword is completed by a SDR byte (regeneration).

I/O Bus Out and In

- The CPU communicates with the I/O attachments via the I/O Bus only.
- The I/O Bus consists of an Address Bus and a Data Bus.
- Besides Address and Data Bus, able to transport one byte at a time, a small number of control bus lines is provided.
- The Address and Data Bus are line loops which leave CPU (device addresses SENS or CTRL from OP REG loworder byte; CTRL data from loworder byte after Eight Shift) and enter CPU by setting into FDR.
- For SENS, the Data Bus is activated in the corresponding attachment by data provided by a source which is selected according to the device address.
- For CPU internal SENS's the sensed information is directly set into FDR.
- CPU SENS data are either of byte or halfword format.
- The Data Bus entry is parity checked (Bus Check).

CPU Cycle Steal Control

- The CPU Cycle Steal Control is activated by four Cycle Steal requests generated in the Cycle Steal devices (part of the attachments).
- All control information necessary to execute cycle stealing is provided by the Cycle Steal devices.
- Status information of the executed cycle steal operations returns to the

- Cycle Steal devices (e.g., a CS Inhibit Check does not stop CPU, but the information is saved in the CS device).

- Cycle Stealing stops all current CPU operations for one cycle.

- The CPU Cycle Steal Controls data transfer (byte or halfword) from or to CPU via the Cycle Steal (CS) Data Bus.

CPU BASIC TIMING

- The CPU Basic Timing circuits and a corresponding timing chart are shown in FEMDM, 2020 CPU, figures 4-30 and 4-31.

OSCILLATOR

- The oscillator provides the pulses A and B and the Oscillate pulse.
- Depending upon the used core storage type, SJ2 or SJ4, the oscillator frequency is either 4 or 2.2 MHz.
- Pulses A and B are adjustable.
- The Oscillate pulse controls the generation of the two I/O Clock pulses (1, 2) and the Storage select signal.
- Storage Select is used for SJ2 only.

The oscillator consists of a crystal oscillator and pulse former stages which provide three different pulses at the oscillator output. These pulses, Pulse A, Pulse B and Oscillate, define the basic CPU operating timing.

The crystal oscillator produces pulses of a constant frequency. Depending upon the used core storage type, SJ2 or SJ4, the oscillator frequency is either 4.000 MHz (SJ2) or 2.222 MHz (SJ4); thus, an oscillating period lasts either 250 nanoseconds (SJ2) or 450 nanoseconds (SJ4). The oscillator circuits contain two potentiometers which allow Pulse A and B adjustments (Figure 2-1). Potentiometer 1 adjusts the distance between the raising slopes of Pulse A and Pulse B. Potentiometer 2 adjusts the duration of Pulse A and B.

The + Oscillate pulses control the Storage Select signals (SJ2 only) and the latch ring of the four Binary Latches which provide the I/O Clock Pulses 1 and 2.

To generate the Storage Select signal at the correct time (within T2 or T6), the + Oscillate pulse is delayed (delayed Oscillate) by a delay line which is adjustable from 0 to 125 nanoseconds by changing tabs. Pulse A and Pulse B control several internal CPU functions, which are executed within one T-pulse period.

SHORT TIME CLOCK

- The Short Time Clock provides the timing pulses T1 through T8
- This clock is continuously running, controlled by pulses A and B, when System power is applied.

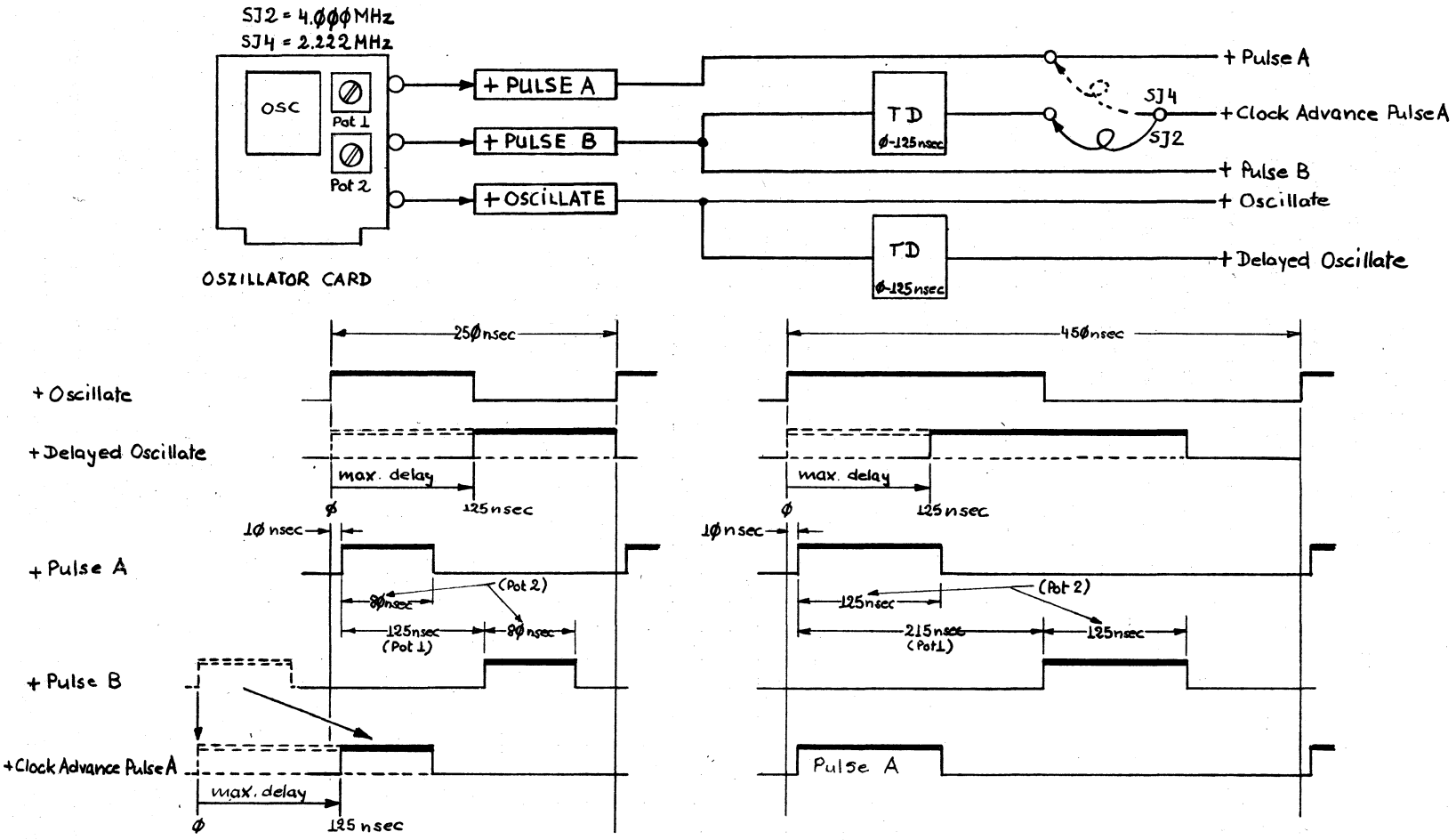


Figure 2-1 CPU Basic Timing

- For SJ2 the Clock Advance Pulse A is adjustable to move pulse B as far as possible to the end of a T-pulse period.

The Short Time Clock consists of the Short Time Latches and the decoder circuits. The decoder circuits provide eight timing pulses numbered from 1 to 8 (T1 to T8).

The pulses T1 to T8 are generated depending upon the switching sequence (figure 2-2) of the Short Time Latches 1, 2 and 3. These latches are advanced by each Clock Advance Pulse A. The output of the latches are also used to define processing periods which are longer than one single T-pulse.

These periods are :

Read Cycle SJ4 = T6 - T1

Write Cycle SJ4 = T2 - T5

CS LS Select Time = T5 - T2

Time T8 - T3

Time T4 - T7

The delta Short Time Latches 1, 2 and 3 switch in the same sequence as the Short Time Latches. Since the delta latches are advanced by the Pulse B, their switching is delayed for a half T-pulse period (half of an oscillator period). This allows generation of the delta T8 pulse and especially

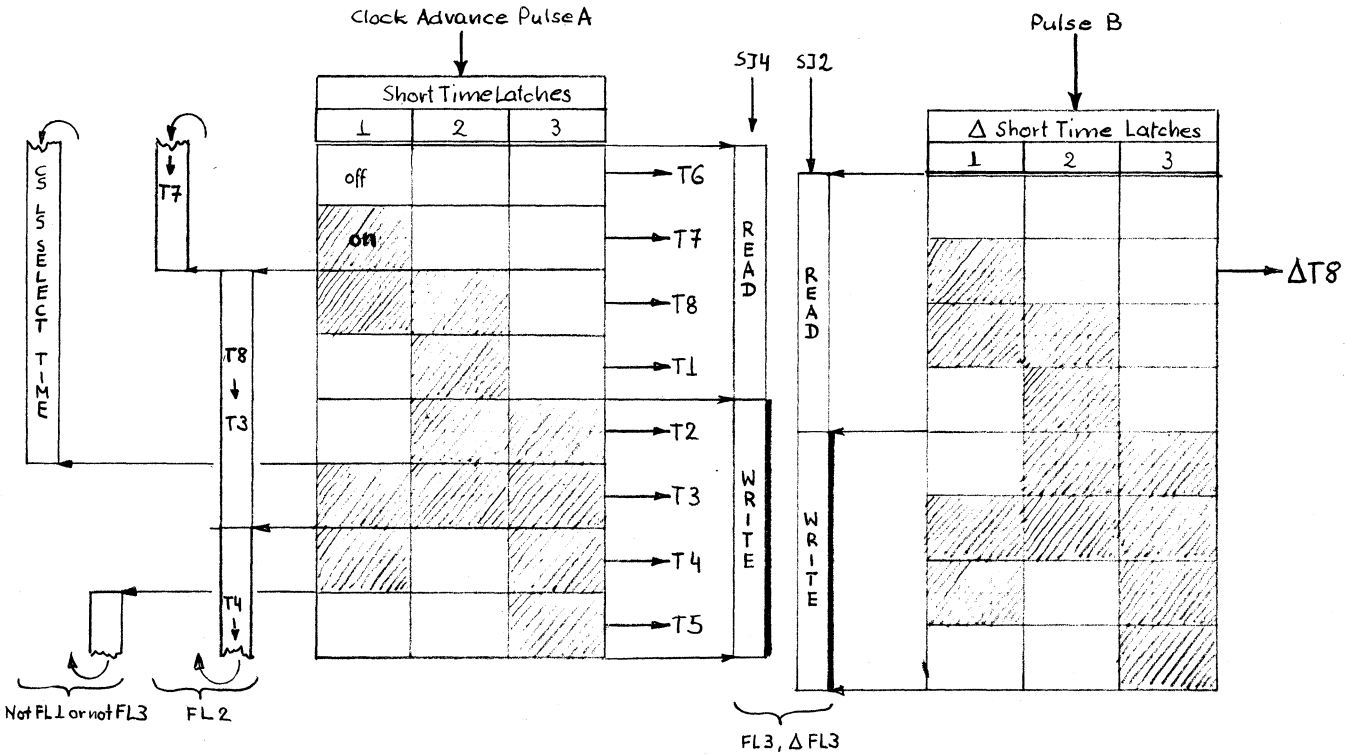


Figure 2-2 Short Time Clock Switching

for SJ2 shifting of the Read and Write Cycle for a half of a T-pulse.

The source of the Clock Advance Pulse A (defines the start of the T-pulses) is different for SJ2 and SJ4. For SJ4 the Pulse A is directly used as Clock Advance pulse. For SJ2 the Clock Advance Pulse A is generated by the delayed Pulse B.
The delay is adjustable from 0 to 125 nanoseconds. If the maximum delay is selected,the delayed Pulse B occurs at the same time as Pulse A.

RUN CONTROL

- The Run Control contains all circuits necessary to start or stop the CPU, as well as the circuits to generate all reset signals.

- The CPU is started when one of the following keys is operated :
  - Power On key
  - Start key
  - System Reset key
  - Load key

The Run Control circuits control CPU start and CPU stop. Besides the start-stop control, the circuits provide the necessary reset signals. Run Control operations are initiated by operating the keys on the customer console or on the remote control box which is connected to the socket on the CE console. The remote control box attached blocks the function of the console start key (Block Normal Start). The CPU is started by:

1. Power On (to perform System Reset).
2. Operating the Start Key (remote control box or console; Normal Start).
3. Operating the System Reset Key.
4. Operating the Load Key.

When the CPU is stopped while any I/O device is still operating, the CPU is restarted either by UNEQUAL PL (Trap request) or by ANY CS REQUEST. This restarting allows completion of the current I/O operation even if the CPU has been stopped before.

The Initial Control Program Load (ICPL, circuit control reading of the control program loader card) is performed by a sequence of start-stop operations (restarts by Trap requests 5).

CPU START

The CPU operates when the delta Process latch and the Process latch are turned on. Thus, CPU start can be considered as a switching sequence (figure 2-3) of several start latches which results in turning on the two process latches.

CPU Start by Power On

- Power On starts CPU to perform the system reset routine.

During power on the logical voltages reach their operating levels at different times. This causes undefined activating of CPU control components (latches, registers). To ensure a defined CPU start situation, the System Reset Routine is performed. However, performing the System Reset Routine (micro-program) requires the CPU to be started.

Figure 4-41 in the FEMDM is a timing chart of the run control action during power on.

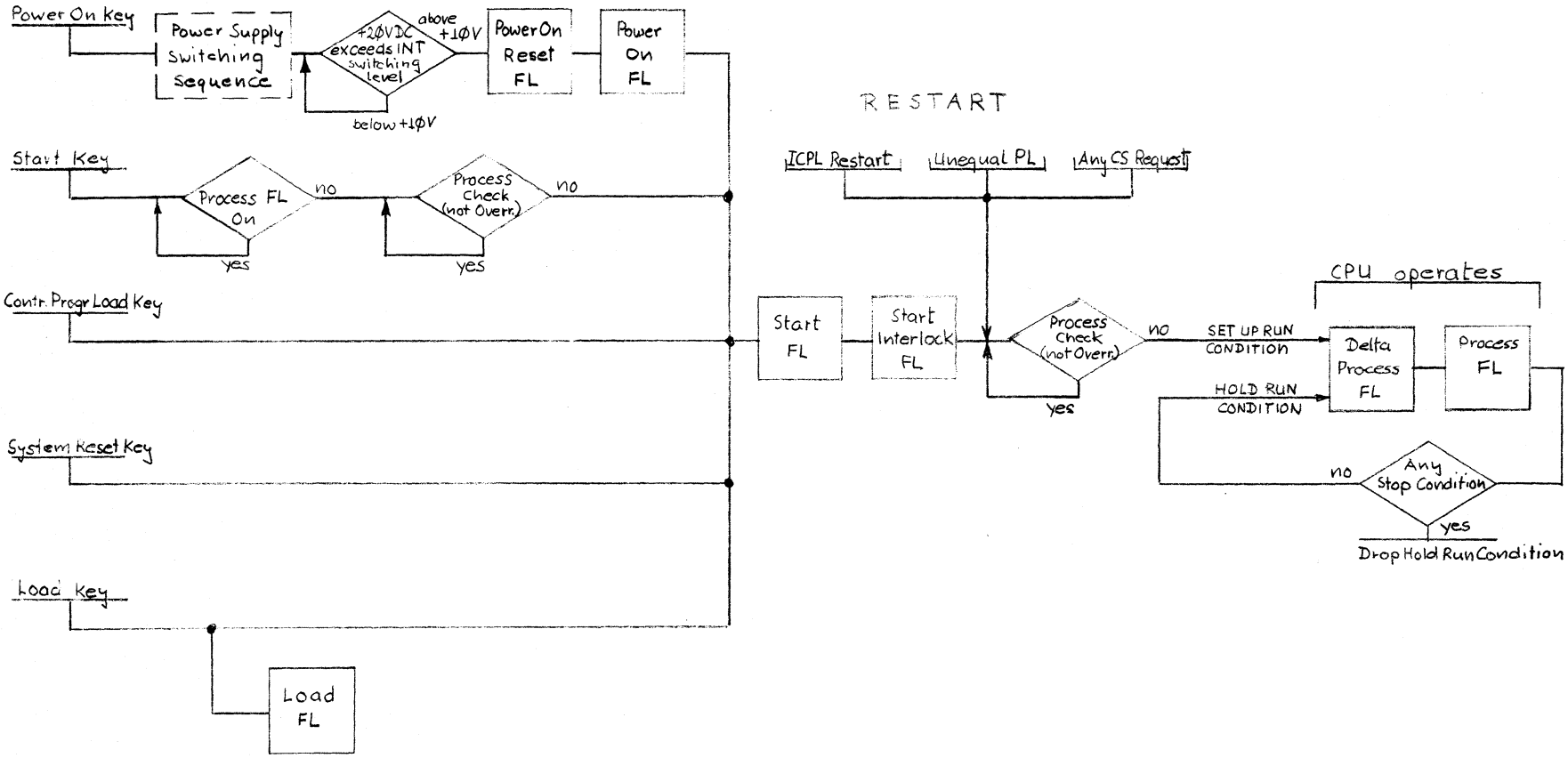


Figure 2-3 CPU Start Sequences

CPU Start by Operating Start Key

- Operating the Start key starts CPU to execute the micro-program or manual operations.

Operating the Start key on either the customer console or the remote control box initiates the action shown in Figure 4-42 in the FEMDM.

Continuously Alter or Display

- Storage or local Store Alter or Display is repeated as long as the start key is operated when the Single Cycle Switch and the Single Micro-Instruction switch are both turned off(normal).

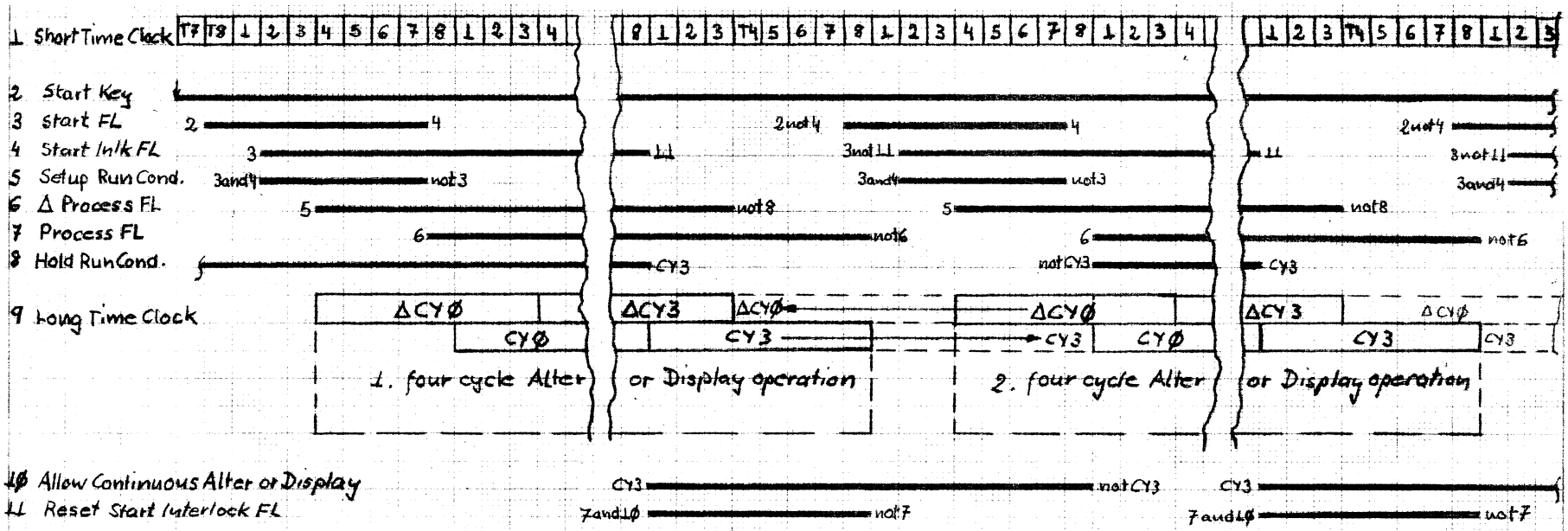
Storage or CE Local Store Alter or Display are manual operations which are performed by four cycles (cycles 0 to 3). These operations are initiated by pressing the Start Key after the mode switch on the customer's console and the CE mode switch on the CE console have been set accordingly.
The start sequence is performed as described in "CPU start by Operating Start Key."
The stop condition which deactivates the Hold Run Condition occurs at Alter or Display cycle 3 (figure 2-4). However, when the Start Key is continuously pressed, the Allow Continuous Alter or Display condition occurs, which resets the Start Interlock Latch. Thus, the start latch cannot be turned off.
The continuous operation ends when the Start key is released.

When the Single Micro-Instruction switch is on, only one Alter or Display operation is performed.
For CE Single Cycle operations the CPU stops after each of the four cycles.

CPU Start by Operating the Control Program Load Key

- Operating the CPL key initiates ICPL.
- ICPL controls reading of the control program loader card by circuits.
- After the loader card has been read the further control program is loaded under control of the micro-program.

CPU operations, except manual operations (MANOP's) , are performed under control of the micro-program. When the no micro-program is stored in the Main storage,only circuit controlled CPU operations like MANOP's can be performed. One of these circuit controlled operations is Initial Control Program Load (ICPL). The ICPL operation performs all necessary controls to read one punched card by an I/O device which is specified by set up CE Address switches.
The ICPL operation consists of four cycles (0 to 3). The cycles 2 and 3 are performed for each card column being read



Note: Allow Continuous Alter or Display (1Ø) is prevented if either the Single Cycle or the Single Micro-Instruction switch on the CE console is turned on. The operations are repeated until the Start key (2) is released.

Figure 2-4 Continuously Alter or Display Start-Stop Control

until the field length (up to 80) is decreased below zero. After the first card has been read into Main storage the CPU performs the micro-instructions read in by ICPL.

Operating the Control Program Load key (figures 2-5, 2-6) turns on the Start Latch at the next T8. At T2 the active Start Latch turns on the Start Interlock Latch. Start and Start Interlock Latch active generate the Set up Run Condition gate which allows the delta Process (T4) and Process Latch (T1) (if no Process Check exists) to turn on.
The active process latches allow the performance of delta cycle 0 and cycle 0. However, Hold Run Condition is prevented and delta Process and Process Latch drop again after cycle (delta cycle) 0.

After this CPU stop, the delta cycle latches of the Long Time Clock are advanced to delta cycle 1 while the cycle latches still contain the cycle 0 information. The cycle 0 condition remains active after the stop, until the CPU is restarted.
The cycle 0 condition allows the CPU to restart when no I/O is working (Not Any I/O Working condition). The restart signal which generates the Set up Run Condition gate is active timed by cycle 0 (until T1 after restart).

The delta Process and Process latches are turned on for the next following cycle 1 (delta cycle 1). During cycle 1 the read execute latch of the selected card read device is turned on to initiate reading of one punched card (Control Program Loader card). Since the Hold Run Condition is also prevented during cycle 1, the delta Process latch and Process latch drop again (CPU stops) at the end of cycle (delta cycle) 1. The Long Time Clock delta cycle latches are advanced to the next delta cycle (delta cycle 2) while the cycle latches still contain the last performed cycle information (cycle 1).
Restart after cycle 1 is performed when the Trap Request line 5 is activated indicating that the first card column has been read and the first character can be transferred into the Main storage. Trap Request Line 5 active and Long Time Clock cycle 1 allow activating of delta Process and Process latch.

During the following delta cycle and cycle 2 the read character in the read buffer is sensed into the CPU (FDR).
During cycle 2 the Hold Run Condition becomes active which prevents dropping of the two process latches after cycle 2. Thus, also delta cycle and cycle 3 are performed. Cycle 3 is a Store cycle during which the sensed character (cycle 2, in FDR) is stored into the Main storage position defined by the address setup in the four Address switches on the customer console.
In case of any I/O error the Hold Run Condition is inhibited during cycle 2 and the CPU stops at the end of cycle 2.
To override this error stop for trouble shooting, the Initial Load Loop switch on the CE console has to be turned on.

Since the Hold Run Condition is active only during cycle 2, the two process latches drop at the end of cycle (delta cycle) 3.
The cycle 3 condition, still present after CPU stop, allows renewed restart when the next character has been read (Trap

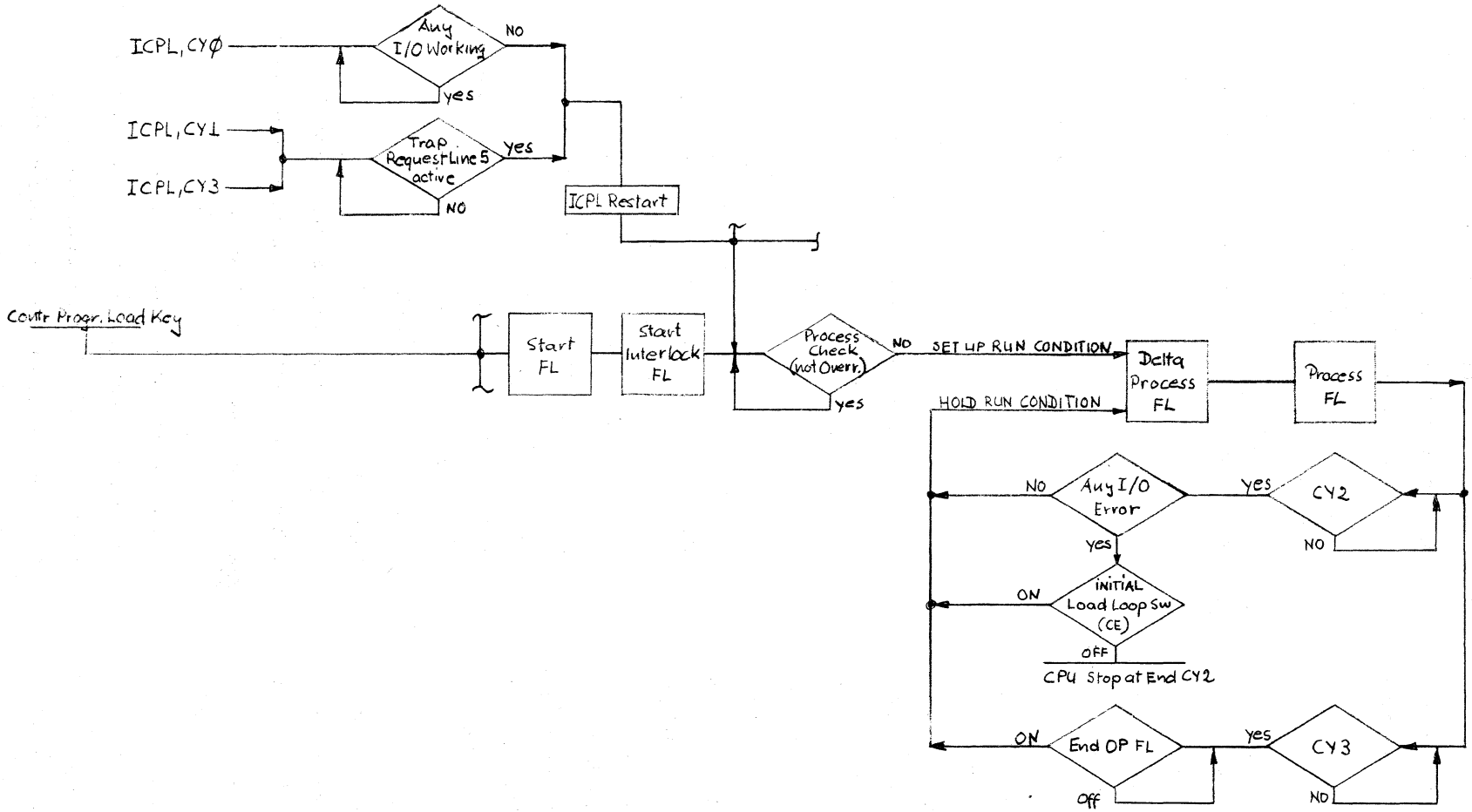


Figure 2-5 ICPL Start and Restart Sequence

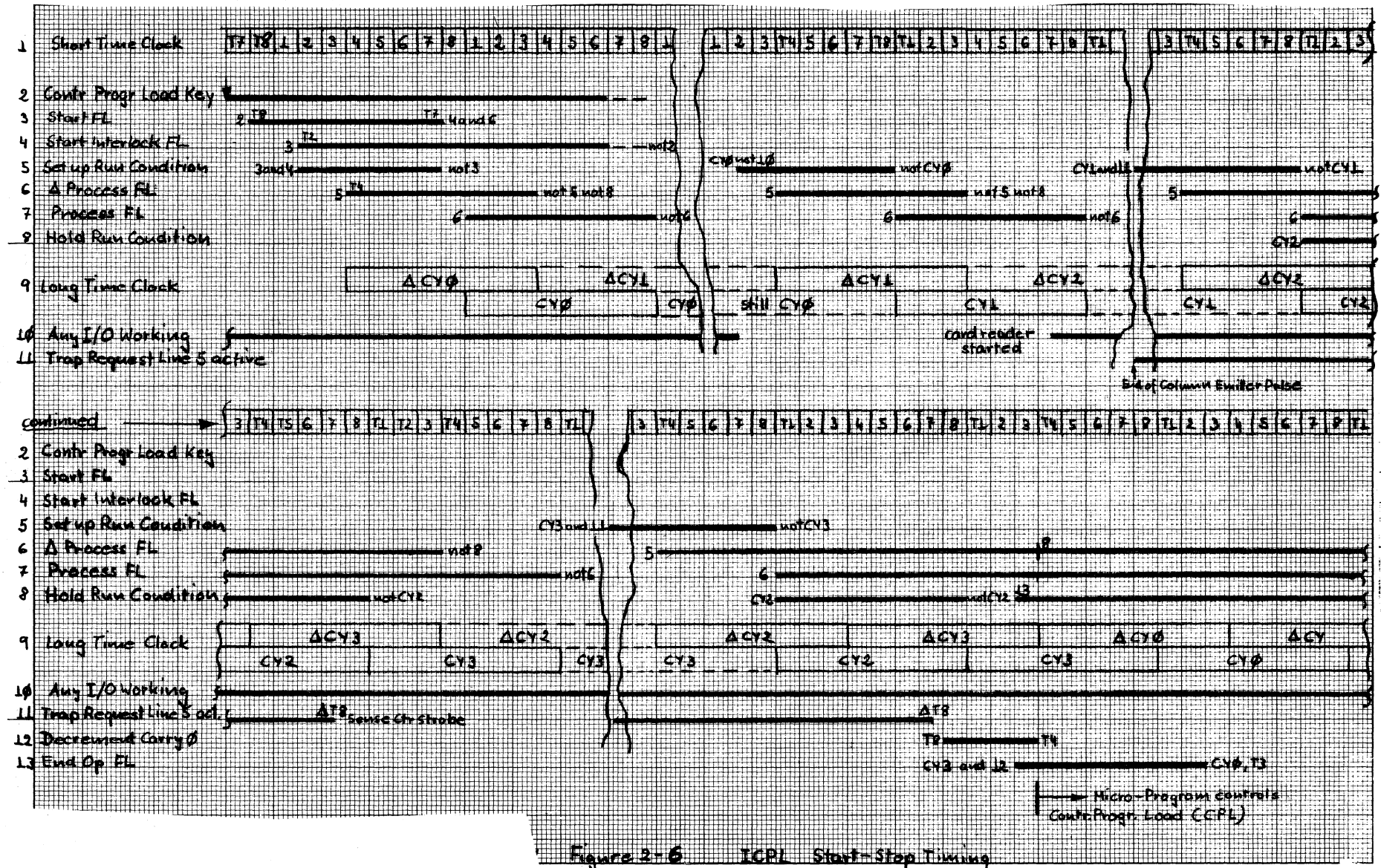


Figure 2-6. ICPL Start-Stop Timing

IBM CONFIDENTIAL

Request Line 5 active).
For each Trap Request 5, the sense and store cycles (cycles 2 and 3) are executed until the field length, updated during each character transfer, is decreased below zero (Decrement Carry 0). Then the End Op Latch is turned on (cycle 3) which activates the Hold Run Condition continuously and causes the Long Time Clock to advance to delta cycle and cycle 0
The active End Op Latch indicates that the circuit controlled ICPL operation is terminated and the CPU starts to execute the micro-program placed into Main storage.

CPU Start by Operating the System Reset Key or the Load Key

- System Reset and Load start with the System Reset Routine.

- Operating the Load key turns on the Load latch.

- For System Reset the CPU stops after the System Reset Routine while the active Load Latch forces a branch to the Load routine.

CPU start by operating the System Reset or the Load Key on the customer console is performed similar.
Operating one of the keys initiates the action shown in figure 4-45 in the FEMDM.
The active LOG latch forces the four LOG cycles (0 to 3) to be performed as soon as the delta Process and Process latch are turned on. During LOG cycle 3 the start address of the System Reset Routine /C002/ is set by circuits into IAR of the Local Store zone 7. After LOG cycle 3 the LOG latch turns off and during the following delta cycle and cycle 0 the first micro-instruction of the System Reset Routine is read out from Main Storage. At the end of the System Reset Routine the Load Latch is checked for being on by a SENSE instruction. Load latch on causes the micro-program to branch to the load routine while load latch off (System Reset) causes a CPU stop by a HALT micro-instruction.

CPU STOPS

The CPU stops when the delta Process latch and the Process latch are turned off by dropping the Hold Run Condition.
An overall picture of all possible CPU stop conditions is given in figure 2-7.
Details about stop circuits are shown in figure 4-40 in the FEMDM.

Common CPU Stops

Common Stops are those, which can occur during customer's use of the system as well as during CE operations.
The common stops are divided in normal

COMMON STOPS	NORMAL	micro-instruction HALT (OP REG Dec/D/, not Decimal)
		Storage Scan or Fill and End Op (Stop key operated)
		Storage Display or Alter and Cycle 3
		System Reset key, CPL key, or Load key operated during CPU run (Process, not Start Intlk FL)
		ICPL, not Cycle 2 and not End Op
	Check	ICPL, Cycle 2, any I/O Error, when not Initial Load Loop Sw. or not Process Check Override Sw. on.
	Check	Process Check, not Process Check Override Sw. on,
	Check	Storage Protect (power supply failure) active.
CE STOPS	NORMAL	Single Cycle Sw. on
		Single Micro-instr. Sw. on and Allow PL Switching (indicates end of instruction or end of ALC repetition).
		Compare Equal Stop Sw. on and selected Address Bits Ø-15 equal.
		Storage Test, End Op run 2 and Stop key FL on (Stop Key operated).
	Check	Storage Test, Invert Parity Sw. on, and no Process Check and not Process Check Override Sw. on

Figure 2-7 CPU Stop Conditions

and check stops.

Normal CPU Stops

CPU Stop by Micro-Program: The micro-instruction HALT (Halt and Display) allows the CPU to be stopped by micro-program. Operating the Stop key, programming errors and I/O errors, which can be detected by the micro-program may result in a micro-instruction Halt.

CPU Stop for Storage Scan or Fill: Storage Scan or Fill are MANOP's which are continuously running until the Stop key is operated. Operating the Stop key turns on the End Op Latch in the last cycle (cycle 3) of the current Scan or Fill repetition. The Mode switch in position Storage Scan or Storage Fill and the End Op FL active (cycle 3) drops the Hold Run Condition.

CPU Stop for Storage Display or Alter: Storage Display or Alter are MANOP's which require four cycles (0 to 3) to perform their function. Since the display or alter function is terminated after the fourth cycle (cycle 3) the Hold Run Condition drops when this last cycle occurs.
Allow Continuous Alter or Display (no Single Cycle or Single Micro-Instruction) does not influence this stopping. It allows only renewed Alter or Display operations to be initiated after the CPU has been stopped for eight T-pulses (one cycle) as long as the Start key is pressed.

CPU Stop if System Reset Key, Control Program Load Key or Load Key is operated: Operating the System Reset, Control Program Load (CPL) or the Load Key drops the Hold Run Condition when the CPU is running (Process, not Start Interlock Latch). The CPU stops after the current cycle. This stop is necessary to allow the CPU start sequence. Especially the Start Latch can only turn on when the delta process latch is off. Turning on the Start latch activates the necessary reset signals. When the delta Process and Process Latch are turned on processing starts either with LOG-System Reset Routine (System Reset Key, Load Key) or with ICPL (Control Program ~~Progr~~. Load Key).

CPU Stops during Initial Control Program Load (ICPL):
During ICPL the CPU stops by dropping the Hold Run Condition after cycle 0, cycle 1 and cycle 3. See figure 2-6.

CPU Check Stops

CPU Stop by Process Check: The Process Check latch turned on
by any CPU check condition drops the Hold Run Condition. The Process Check Override switch (CE console) turned on prevents dropping of the Hold Run Condition.

CPU Stop by Power Supply Failures: Any power supply error causes the Storage Protect signal which immediately drops the

Hold Run Condition and the CPU stops at the end of the current cycle.

CPU Stop by Any I/O Error during ICPL:
During reading of the control program loader card, the bus Any I/O Error active prevents Hold Run Condition during cycle 2 and the CPU stops at the end of cycle 2. The bus Any I/O Error can be activated by any check condition (Read Check, Feed Check) in the selected card read device.
Turning on the Initial Load Loop switch on the CE console allows continuous ICPL operations without CPU stop in case of failing I/O operations . This Initial Load Loop is used for trouble shooting when ICPL fails to operate.

Turning on the Process Check Override switch (CE console) also prevents stopping in case of I/O error, however, there is no assurance that the loaded information is correct.

CPU Stops for CE Operations

To terminate CE operations the Hold Run Condition is dropped by a normal stop or a check stop condition.

Normal CE CPU Stops.

Single Cycle Stop : The Single Cycle switch on prevents Hold Run Condition. The active Set up Run Condition gate turns on the delta Process and at T4 the Process latch at T1. Since the Set up Run Condition gate drops already at T8 (Start latch off) and no Hold Run Condition exists both latches drop again (CPU stops) eight T-pulses (one delta cycle or cycle) after they have been turned on.

CPU Stop for Single Micro-Instruction Operations : Micro-Instructions can differ in the number of cycles necessary to execute the required function. To stop the CPU at the end of a micro-instruction, the Hold Run Condition must be prevented during the last instruction cycle. The last cycle of a micro-instruction or of an ALC repetition is indicated by the active Allow PL Switching signal. Thus, Single Micro-Instruction switch turned on and Allow PL Switching signal active drop the Hold Run Condition.
Micro-instruction with automatic length count (ALC) stop after each repetition of the required function cycles and the micro-instruction is not terminated before the length count is decreased below zero.

Compare Equal Stop: A Compare Equal Stop is performed if the corresponding switch in the CE console is turned on and the bit pattern provided by the source selected by the CE Display/Compare Select switch is identical to that set up in the four CE Select Switches.

When both bit patterns are equal the Address Equal bits 0 to 15 signal occurs which drops the Hold Run Condition.

CPU Stop to Terminate Storage Test :

Storage Test operation is performed when the CE Storage Test Switch is turned on and the Start key is operated. The Storage Test performs 4 different runs (from address /0000/ to /FFFE/). During run 1 and 2, operations are performed with correct (odd) parity. The test is continuously running and the CPU stops either by Process Check (CE Check Stop) or after the Stop Key has been operated. However, to provide correct parity in the Main Storage, the Hold Run Condition drops at the end of run 2 (End Op Latch on) after the Stop Latch has been turned on by operating the Stop key.

CE Check Stop

- The only CE check stop occurs during Storage Test when the Invert Parity switch (CE) is turned on and not all CPU checks are activated.

The only CE Check Stop occurs when the Storage Test runs with the Invert Parity switch (CE console) turned on and no Process Check is up. The Invert Parity switch on causes incorrect parity during Storage Test and the following CPU check latches have to be turned on :

SAR Check
Inhibit Check
ALU Check
Modifier Check
Shift Unit Check

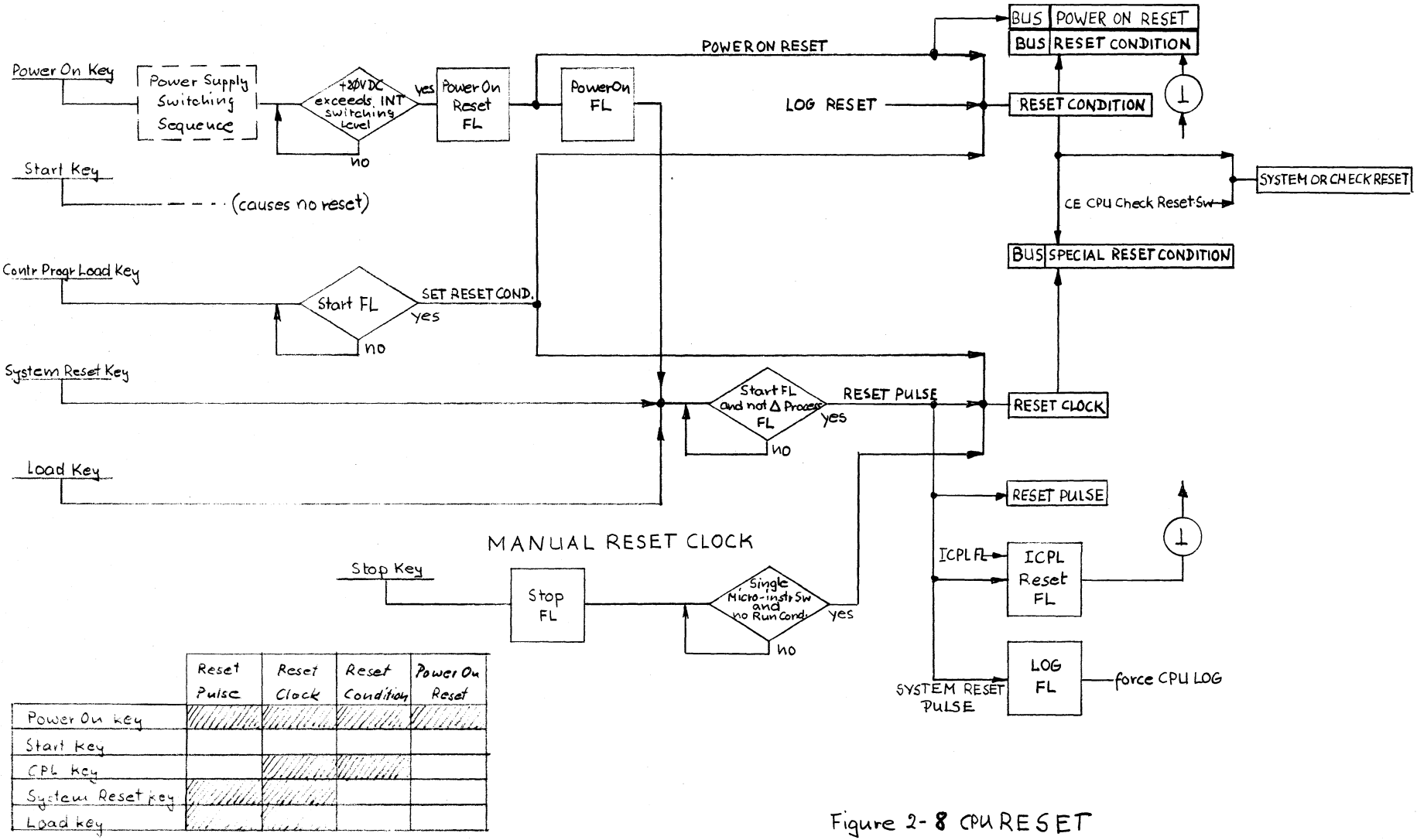
These five checks active generate the signal All Checks which turns on the Process Check. When one of the checks fails to turn on, the Process Check remains off and the CPU is stopped by dropping the Hold Run Condition.

CPU RESET

Operating the Power On key, the Control Program Load key, the System Reset key or the Load key forces the CPU into a basic situation by several reset signals.
These reset signals are:

Reset Pulse (System Reset Pulse)
Reset Clock
Reset Condition
Power On Reset

The reset signals generated depending upon which key has been operated are shown in figure 2-8.
The reset signals are used within the CPU circuits and are sent out to the attachments via bus lines.



	Reset Pulse	Reset Clock	Reset Condition	Power On Reset
Power On key				
Start key				
CPL key				
System Reset key				
Load key				

Figure 2-8 CPU RESET

CORE STORAGE

- The core storage unit, 4K or 8K, is selfcontained on a single SLT board.
- There are two core storage units available : 4usec and 2usec.
- MDM page SD011 is a block diagram of the 4usec core storage.
- MDM page SA011 is a block diagram of the 2usec core storage.

MAGNETIC CORE THEORY

A magnetic core is a small doughnut-shaped ring that is uniformly constructed of ferrite particles bonded together by a ceramic material. The ferrite particles have good magnetic properties and the core has a high retentivity of the magnetic flux lines after the magnetizing force is removed. It is this property of retentivity that makes a magnetic core useful as a storage device.

The operation of a magnetic core can best be described by reference to its hysteresis curve, Figure 2-9. This curve is a plot of the relationship between a magnetizing current and the flux density of the core.

A magnetic core is capable of maintaining indefinitely one of two stable magnetic states, either at point A or at point D on the hysteresis curve. Because the core has two stable states, it can be used as a binary storage device. At point A the core has a residual flux in a negative direction, and at point D a residual flux in the positive direction. These two directions can be arbitrarily assigned as binary 'zero' and binary 'one', respectively.

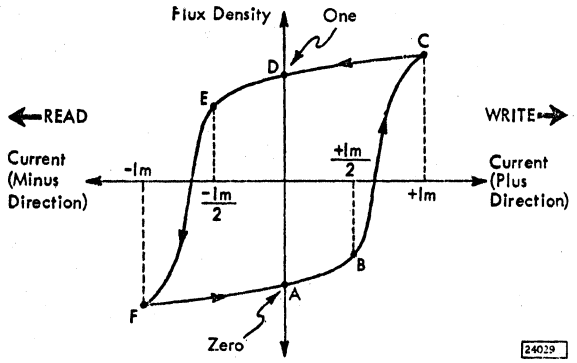


Figure 2-9. Hysteresis Curve of Magnetic Core

I_m is the amount of current necessary to change the state of the core. Plus I_m is the amount of current required to 'flip' the core from binary zero to binary one. Minus I_m is the same amount of current in the opposite direction required to flip the core from binary one to binary zero.

On the hysteresis curve, it can be observed that a magnetizing current of plus I_m will change the magnetism of the core from point A, a binary zero, to a value in the positive direction at point C. When the current is removed, the total amount of magnetization drops back to point D (binary one). If, instead of the full magnetizing current I_m , a current of $I_m/2$ were applied, the flux would change only the small amount from point A to point B on the curve, and when the current returned to zero, the flux would return to its original value.

A reverse current, minus I_m , develops flux of opposite polarity and , if the core is in the 'one' state, changes the magnetic state

of core from point D to point F. When the driving current is removed, the magnetization drops back to point A (binary zero).

Writing Into Core

The magnetic properties of the core make it ideally suited for use in a storage matrix employing X and Y drive lines. Each core of the matrix is threaded by three windings (Figure 2-10). One winding is an X drive line that carries a current $I_m/2$ and one winding is a Y drive line that carries a current $I_m/2$ in the same direction. A coincidence of current in these two windings occurs at one core storage position (18 cores) thereby 'selecting' that word.

The third winding is the inhibit/sense winding. When writing into a core storage position, each core that is not to receive a 'one' bit is inhibited by a current $I_m/2$ flowing in its inhibit/sense winding in a direction opposite that flowing in the X and Y drive lines. The magnetic field produced by the inhibit/sense winding effectively cancels half the field produced by the X and Y drive lines. The resulting magnetic field is insufficient to flip the core to the 'one' state.

Reading Out of Core

When information is to be read out of a core storage position, the currents $I_m/2$ in the X and Y windings are reversed (Figure 2-10). Each core that flips from a 'one' to a 'zero' state induces a pulse into its inhibit/sense winding. The inhibit/sense winding at each core does not carry inhibit current and is used instead to detect the change in state of the magnetic flux. The sensed bits are set into the corresponding positions of the B register.

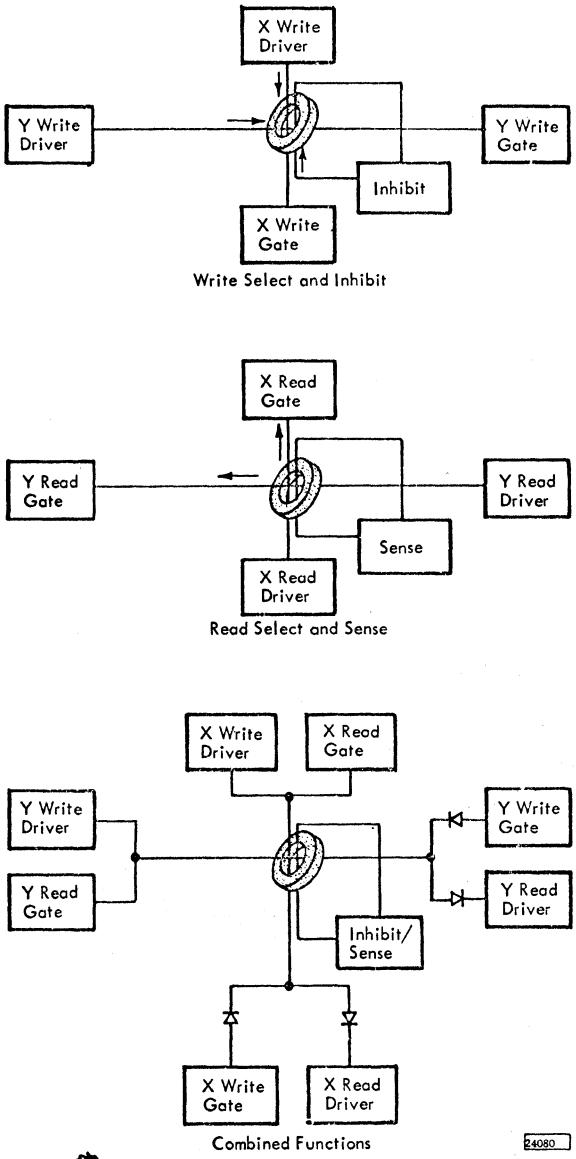


Figure 2-10 Core Storage Write and Read (4-usec Storage)

ADDRESSING

- One X drive line and one Y drive line are activated by selecting both ends of the lines.

- Each active drive line carries half-select current.

- The two drive lines intersect at one core in each bit plane (18 cores) to provide full select current at that one halfword.

Eight-K Addressing

The magnetic cores are arranged in matrices of 128 x 64 cores, called planes. (Each plane is actually 128 x 68, providing 512 positions of auxiliary storage that are not program addressable and are described in another section.) The 8K array consists of 18 of these planes with each plane assigned to one bit position of a core storage halfword. Corresponding core positions in each plane are addressed simultaneously by the X and Y drive lines to select one 18-bit halfword. Since there are 8192 cores in each plane, the 8K array has a capacity of 8192 halfwords.

The 128 x 64 matrix in MDM page SD041 represents one plane of the four-usec core storage, 8K array. Eight-K addressing for the two-usec core storage is shown in MDM page SA 041. It differs from the four-usec core storage only in the drive circuits which are described in another section. The one core that is shown is selected by the address represented by Address Register bits 3 to 15.

At side A, the 64 X drive lines, which run through all planes, are divided into eight groups of eight lines each. One group is activated by a decode of Address Register bits 3, 4, and 5. At the other end of the array, one line in each group of eight is activated by a decode of Address Register bits 6, 7, and 8. Thus, a single X line is activated and carries half-select current.

At side B, the 128 Y lines which run through all planes, are divided into eight groups of 16 lines each. One group of 16 lines is activated by a decode of Address Register bits 9, 10, and 11. At side D, actually the other end of the array, one line in each group of 16 is activated by a decode of Address Register bits 12, 13, 14 and 15. A single Y line is thereby activated and carries half-select current.

In each plane, the core at the intersection of the active X and Y drive lines receives full-current and is the only core in the plane that is selected.

Four -K Addressing

One plane of the four-usec core storage 4K array is shown in MDM page SD042. Four-K addressing for the two-usec core storage is shown in MDM page SA042. The 4K and 8K arrays occupy the same area of the SLT board. The core planes used in the 4K array are identical to those used in the 8K array but only nine planes are used. For addressing purposes, each 128 x 64 plane is divided into two 128 x 32 half-planes. Each of the 18 half-planes is assigned to one bit position of the core storage word.

Since there are 4,096 cores in each halfplane, the 4K array has a capacity of 4,096 halfwords. (An auxiliary storage of 256 halfwords is actually part of the 4K array ; it is described in another section.) The 32 X drive lines are divided into four groups of

eight lines each (one-half as many groups as in the 8K array). The X lines run through the nine B halfplanes, then loop back through the nine D halfplanes. One group is activated by a decode of Address Register bits 4 and 5 (bit 3 is not used with 4K core storage). At the other end of the X lines, one line in each group is activated by a decode of Address Register bits 6, 7, and 8. A single X line is thereby activated and carries half-select current. The 128 Y drive lines are divided into eight groups of 16 lines each, as in the 8K array, and run through all nine planes. One group of 16 lines is activated by a decode of Address Register bits 9, 10, and 11. At the other end of the array, one line in each group of 16 is activated by a decode of Address Register bits 12, 13, 14, and 15. This active Y line intersects the active X line at two cores in each plane; therefore, nine planes provide 18 bits for each of 4096 halfwords.

MODULE SELECTION

- One module (one SLT board) of core storage can contain a 4096 - or 8192-core array.
- Address register bit 1 and bit 2 lines select the module.
- Module selection lines are connected through wired-logic (Figure 2-11).

The circuits that activate the core storage clock and timing circuit, and therefore select the unit, require a specific combination of address-register-bit-1 and - 2 lines. These lines connect from the SAR Power output to the core storage circuits through wired logic. Figure 2-12 shows the combinations required to activate each module.

ADDRESSING - AUXILIARY STORAGE

- Within 2020 CPU's with a serial number 50,000 and above the auxiliary storage areas are part of the control storage.

Auxiliary storage is selected by the auxiliary read-driver and auxiliary-read-gate circuits. These two circuits are activated by the address-register-bit-aux. line. This line is activated by SAR Power depending upon SAR bit 0 and 1 active.

CORE STORAGE ARRAYS

- Core storage arrays are available in two sizes, 4096 (4K) halfwords and 8192 (8K) halfwords.
- Each halfword consists of 18 bits.
- Both arrays consist of core planes in a 168 x 68 matrix (168 x 64 program-addressable).
- The 8K array contains 18 physical planes, each of which contains one bit-position for each halfword.

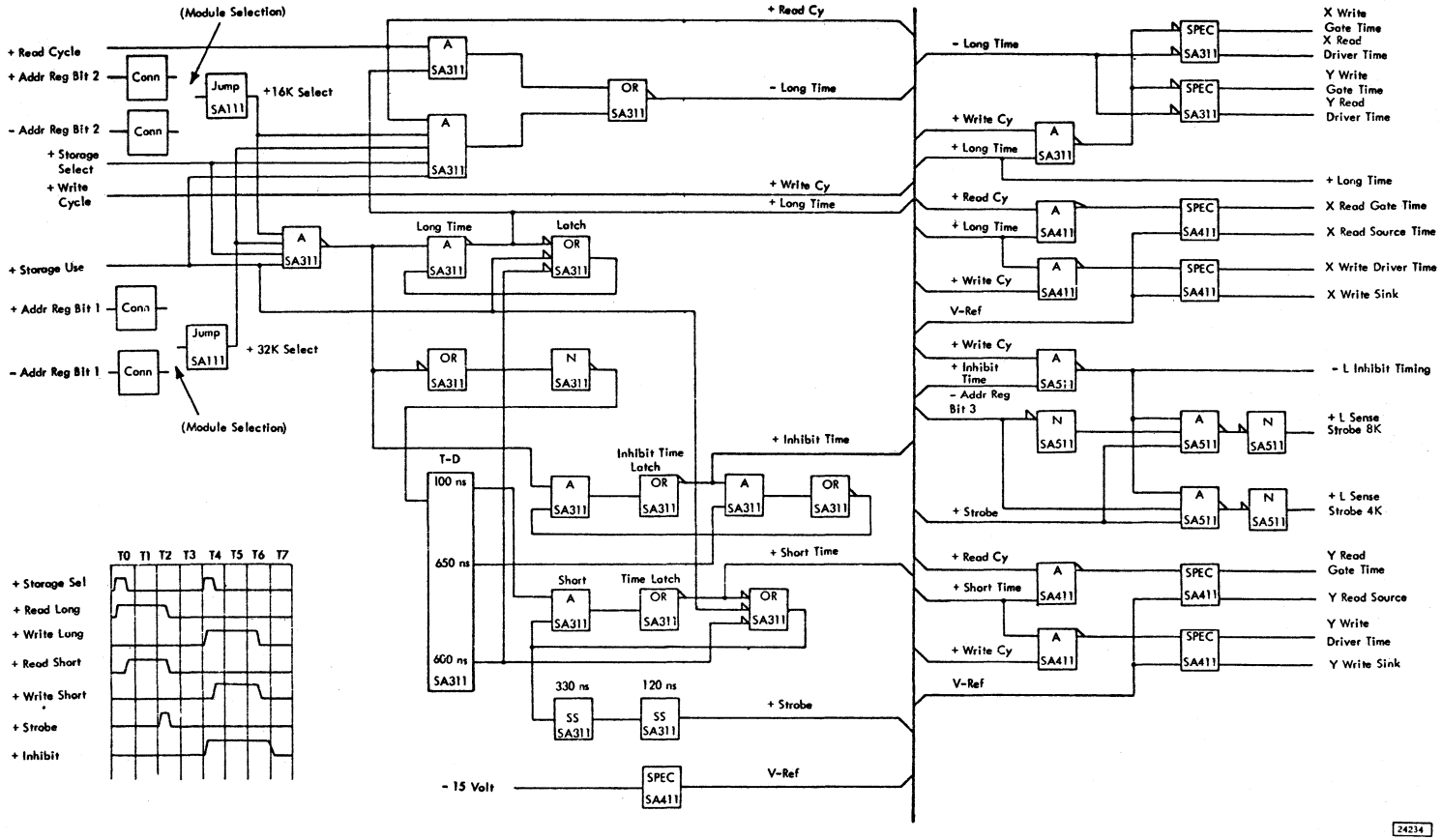


Figure 2-8. Two-Microsecond Core Storage Clock

System Storage Capacity	Module Wiring			
	8K	16K	24K	32K
8K	No Connections			
16K	- Bit 2	+ Bit 2	No Connections	
32K	- Bit 2 - Bit 1	+ Bit 2 - Bit 1	- Bit 2 + Bit 1	+ Bit 2 + Bit 1

24237

Figure 2-4. Core Storage Module Selection

The 4K array contains 9 physical planes, each of which contains two bit-positions for each halfword.

Two usec and Four usec Core Storage Arrays. Because the two-usec and four-usec core storage units use different address drive circuits and different inhibit/sense circuits, the use of the array terminals is different. The 8K and 4K array descriptions given here are based on the four-usec storage. It applies to the two-usec storage arrays except for references to the drive circuits.

The differences can be seen by comparing bottom-board and diode-board diagrams for the two units. The MDM page numbers for the four-usec storage begin with SD ; the corresponding diagrams for the two-usec storage have the same page number except they begin with SA. The inhibit/sense terminals for the two units are shown in SA/SD 061.

Eight-K Array

The 8K core storage array consists of a bottom board, 18 core planes, and a diode board. The bottom board plugs into the SLT board and provides connection from the SLT circuits to the X, Y, and inhibit/sense lines of the array. MDM page SD 012 shows the four-usec core storage SLT board with an 8K array disconnected and indicates the path of typical address and inhibit/sense lines.

MDM page SD 071 shows the land pattern for the bottom-board terminals; the connectors are not shown. The even-addressed Y write driver and read gate lines go from the connectors, through the land pattern, to terminals B4-35 and B46-77 ; the odd-addressed lines go to terminals D4-35 and D46-77. The even-addressed X write driver and read gate lines go to terminals C2-17 and C20-35; the odd-addressed lines to terminals A2-17 and A20-35. The write driver/read gate windings of Bit 17 plane connect directly to the bottom-board terminals.

The X and Y write gate and read driver lines go from the connectors, through the bottom-board land-pattern, to the following terminals: even-addressed lines to B1-3, B36-45, and B78-80; odd-addressed lines to D1-3, D36-45 ; and D78-80. The lines then go along the sides of the array to the corresponding terminals of the diode board. The diode-board landpattern (SD 081) connects the lines to the common terminals of the diode packs. From the individual diodes, the land pattern

connects to the terminals at the sides of the diode board.

Diode Board Terminals : Refer to MDM page SD 081 and note that the storage address register is divided into X and Y High Order and Low Order positions. If the X portion of the specified address is even (pos. 8 = 0), the active X terminal is on side C of the array. If the X portion of the address is odd (pos. 8 = 1), the active X terminal is on side A of the array.

If the Y portion of the address is even (pos. 15 = 0), the active Y terminal is on side B of the array ; if odd (pos. 15 = 1), the active Y terminal is on side D.

Locate the specified terminals by finding the group of lines associated with the X and Y high order positions of the address. The relative position of the line within the group is the same for all low order values and is identified in the enlarged drawing (within broken lines) for each side of the diode board.

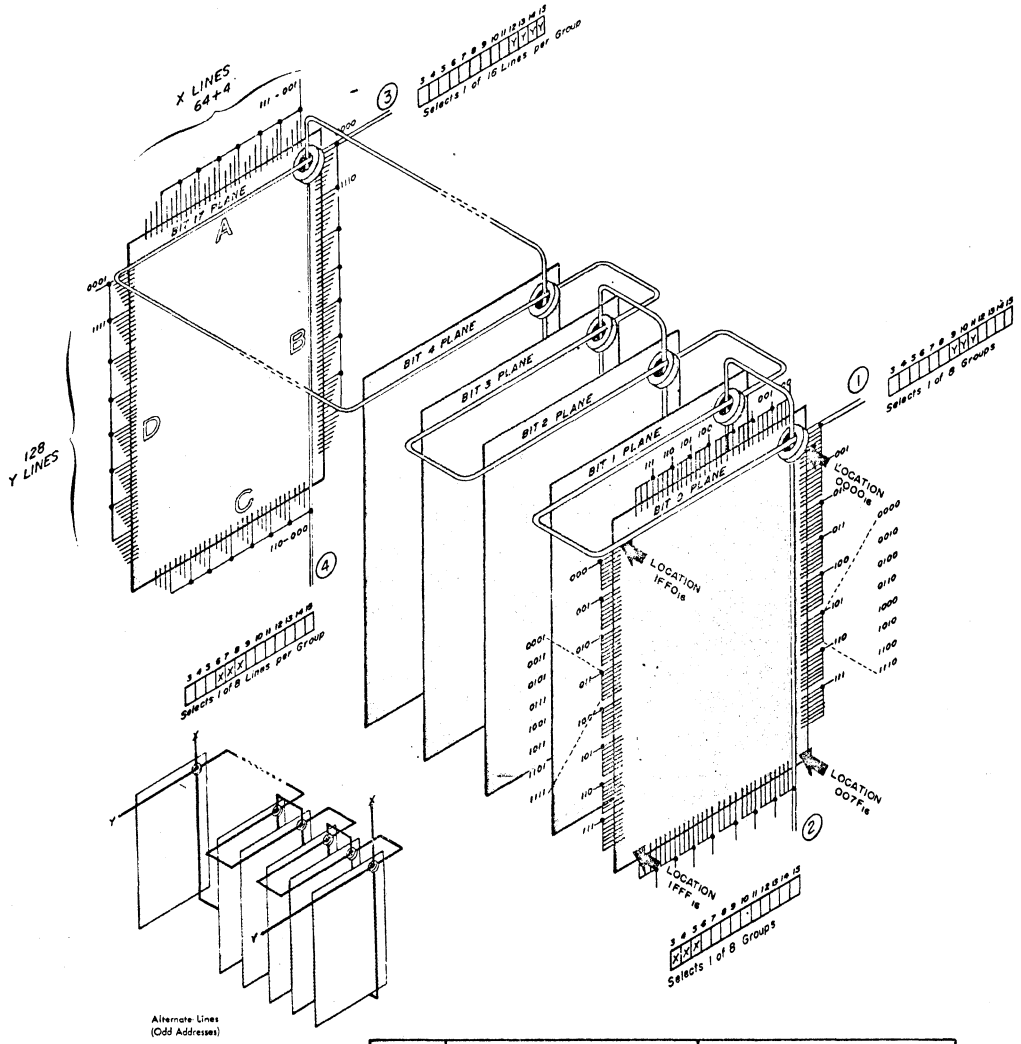
Array Windings : The diode-board terminals can be related to the terminals on each plane and the lines can be traced through the array by referring to Figure 2-12. This figure shows the relative location of the 18 cores that form the halfword at core storage address 0000_{16}. Note that the Y address line for this address enters Bit 0 plane at side B and is activated by Address Register bits 9, 10, and 11 (Y high order).

The X line enters the Bit 0 plane at side C and is activated by Address Register bits 3, 4, and 5. The two lines go through every plane, intersecting at the core at location 0000_{16} in each plane.

At bit 17 plane, the Y line exits at side B and is activated by Address Register bits 12-15 (Y low order). The X line exits at side C and is activated by Address Register bits 6, 7, and 8 (X low order).
The X and Y lines for an odd address enter Bit 0 plane and exit Bit 17 plane at the sides of the array opposite those for an even address, as shown in Figure 2-13 inset. The inhibit/sense lines for an 8K array are shown in Figure 2-14. The lines for plane 1 (bit 0) connect to bottom board terminals at side C ; alternate planes connect to sides A and C. The bottom board connects the lines to the inhibit/sense cards through the land patterns and connector blocks (MDM SD 012).

Bottom Board Terminals : From the terminals of Bit 17 plane, the lines go to the bottom board terminals (MDM SD 071), through the land pattern to the connectors(not shown) and through the SLT board land-pattern to the drivers and gates.

The bottom board also provides connection from the inhibit/sense connectors (not shown) that plug into the SLT board, to terminals 1S-54S at sides A and C.



Note	Four μsec Storage	Two μsec Storage
①	Y Read Driver/Write Gate	Y Read Gate/Write Driver
②	X Read Driver/Write Gate	X Read Gate/Write Driver
③	Y Read Gate/Write Driver	Y Read Driver/Write Gate
④	X Read Gate/Write Driver	X Read Driver/Write Gate

Figure 2-14. X and Y Address Lines for Location 0000, 8-K Array

Four - K Array

The 4-K array consists of a bottom board, 9 core planes, and a diode board. As described in the addressing section, the core planes used in the 4K and 8K arrays are the same. The Y windings in the two arrays are connected in the same way but the X windings are connected differently. Because each physical plane must provide two bit-planes, the X windings go through the B-half of each plane in the array and then through the D-half. MDM page SD 072 shows the land pattern of the 4K bottom board.

Note that at sides A and C the four groups of X lines in the B-half are common to the four groups in the D-half.

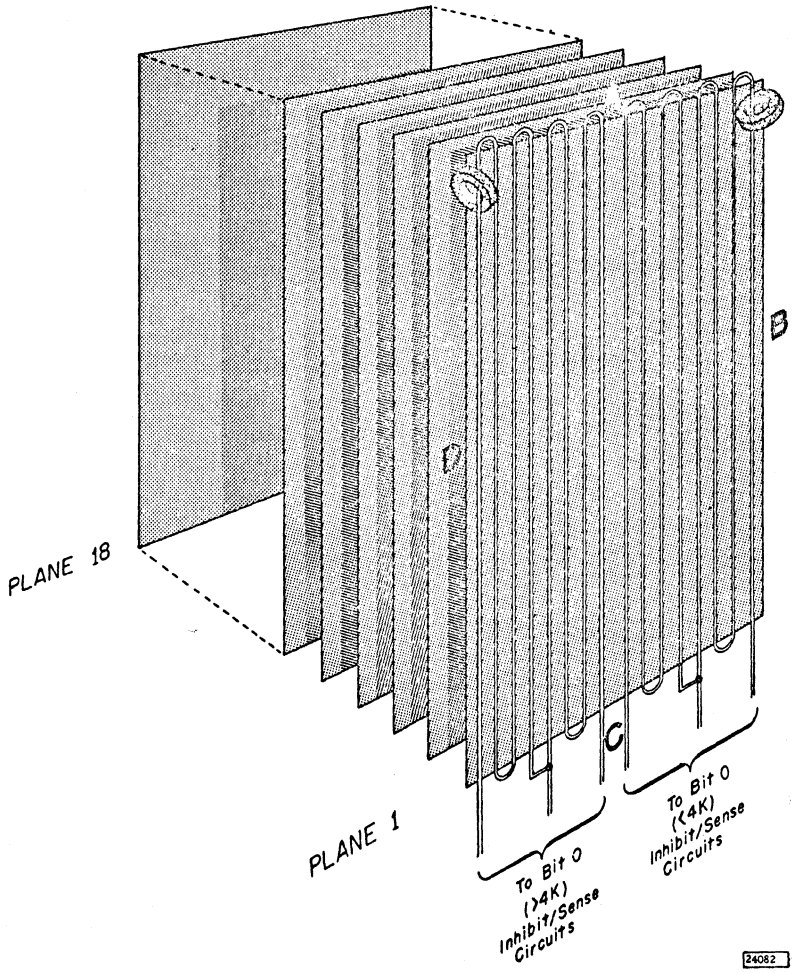


Figure 2-14. Inhibit/Sense Lines for Bit 0, 8-K Array

The Bit 16 and 17 X lines shown in figure 2-12 connect to the bottom-board terminals at side A and C. Side A of the bottom-board land-pattern thus provides the 'jumpers' for the even-addressed X lines between Bit 16 and 17 and side C provides the 'jumpers' for the odd-addressed X lines. The MDM shows the land pattern for the terminals ; the connectors are not shown.

The X and Y write driver and read gate lines go from the connectors, through the bottom board land pattern, to terminals at the sides of the array. The write driver/read gate windings of Bit 16 and 17 planes connect directly to the bottom board terminals.

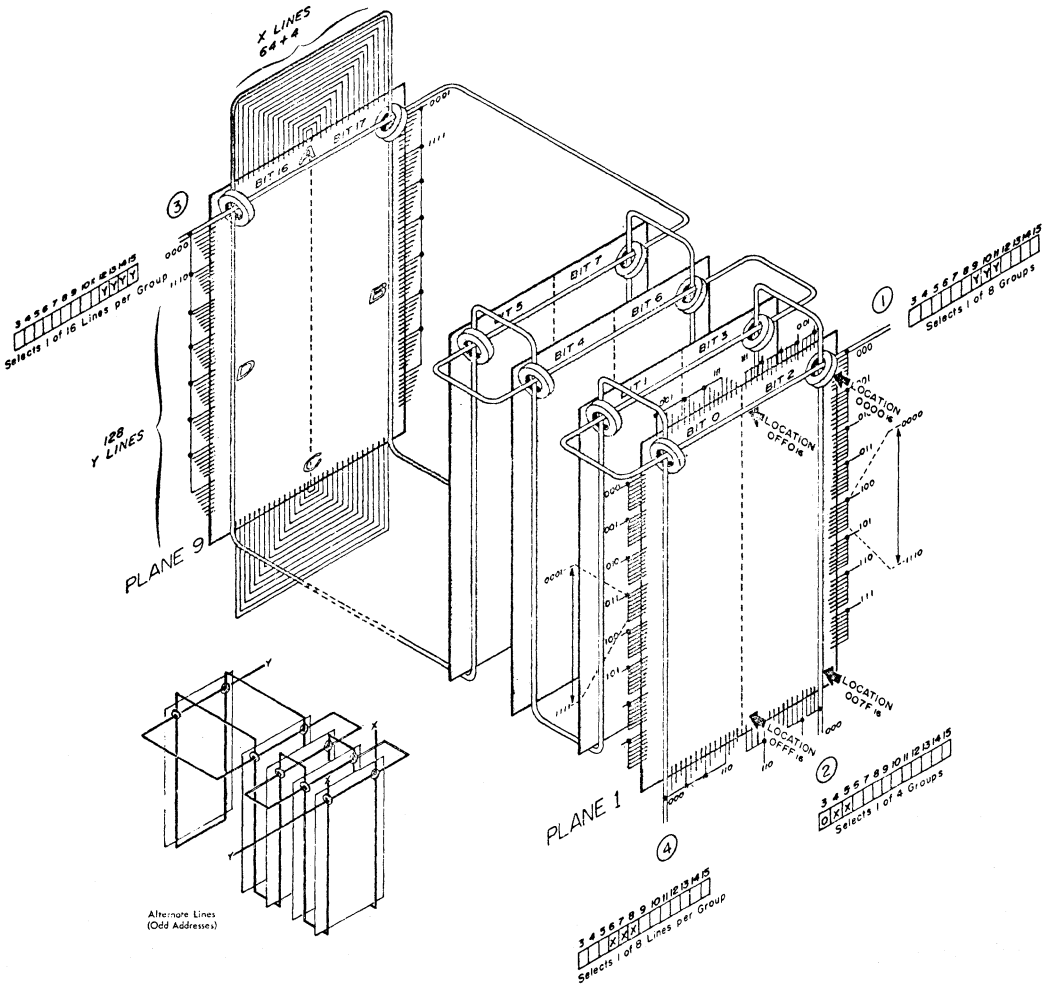
The X and Y write gate and read driver lines go from the connectors , through the bottom-board land-pattern to the indicated terminals, and along the sides of the array to the corresponding terminals of the diode board. The diode-board land-pattern (MDM 082) connects the lines to the common terminals of the diode packs. From the individual diodes, the land pattern connects to the terminals at the sides of the diode board.

Diode Board Terminals : The function of the 4K diode board (MDM SD 082) is the same as the previously described 8K diode

board. One-half of the X read driver/write gate diode-packs are not used because there are half as many X drive lines in the array. The terminals are numbered differently and sides A and C are used differently.

The B-half of sides A and C provide connection from the diodes to the X read driver/write gate lines in groups of eight lines as do the corresponding terminals of the 8K diode board. The D-half of sides A and C provide connection from the other end of the lines to the X read gate/write driver lines within the groups.

Array Windings : Figure 2-15 illustrates the 4K array bit-plane layout, the paths of the X and Y windings, and the relation of the windings to the terminals of the bottom board and diode board. The windings within the planes are the same as those in the planes of the 8K array ; it is the bottom board and diode board that cause the array



Note	Four μsec Storage	Two μsec Storage
①	Y Read Driver/Write Gate	Y Read Gate/Write Driver
②	X Read Driver/Write Gate	X Read Gate/Write Driver
③	Y Read Gate/Write Driver	Y Read Driver/Write Gate
④	X Read Gate/Write Driver	X Read Driver/Write Gate

Figure 2-15. X and Y Address Lines for Location 0000, 4-K Array

to function as 4096 eighteen-bit halfwords.

The even-addressed Y lines, from the read driver/write gate circuits, enter plane 1 through the Bit-2-plane half (side B), go through every plane, and exit plane 9 at the Bit-16 plane half (side D). Odd-addressed Y lines (Figure 2-15 inset) enter plane 1 at side D and exit plane 9 of side B. Even-addressed X lines, in four groups of eight each, enter plane 1 at the B-half of side C, go through the B-half of every plane, and loop back (by the bottom-board land-pattern) to the D-half of plane 9. The lines then go through the D-half of every plane and exit plane 1 at side C to connect to the diode board. The lines are connected, through the diode board and the bottom board, to the X read gate and write driver circuits.

Odd-addressed X lines (Figure 2-15 inset) enter plane 1 at side A, go through every plane, loop back (by side C of the bottom board), and exit plane 1 at side A.

The inhibit/sense lines for a 4K array are shown in Figure 2-16. The lines for plane 1 (bit 0 and 2) connect to bottom board terminals at side C; alternate planes connect to sides A and C. The bottom board connects the lines to the inhibit/sense cards through the land patterns and connector blocks (MDM SD 012).

Drive Current Generation

Four - usec Storage

- Gate and driver circuits direct the drive current through the array.
- The current control circuit provides a constant current (current sink) for the array.
- A temperature compensated voltage reference is applied to the current control circuit.

The direction in which the half-select current flows in the X and Y address drive-lines is controlled by the driver and gate circuits. These circuits are conditioned by timing, for read and write positions of the cycles, and by the address lines. The block diagram of the four-usec core storage (MDM SD 011) shows the read and write drivers and gates controlled by the Address Register bit lines. Timing control is implemented through the current control block at the top of the diagram.

MDM page SD 042 shows more detail of the drive line circuits. Looking at the one X drive line shown, note that address lines 3, 4, and 5 condition eight decode circuits. One decode circuit (decode 000 is shown) conditions both an X write gate circuit and an X read driver circuit. During read time, X read driver control will further condition

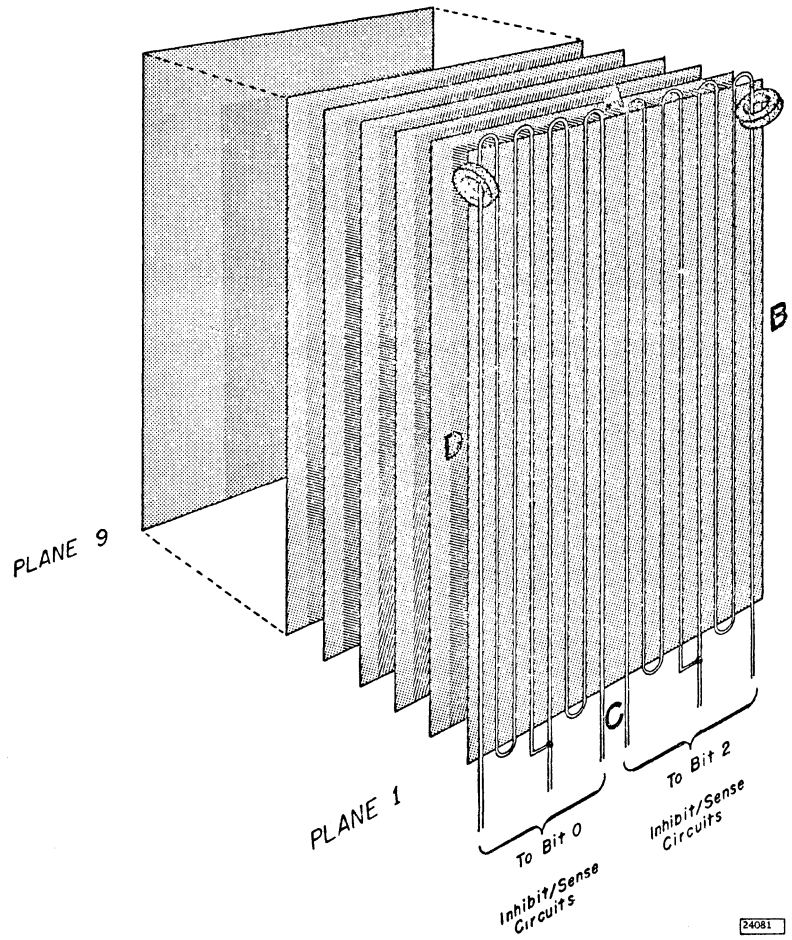


Figure 2-16 Inhibit/Sense Lines for Bit 0 and 2, 4-K Array

the X read driver to allow current to flow through the read driver to the X read sink. (Conventional current flow is indicated in the core storage MDMs.) This X read driver activates a group of eight X lines.

One of the eight active lines is also activated at the other end of the array by an X read gate circuit. The read gate is conditioned by a decode of address lines 6, 7, and 8 (decode 111 is shown). The read gate is further conditioned by X read gate control and current is allowed to flow from the 8.3 volt source through the X drive line. This read gate activates one line in each group of eight but only one line is activated at both ends.

During write time, the same address decodes are still conditioned (the address lines do not change between read and write time). The X read driver control is no longer active; the X write gate control becomes active and conditions the X write gate to activate the same group of eight lines. At the other end of the array, the X write driver is conditioned by the same address decode and by X write driver control to allow current to flow to the X write sink. Half-select current now flows in the same line as during read time but in the opposite direction.

MDM page SD 043 shows still more detail of the driver and gate circuits. In this diagram, read current flow (conventional current flow direction, pos. to neg.) is shown by the heavy lines. The controlling circuits, identified by broken lines, must be conducting to activate the read gate and read driver circuits.

When the read or write timing pulses are not conditioning the control circuits (quiescent state), current flow is in the driver control and gate control circuits only. The center transistors of these two circuits are forward biased by the 1.4 volts at their bases ; the only current flow is through these transistors.

The read gate circuit conducts when all the address-line (low-order bits) inputs are positive and the read gate control circuit is conditioned by the read-timing inputs.

The read driver circuit conducts when all the address-line (high-order bits) inputs, located in the write gate portion of the diagram, are positive and the read driver control circuit is conditioned by the read-timing inputs.

The current in the read gate control circuit places a bias on the emitter of the center transistors of the driver control and gate control circuits, cutting them off.

Half-select current is now flowing through the selected X and Y drive lines in a direction to flip the cores from the 'one' state to the 'zero' state or 'read' the addressed position of core storage.

The circuits return to the inactive state, current flows in the gate control and driver control circuits only, after the read timing pulses fall. When the write timing pulses rise, current is switched to the write gate and write driver circuits and therefore flows through the array in the same drive line but in the opposite direction. This X and Y half-select current causes the selected cores to flip to the 'one' state, thus writing into the addressed position of core storage.

The V-Reference voltage applied to the current control circuit is temperature compensated, causing the X-Y drive current to track along its optimum value over a specified temperature range.

Drive Current Generation-Two-usec Storage

- Gate and driver circuits direct the drive current through the array.
- The read source circuit provides a constant current for the array during the read cycle.
- The write sink circuit provides a constant current for the array during the write cycle.
- A temperature compensated voltage reference is applied to both the read source and write sink circuits.

As in the four-usec core storage, the direction in which the half-select current flows in the X and Y address line is controlled by the driver and gate circuits. These circuits are conditioned by timing, for read and write portions of the cycle, and by address lines.

Comparison of the four-usec addressing diagrams with the two-usec storage addressing diagrams (SA 041 and SA 042) shows two basic differences.

The driver and gate circuit locations are reversed ; that is, where the four-usec storage uses the Y write gate, the two-usec storage uses the Y read gate. The other basic difference that can be seen in these diagrams is the current source and sink. The four-usec storage current flow (conventional current flow, pos. to neg.) is from a +8.5 v to the array current sink (-3 v) for both read and write currents. The two-usec storage read current flow is from the read source to -15 v and the write current flow is from ground to the write sink.

MDM page SA 043 is a detailed diagram of the two-usec storage drive circuits. Conventional current flow for the read portion of the cycle is from the read source circuit (+12 v), under control of read timing signals, through the read gate circuit that is selected by a decode of the address lines. The current flows through an isolation diode and the drive line that is connected to the active read driver at the other end of the array. The read driver is conditioned by a decode of the address lines and by a read timing pulse. The current flow is through the driver to -15 volts.

Write current flow is from ground, through the active write gate - activated by write timing pulse and an address decode. Current flows through the array line that is in the group of lines connected to the active write driver. Current flows through the write driver, activated by write timing pulses and an address decode, to the write sink circuit and -15 v.

The current magnitude is controlled by a temperature controlled reference voltage, V-Ref. that is applied to both the read source and write sink circuits.

The array termination, consisting of a resistor at the common of each group of isolation diodes, provides a characteristic impedance to the array lines. These circuits absorb the discharge pulse from the non-selected array lines and any reflected waves that build up on the lines.

Inhibit/Sense-Four-usec Storage

- The inhibit/sense winding is common to every core in one plane in the 4K array ; common to every core in one-half of one plane in the 8K array.
- During read time, the inhibit/sense windings conduct pulses, caused by the 'flipping' of selected cores, to the sense amplifier circuits.
- During write time, the inhibit/sense winding carries inhibit current to prevent the 'flipping' of selected cores.

Each inhibit/sense winding passes through 4096 cores of one plane. The 8K array has two inhibit/sense windings per bit-plane; the 4K array has one per bit-plane. Figure 2-14 shows the inhibit/sense scheme for the 8K array. Figure 2-16 shows the inhibit/sense scheme for the 4K array.

The functions of the inhibit/sense winding are explained in the Magnetic Core Theory section of this chapter.

Inhibit Driver

MDM page SD 051 shows the four-usec storage inhibit/sense circuits ; MDM page SD 051 shows some of the timing pulses of these circuits. The inhibit driver has two outputs that drive a center-tapped winding. Each output supplies approximately 200 ma of inhibit current. Data from the CPU Inhibit Switch is gated to the inhibit drivers by a storage timing pulse. When the inhibit driver is conducting, inhibit current prevents the writing of a 'one' in the selected core storage location.

Sense Amplifier

The sense amplifier has a differential input and a single-ended output. The sense control voltage determines the sensitivity of the amplifier and compensates for voltage supply and temperature variations. The first stage of this three stage amplifier is de-activated during the write cycle by the emitter-strobe pulse.

The first stage is kept out of saturation by a 0.7-volt offset voltage that is applied to the second stage. During the read cycle, the output of the second stage is gated to the final stage by the sense strobe. When a 'one' is sensed, the output of the final stage is a negative pulse.

Inhibit/Sense - Two-usec Storage

- The four-and two-usec storage inhibit/sense windings serve the same purpose, as described in the magnetic core theory section of this chapter.
- The connections on the array are different for the two units as shown in MDM page SA/SD 061.
- The two-usec storage inhibit/sense circuits are shown in MDM page SA 051.

Inhibit Driver

Three inputs condition the inhibit drivers, as follows: data, timing, and an address line. The data inputs are the Not lines from the Inhibit Switch that allow inhibit current to flow in the bit planes in which a 'one' bit is not to be written. The timing input is the inhibit time pulse, described in the Timing section that follows. The address line is the Address Register bit-3 line that separates inhibit drivers into less than 4K and greater than 4K circuits.

The inhibit driver output attaches to the inhibit/sense lines at the center point of 4096 cores. The two end-points of these lines connect to ground. Each inhibit circuit provides control for one bit-plane in a 4K array or for one-half a bit-plane (< 4K or > 4K) in an 8K array.

Inhibit current flow is from ground, through 2048 cores, through the inhibit driver that is conditioned, to -V Inhibit (V_Z). V_Z is a temperature responsive voltage that ranges from -9 to -12 volts.

Sense Amplifier

As in the four-usec storage, the two-usec storage sense amplifier has a differential input from the two ends of the inhibit/sense line for 4096 cores. The center point of this line connects to the corresponding inhibit driver.

The first stage collectors contain a transformer circuit that eliminates any dc unbalance to provide an equal detection-threshold for both polarities of signal. This stage does not require the emitter strobe that the four-usec sense amplifier requires. The 0.8 volts at the emitter of the second stage keeps the first stage out of saturation.

The sense - strobe pulse gates the output of the second stage to the final stage. A negative pulse output of the final stage results when a 'one' bit pulse is conducted from the array to the sense amplifier input.

Timing - Four-usec Storage

- MDM page SD 031 shows the timing pulses for the four-usec storage.
- X read gates and drivers are conditioned at T0 (CPU T6) time for a duration of approximately one usec.
- X write gates and drivers are conditioned at T4 (CPU T2) time for a duration of approximately one usec.
- The conditioning of Y read and write gates and drivers is delayed to allow X drive line noise to subside.
- Four-usec storage timing-pulse generation is initiated by the transition of the CPU read/write cycle.

Figure 2-17 represents the four-usec storage 'clock' and timing circuits. The CPU provides the following signals to the storage timing circuits : read cycle, write cycle, storage select, and the Address Register 3 line. The following timing pulses are generated by the 'clock' circuit: long time, short time, and strobe. Combinations of the CPU signals and the generated timing pulses provide read and write gate and driver control, strobe pulses for less than and greater than 4K, emitter strobe pulse, and inhibit pulses for less than and greater than 4K.

Note in the timing chart (MDM SD 031) that Y write current and Y read current rise after X read current and X write current.

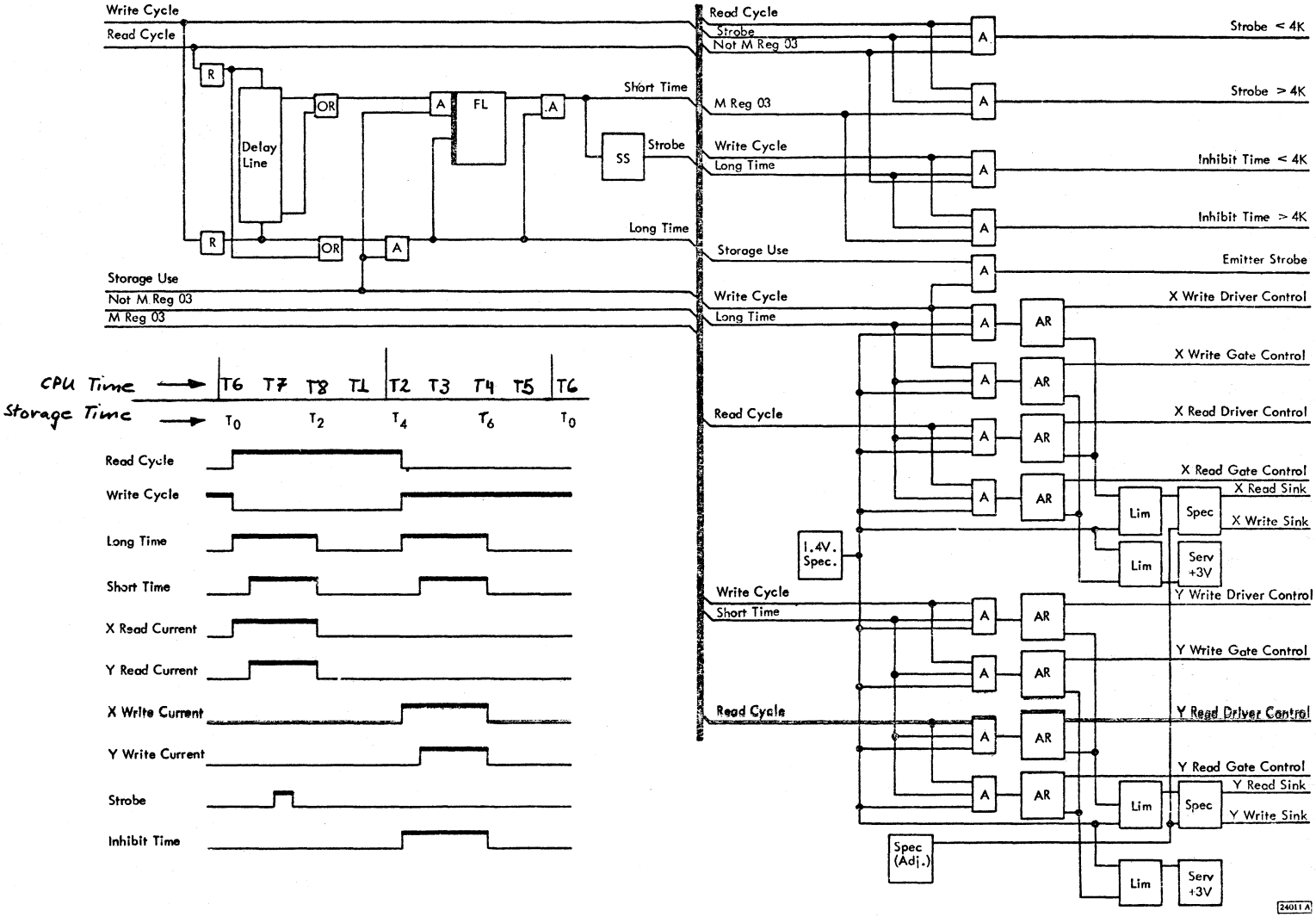


Figure 2-17. Four-Microsecond Core Storage Clock

The delay of the Y currents minimizes the effect of excess noise at the rise of the X current pulse. This noise is caused by the discharge of array capacitance.

Four - usec Storage Clock

- A time delay 'clock' circuit develops the necessary timing pulses for addressing, reading from, and writing into core storage.
- Long time : a time delay circuit provides a one usec pulse at T0 (CPU T6, read cycle) and at T4 (CPU T2, write cycle).
- Short time : a latch provides a pulse that begins after the rise of long time and ends with the fall of long time.
- Strobe : a single shot, activated by short time, provides a pulse that is ANDed with read cycle and an address line.
- Inhibit time : long time and write cycle are ANDed to develop inhibit time.

Time Delay Circuit (Figure 2-18). Read cycle and write cycle are connected to opposite ends of a one usec delay line. The low resistance of the delay line causes the levels at the inputs (points 1 and 2) to balance. For example, if read cycle is 0 volts and write cycle is +3 volts, points 1 and 2 balance at +1.5 volts.

When read cycle and write cycle change levels (at T0 and T4 time), the input points reflect the respective levels for one usec then balance again. The same effect is evident at the taps (points 3 and 4) except for a shorter duration.

Timing - Two-usec Storage

- MDM page SA 031 shows the timing pulses for the two-usec storage.
- Figure 2-1 shows the two-usec storage 'clock' and pulse generation circuits.
- X read gates and drivers are conditioned at T0 (CPU T6) time for approximately 750 nsec.
- X write gates and drivers are conditioned at T4 (CPU T2) time for approximately 750 nsec.
- Y read drivers and write gates are conditioned at the same time as the X drivers and gates.
- The conditioning of Y read gates and write drivers is delayed 100 nsec after the X drivers and gates to allow X drive line noise to subside.

The two-usec 'clock' consists of a time delay circuit three latches, and two single-shots. The timing pulse generation is initiated

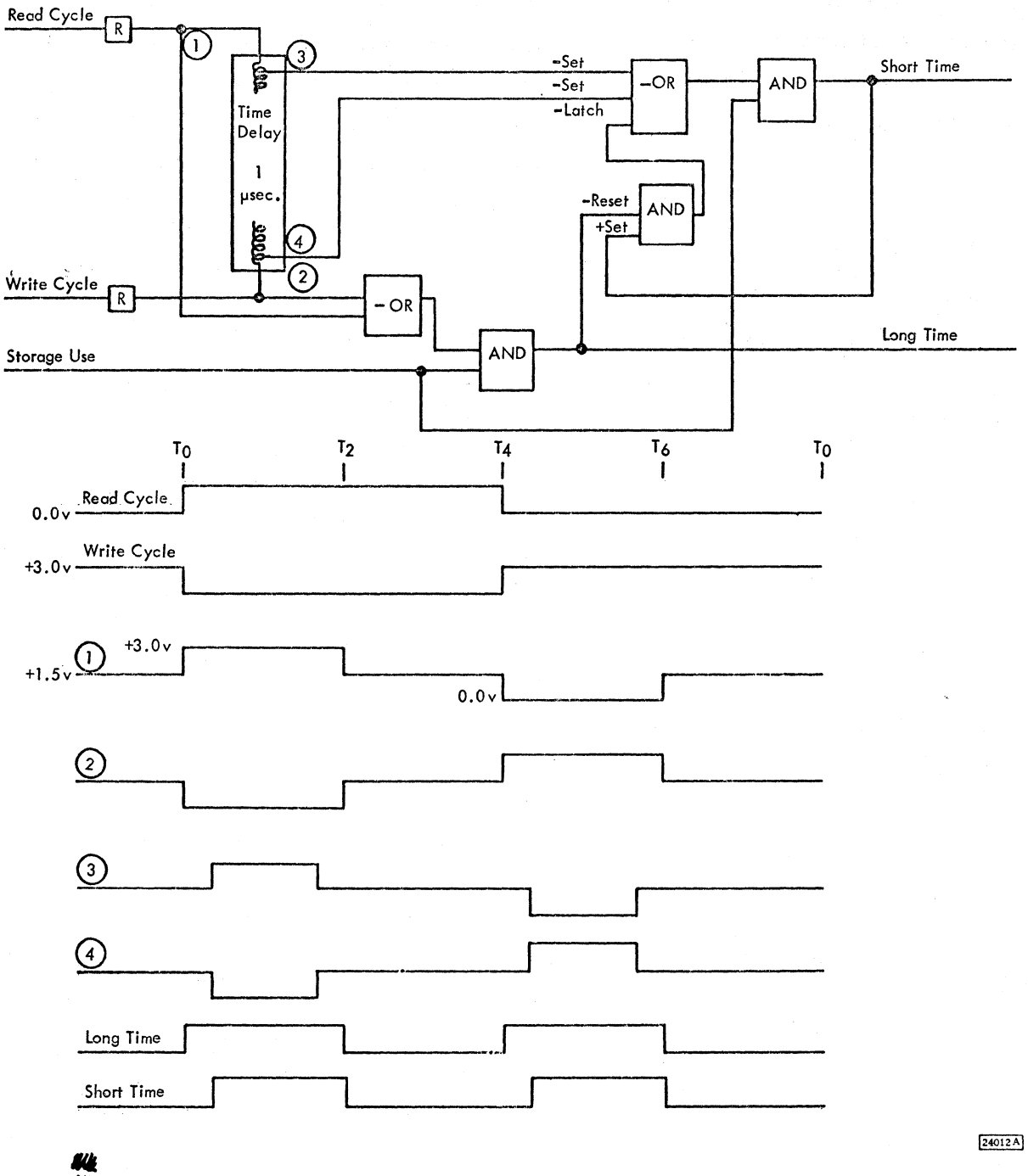


Figure 2-18. Time Delay Circuit, Four-Microsecond Storage Clock

by the storage select pulse from the CPU. Storage-select, which is a 100-nsec pulse at T0 and T4 or X0 and X4, turns on the long-time latch and the inhibit-time latch. Storage-select is also connected to the input of a time delay circuit for the following; the 100-nsec delayed output turns on the short-time latch, the 600-nsec output turns off long-time and short-time, and the 650-nsec output turns off inhibit - time.

The timing chart shows theoretical and actual timing for each signal. It shows long time coming up soon after T0 time, falling at the 750-nsec point, rising again soon after T4 time, and falling 750 nsec later. The inhibit-time latch is turned on at the same time that long-time is but the timing shown is inhibit-timing, after it has been ANDed with write cycle.
Short-time rises 100 nsec after long-time and falls at the same time. Strobe is a 330-nsec pulse generated by two single-shots that are started by short-time.

Long-time and short-time are ANDed, in various combinations, with read-cycle and write-cycle to produce X and Y gate/driver circuits are conditioned by long-time. The Y read driver and write gate is also conditioned by long-time ; but, the Y read gate and write driver are conditioned by short-time.

Strobe is ANDed with not-inhibit-timing and address-register-bit-3 to produce sense-strobe-8K ; not-address-register-bit-3 is used to produce sense-strobe -4K.

In the timing chart, the Y current (low order end) shows a pulse at T0 and T4 time. This pulse is present, due to the initial charging of array capacitance, although the X line is not conditioned at the other end at this time. The Y read current (high order end) and Y write current (high order end) begins at short time, when both ends of the line are conditioned.

Inhibit-time is ANDed with write-cycle to produce inhibit-timing, shown in the timing chart. Inhibit-timing is ANDed with the data lines and address-register-bit-3 or not-address-register-bit-3 (not shown in the diagram) to activate the inhibit drivers for less than and greater than 4K.

STORAGE ADDRESS REGISTER

- The SAR saves Main storage addresses for Read and Write.
- The addresses are of halfword formats
- Addresses are provided either by LS or ALU output.

The Storage Address Register (SAR, Fig. 4-53 FEMDM 2020 CPU) saves the current Main storage address to be available for the whole core storage cycle (Read/Write)
The SAR, able to hold addresses one halfword in length, consists of 18 latches numbered P0 to 15. Only the bits 0 to 15 are used for addressing.
The addresses may arrive either from the local store (LS) or from the ALU output (Sum Bits).

SAR Input Control

- The SAR Input is controlled by LS to SAR or ALU to SAR.
- The SAR is reset/set at T5.
- ALU to SAR is only performed for TRBS/TRBL and the RD format micro-instructions (cycle 0 or 1).

LS to SAR

This is the common signal. It is active when the Delta Process latch is on without any CPU check. However, ALU to SAR prevents LS to SAR.

ALU to SAR

This signal is necessary to place the ALU output into SAR. It can only occur when the Process latch is active and no CPU check is turned on. For Cycle Stealing, the ALU to SAR signal is suppressed (LS to SAR active). ALU to SAR is only forced by a small number of micro-instructions.
These micro-instructions are :

TRBS/TRBL
LH/STH
BZ/BM
BP/BAC
B (unconditional)

These instructions have one common function, viz., to change addresses either by branching or using a displacement address.
ALU to SAR can only occur during cycles 0 and 1.

SAR Output

- Addresses in SAR define bytes
- Since the core storage operates on a two byte (halfword) basis only the SAR bits 0 to 14 (even) are used for storage addressing.
- SAR bit 15 defines the byte in the halfword addressed by SAR bits 0 to 14.

The Main Storage is addressed on a halfword basis. However, the addresses in SAR define bytes. Halfword addressing is achieved by using the even part of the byte address (SAR bits 0-14) only.
The unit bit (SAR bit 15) is used to specify the left P0-7) or right (P1-15) byte of the halfword, which is read out from storage or will be written into storage.

SAR Powering

- SAR Powering provides the necessary powering of the address bit lines.
- After SAR Powering the SAR bits are defined as Address Register bits.
- Aux storage areas are addressed when SAR bits 0 and 1 are both on.
- The storage addresses for CPU LOG are forced by circuits.

SAR bits 0-14 are fed to SAR Powering.
The SAR powering circuits provide driver circuits for powering the address lines (to main storage) as well as logical circuits to force addresses for CPU LOG in, and to decode Auxiliary storage addresses.
The bit outputs of the SAR powering circuits are defined as Address Register bits to be compatible with the line definition of the core storage circuits.

SAR Check

- The SAR is checked for correct parity.

The SAR contents are checked for proper parity (odd). SAR Check causes Process Check and the CPU stops at the end of the current cycle.

The erroneous address in SAR remains unchanged.

SAR Display

The SAR contents are displayed on the CE console when the CE Display Select switch is set to SAR.

STORAGE DATA REGISTER

- The SDR is set directly by the core sense pulses at T8
- The SDR is reset when a core storage cycle starts (T6).

The Storage Data Register (SDR, Figure 4-54 FEMDM 2020 CPU) is a halfword register.
It is set by the halfword coming from the core storage sense amplifiers.
The Sense Bits are numbered from 1 to 17. The bits 16 and 17 are the parity bits P0 (16) and P1 (17).
The SDR is reset at each T6 (Start of Read time) when the Delta Process latch is on.

SDR Parity

- No Storage Use (no sense bits) forces SDR bits P0 and P1.

If Storage Use is prevented, all core storage functions are suppressed and no Sense Bits are provided. Thus, SDR would be blank (no data bits, no parity bits) after read time. However, SDR can be transferred to other CPU components even if Storage Use is prevented. This would cause any parity check. Thus, no Storage Use condition forces SDR bits P0 and P1 to provide correct parity.
Invert Parity condition, caused either by Storage Test operations or by operations with the Invert Parity switch (CE Console) turned on, prevents forcing of P0 and P1, because even parity (Inverted parity) is required in this special case.

SDR Output

- The SDR data is provided to : OP REG, TDR, Inhibit and as CS data out bits to the active CS device

The SDR output is fed to the Op Register (micro-instructions), to the To-Data Register (Data), to the Inhibit Switch (for regeneration), or to the Cycle Steal Unit (CS Data out).

SDR Display

The SDR contents are displayed on the CE console when the CE Display switch is set to SDR.

INHIBIT SWITCH

- The Inhibit Switch selects the store data sources.

o The Inhibit Switch processes always one halfword at a time.

The Inhibit switch (Figure 4-55, FEMDM 2020 CPU) contains the control circuits necessary to select the different sources, which can provide data to be stored. The sources for normal operations are SDR and ALU output. For Cycle Stealing , the data sources are the Cycle Steal buffers in the IOC, Storage Control Feature, or in the BSCA.
CPU LOG in uses special circuits and the normal operation circuits.

Normal Inhibit Operations

SDR to Inhibit

- The SDR data is regenerated.
- SDR to Inhibit is forced for SAR or Address Check.

The data movement from SDR to Inhibit is defined as regeneration , because the data read out during read time are immediately stored back into the origin storage position (Address in SAR is the same for Read and Write).

ALU to Inhibit

- The ALU output is stored during cycle 3.
- The result can be a halfword or a byte in length.
- ALU to Inhibit is prevented when a SAR or Address Check occurs.
- Only for Storage Test the result is stored during cycle 2.

The ALU output provides the final result, which shall be stored into core storage, during cycle 3. Only a few numbersof micro-instructions provide a final result of one halfword in length. These instructions are :

STH
SLM/SRM
MVH
AH/AHSC
SH/SHSC

During cycle 3 of these instructions, the complete ALU output (halfword) is transferred to Inhibit. However, transferring is suppressed in case of SAR or Address check (regeneration, see SDR to Inhibit). Especially for Storage Test (Store Runs 1 or 3), the halfword to be stored is already present at ALU output during cycle 2. Thus, transferring to Inhibit is performed during cycle 2.

Mixed ALU/SDR to Inhibit

- When the ALU result is one byte in length, the second Invert Switch byte derives directly from SDR (regeneration).
- SAR Bit 15 defines the Invert Switch byte into which the result byte is inserted.
- The byte result is always provided by Sum bits P1 to 15 (ALU loworder byte)

For micro-instructions , which provides a final result (cycle 3), only one byte in length, the

second byte of the halfword to be stored is provided by the SDR (regenerate the byte not affected). SAR Bit 15 (odd or even byte address in SAR) defines into which byte of the halfword the result byte has to be placed. The result byte is always provided by the right half (Sum bits P1-15) of the ALU output. Thus, there is a control to transfer ALU 8-15 either to Inhibit 0-7 or 8-15.
The byte to Inhibit movement is suppressed in case of SAR or Address Check (regenerate SDR).
For Storage Alter or Fill the byte movement occurs during cycle 2.

Inhibit Cycle Steal

- CS Inhibit control functions only for CS Read.
- CS Inhibit data are provided either as halfword or as byte.
- For CS byte transfer SAR bit 15 decides whether the highorder or loworder byte of the Inhibit Switch halfword is used.
- The Invert Switch halfword is completed by a SDR byte (CS Force Regen SDR).

During Cycle Steal (CS) operations the normal inhibit control circuits are blocked. The CS Inhibit control functions only for CS Read operations (data is written into core storage). Data is transferred from the CS devices to the Inhibit either in halfword or byte format.

For byte transfer (CS Halfword FL off), SAR Bit 15 decides whether CS Data in bits P0-7 or P1-15 are gated to Inhibit. The second byte of the halfword to be stored is provided from SDR, controlled by either CS Force Regen SDR 0-7 or CS Force Regen 8-15.
Both CS Force Regen signals are active for CS Write operations (regeneration of the halfwords read out and transferred to the CS devices).

Inhibit Check

- Incorrect (even) parity in either the highorder or loworder Inhibit byte causes Inhibit Check (indicator on CE console).
- Inhibit Check stops the CPU after the current cycle.
- The CPU stops after the current cycle.
- CS Inhibit Check does not stop the CPU immediately.
- Inhibit Check does not prevent writing of the faulty data.

An Inhibit Check occurs if either the inhibit bits P0-7 or the inhibit bits P1-15 are of incorrect parity. The inhibit check turns on the Inhibit Check latch, when operating normally, or the CS Inhibit Check latch, when operating in cycle steal mode.
The Inhibit Check latch signals the check condition via interface lines to the CS devices.
An Inhibit check during normal operations stops the CPU after the current cycle. The erroneous data is written into the addressed core storage position.

Inhibit LOG

- Inhibit LOG provides circuits able to test the contents of Data or Address registers for correct parity.
- The bytes of the tested register are forced to ones (/FF/) or zeros (/00/) depending upon the found parity.
- The correct byte is set to /00/, while the byte with wrong parity is forced to /FF/.
- Testing is performed during LOG cycle 2.
- During cycle 0, 1 and 3 the CE Display bits are selected as LOG information.

The LOG inhibit control has two different functions. During the LOG cycles 0, 1 and 3 , the LOG-in informations deriving from the CE Display Bits are gated to the inhibit.
The second function is performed during cycle 2 only. During this cycle the contents of the data or address register which could cause the check is investigated for its parity. The byte with the wrong parity is forced to /FF/ (all bits on) while the correct byte is set to zero. During Write time the halfword is stored. The halfword does not show the real bit pattern of the register, but the forced ones or zeros allow to define in which byte of the register the error has occured.

OPERATION CONTROL

OP REGISTER

The OP Register (OP REG, Figure 4-60, FEMDM 2020 CPU) is a halfword register used to save micro-instructions during their execution.

OP REG Input

- The SDR is set into OP REG at T1, delta cycle 0 (reset/set).
- During CPU LOG in, changing of OP REG contents is prevented (-LOG), because the contents is used as LOG in information (LOG in cycle 2, second halfword).

OP REG Input for MANOP

- For ICPL the Data switches 1 and 2 are set into OP REG bits 8 to 15 at T2, delta cycle 0.
- After T2 OP REG bit 8 to 15 contains the device address (8-11) and function specification (12-15) to read the initial loader card.
- For LS Display OP REG bits 8 to 11 (Data sw. 1) defines the LS zone, while the bits 12 to 15 define the LS register (Data sw. 2).

At T1 of Delta cycle 1, the SDR is gated into OP REG as for micro-instruction handling. However, Storage Use is prevented and SDR contains zeros (correct parity). Thus, the OP REG is set to zero. At T2, the two hexadecimal values set up in the data switches are placed into OP REG bits 8-15 (without parity). The values are used as device address and specify the I/O which is used to read cards for initial control (micro) program load (ICPL). For LS display or alter the value provided by data switch 1 (OP REG bits 8-11) specifies the LS zone (R1 Decode) while the value in data switch 2 selects one of the eight LS registers (R2 Decode).

OP REG Output

- The OP REG bits 0 to 3 define the OP REG Decode signals /0/ to /F/.
- The OP REG Decode signals are the main part of micro-instruction OP code.
- The OP REG bits 4 to 7 are either part of the OP code or, for R1, RD and I/O format instructions, the R1-field.
- The OP REG is tested for Invalid OP codes.
- OPREG bits 8 to 15 provide either the R1 and R2 field (FF-format) or the device

address and function specification (I/O-format)

- OP REG bits 4, 8, 12 or 15 can define direct or indirect operand addressing.
- The OP REG parity bits do not affect the decoding.
- OP REG bit P1 (sets FDR bit P0) allows parity checking of the Address Bus.

The bit pattern in the OP REG represents the micro-instruction to be executed. This bit pattern remains unchanged until the next micro-instruction is entered in (T1 of next delta cycle 0). The importance of the Op register bits and the generated signals are shown in detail in figure 4-61 in the FEMDM.

OP REG Decode

The OP REG Decode circuits investigate the four OP REG bits 0 to 3 (upper part of OP code). These decode circuits (AND combinations) provide control signals, called OP REG Decode 0 to F, according to the hexadecimal value in bits 0 to 3. The OP REG Decode signals are active during the whole execution time of the micro-instruction.

OP REG Bits 4 to 7

The OP REG bits 4 to 7 are either additional OP REG Decode informations or register addresses (R1, To-register) Invalid OP code. The OP REG bits 0 to 7 and 8 or 12 are checked for invalid OP code. The resulting control signal (Invalid OP code) forces end of operation immediately after cycle 0 (End Op CY 0, no execution).

OP REG Bits 8 to 15

For micro-instructions of R1 or DA format these bits carry the immediate data byte or the displacement address, which are not needed to control the operation.

For RR-format micro-instructions the bits 8 to 15 provide the addresses of the To-and From-register. The bits 8 and 12 are used to indicate whether the adjacent three bits (9-11 or 13 to 15) specify a LS register which accepts or provides data (direct addressing) or the specified register is used to address Main storage positions (indirect addressing).

The bits 8 to 11 are defined as R1-field (1. operand) while the bits 12 to 15 are defined as R2-field (2. operand). For R1-, DA- and I/O - formats , the R1-field is in bits 5 to 7 (4 to 7 if I/O).

The three address bits (5 to 7, 8 to 11, or 13 to 15), able to represent the values from 0 to 7, are decoded. The resulting signals are called either R1 or R2 Decode 0 to 7 in respect to the source field.

The switching between the two R1 fields is performed by investigating the Op code. OP REG bit 0 on (OP REG Decode 8-F) and not B, SENSE or CTRL defines the RR format instructions. Thus, the bits 8 to 11 are used. All other instructions use the bits 5 to 7 as the R1 field.

For LS display or alter (MANOP), the bits 9 to 11, set according to Data switch 1, are used for R1 Decode (selects the LS zone).

The micro-instructions SENSE and CTRL as well as ICPL use the bits 8 to 15 as device addresses and function specifications (SENSE and CTRL numbers), which are sent to the attachments (channels, control feature, BSCA) via the address bus.
The parity bit P1 is directly set into FDR bit P0 for parity checking of the returning address bus.
Especially for MVHS, the three bit address provided by the R1 field is incremented by one to provide the address of the second register for splitting (R1 A, during cycle 2).
The R1 field address is incremented by forcing the unit bit. Thus, incrementing is only possible if the R1 address is even. If R1 is odd, no higher adjacent register can be specified and splitting cannot be executed.

OP REG Display

The OP REG contents are not parity checked, but can be displayed on the CE console, when the CE Display Select switch is set to OP REG.

CYCLE CONTROL

- The Cycle Control switches the Long Time Clock.
- The Long Time Clock defines the delta cycle and cycle periods.
- Normally the Long Time Clock provides the cycles (delta cycles) 0 to 3 in sequence.
- The cycle sequence can be changed, depending upon the micro-instruction to be executed.
- The provided cycle control signals are: End Op Cycle 0, 1 or 2, Skip ahead, and Skip to repetition.

Depending upon the micro-instruction Op code and the bits indicating direct or indirect addressing (OP REG) the cycle control circuits (Figure 4-61, FEMDM 2020 CPU), generate signals to switch the Long Time Clock.
The Long Time Clock fixes operating periodes called cycles and delta cycles.
Normally the Long Time Clock steps from cycle (delta cycle) 0 to 3 and starts again with cycle 0. However, depending on format and type (direct or indirect addressing), a micro-instruction can end after any cycle (End Op) or can skip cycles (e.g., skip from cycle 0 immediately to cycle 3).

Cycle Control Circuits

- The instruction bit pattern in OP REG controls the cycle control circuits.
- The End Op signals cause advancing of the Long Time Clock to cycle (delta cycle) 0.
- The Skip ahead and Skip to repetition signals switch the Long Time Clock to any cycle (delta cycle), except

4. LS register 1 contains the field length.

The CLC, XX type, compares on byte basis. The compare result sets the condition code latches (CC0, CC1, CC2) when the CC bit (bit 6) in the micro-instruction is on. The result may be EQUAL (CC0) or UNEQUAL (CC1 or CC2). The cycle sequence for CLC, XX type, starts with cycle 0 and continues with function cycles 1 and 2. For ALC, the function cycles are repeated (Skip to repetition; Skip CY 2 to CY 1) until an End Op condition occurs. The field length is decremented by one and the data addresses (LS reg. 3 and 5) are incremented by one, each time the function cycles are performed.

End Op

The CLC with ALC ends when the End Op latch turns on. The End Op latch can be turned on at T3 during cycle 1 by two different conditions.

Result Unequal: Since the CLC with ALC compares two strings of bytes only two compare results, EQUAL or UNEQUAL, are possible. During each repetition two bytes are compared and the compare result sets the condition code latches. As long as the compare result is EQUAL (CC0), the comparing must be continued. However, if result is UNEQUAL (CC1 or CC2) , no further comparing is necessary, because this is the final result for the complete string of bytes.
The compare result, setting of the CC latches, is provided at the end of cycle 2 ; too late to terminate the operation immediately. Thus, an additional repetition is performed without setting a new condition code. During cycle 1 of this additional repetition, the End Op latch is set.

Field Length below zero: When the compared bytes are equal until the last repetition, a Decrement Carry 0 occurs when the field length in LS register 1 is decremented by 1 (T8 to T3) during cycle 1. This Decrement Carry 0 sets the End Op Latch (End Op gate, cycle 1, from T1 to T3).

Note that the compare result may be UNEQUAL for the last repetition and the CC latches are set accordingly.

The active End Op latch prevents the Skip to repetition (Skip CY 2 to CY 1) and forces End Op cycle 2 (skip to cycle 0). Thus, the Long Time Clock is set to cycle 0 after cycle 2 and the next micro-instruction starts.

SENSE or CTRL with ALC

The I/O format instructions SENSE and CTRL operate with ALC when indirect addressing (bit 4 on) is specified and R1 field addresses LS register 7 (IAR, bits 5 to 7 on). The R1 field is not used for addressing. It is part of the Op code (defines ALC). The data address is in LS register 7, while the field length is in LS register 1. The data address is incremented by one while the field length is decremented by one, each time a byte is transferred. Note that bytes are transferred depending on cycle time.

SENSE with ALC

After cycle 0 the Long Time Clock is advanced to cycle 3 (Skip ahead; Skip CY0 to CY 3). Cycle 3 is the function cycle (store cycle) during which the sense data enters the CPU and is stored into Main storage. The storage position is defined by the address in LS register 6.

After cycle 3 the Long Time Clock is set to repeat the function cycle (Skip to repetition; Skip CY 3 to CY 3). The Decrement Carry 0, which occurs when the field length is counted below zero (cycle 3), sets the End Op latch. The active End Op latch prevents Skip to repetition and forces a Skip CY 3 to CY 1. During cycle 1 the micro-instruction address (IAR) is updated. For ALC operations the IAR updating during start cycle (cycle 0) is prevented to allow recalling of the instruction in case of trap request interruption. The End Op CY 1 condition (skip to cycle 0) ends the operation and starts the next micro-instruction.

CTRL with ALC

After cycle 0, the Long Time Clock is advanced to cycle 2 (Skip ahead; Skip CY 0 to CY 2). Cycle 2 is the function cycle (read and operand 1) during which the control data is read out from Main storage and set on the data bus. After cycle 2 the Long Time Clock is set to repeat the function cycle (Skip to repetition, Skip CY 2 to CY 2). The Decrement Carry 0, indicating that the field length is below zero (cycle 2), turns on the End Op latch. The active End Op latch prevents skip to repetition and forces a Skip CY 2 to CY 1. During cycle 1 , the IAR is updated.

After cycle 1 , the End Op CY 1 condition (skip to cycle 0) ends the operation and starts the next micro-instruction.

CYCLE CONTROL FOR MANOP'S

- Since MANOP'S use the cycles 0 to 3 in sequence, no End Op and Skip ahead signals are generated.
- ICPL, Storage Scan or Fill, and Storage Test repeat functions, similar to micro-instruction with ALC, by Skip to repetition signals.

The MANOP'S use the cycles 0 to 3 in sequence Thus, no special cycle control for Skip ahead and End Op is necessary. All operations end with cycle 3. However, MANOP'S which must operate strings of data need repetitions of function cycles like micro-instructions with ALC. These MANOP'S are:

1. ICPL
2. Storage Scan or Fill
3. Storage Test

ICPL

- ICPL loads the CPL Loader card.
- ICPL ends when the loader card is read (field length decreased below zero) and the CPU starts execution of Load Control Program (micro-program routine).
- During ICPL, the CPU stops after cycle 0 when any I/O is working.
- ICPL continues with cycle 1, when any I/O working drops

- The CPU stops again after cycle 1 until a read request occurs.
- The read request starts the CPU again and cycles 2 and 3 are executed.
- After cycle 2 the CPU stops and starts again with cycle 2 (Skip to repetition), when the next read request occurs.

Operating the ICPL key starts the CPU. Processing starts (cycle 0) when the Delta Process latch and process latch are active.
After cycle 0 the CPU stops when any I/O is working. The Long Time Check is not advanced. When the any I/O working condition drops, the CPU starts again and the Long Time Clock advances to cycle 1. During cycle 1 the specified read device is started. After this cycle the CPU stops again. As for cycle 0, the Long Time Check still contains cycle 1 after CPU stopping.
Now the CPU waits for the trap request which indicates that a character has been read, which has to be transferred into Main storage.
The trap request starts the CPU and the Long Time Clock advances to cycle 2 and 3. During these cycles, the character enters the CPU and is stored into its specified Main storage position.
After cycle 3, the CPU stops until the next trap request (next character) becomes active. The Long Time Clock still contains cycle 3. Each time a character is transferred, the field length in LS register 4 is decremented by one, while the data address in LS register 0 is incremented by one. To store characters, the number which is specified by the field length, cycles 2 and 3 must be repeated. The repetitions are controlled by the Skip to repetition signal Skip CY 3 to CY 2, which sets the Long Time Clock accordingly each time the CPU is started again by a trap request.

The Skip CY 3 to CY 2 is prevented by the active End Op Latch, which turns on when the Decrement Carry 0 during cycle 3 (T8 to T3) indicates that the field length is decreased below zero. Thus, the Long Time Clock advances to cycle 0 during which the first microinstruction starts which has just before read in by ICPL.

Storage Scan or Fill

- Storage Scan or Fill repeats the function cycles 1 to 3.
- Storage Scan or Fill ends (End Op condition) when the CPU Stop Key is operated.

Storage Scan and Fill differ in data handling. During Storage Scan the data are read out for parity checking and stored back unchanged, while during Storage Fill a byte set up in the data switches is stored into each Main storage position within the customer area.
Storage Fill or Scan operations start with cycle 0 and the Long Time Clock advances in sequence until after cycle 3. The repetition of the function cycles 1, 2, and 3 is forced by the skip to repetition signal Skip CY 3 to CY 1. For Scan or Fill, no field length is necessary. The End Op latch turns on during cycle 3 when the CPU Stop key is operated (Stop latch on). The active End Op latch prevents further repetitions and drops the Hold Run condition to stop CPU.

Storage Test

- The Storage Test is performed by four runs.
- The Length Count (LC) latches are used as run counter.
- During Storage Test the core storage halfwords are loaded with their own address.
- Run 1 and 2 load and compare the addresses true while Run 3 and 4 load and compare the inverted addresses.
- During each run , the addresses are incremented by 2 from /0000/ up to /FFFE/.
- End Op condition (run end) is caused by an Increment Carry (address).
- For each address the function cycles 1 to 3 are repeated.
- Storage Test is terminated after Run 2, when the CPU Stop key is operated.

The Storage Test performs four different runs. The Length Count latches keep the information which run is in process.
The four runs are:

Run 1 = Store Run
Run 2 = Compare Run
Run 3 = Store Invert Run
Run 4 = Compare Invert Run.

A run ends when the End Op latch turns on during cycle 3 by an Increment Carry 0, which occurs when the highest possible address is exceeded.

This address is not the highest Main storage address (including control area), but the largest address which can be kept in a halfword (/FFFE/ = 65,434) when incrementing by 2.

Each run starts with address 0000 (protected area) and a four cycle operation (cycle 0, 1, 2, 3). For the following addresses up to /FFFF/, the function cycles 1 to 3 are repeated forced by the skip to repetition signal Skip CY 3 to CY 1. A run ends when the End Op latch turns on at T3, cycle 3, by an Increment Carry 0. The active End Op latch prevents the skip to repetition. Since the CPU is kept running (Process remains active), the Long Time Clock advances to cycle 0 , and the next run is initiated by a four cycle operation.

The End Op latch active at the end of each run from T3, cycle 3, until T3, cycle 0, advances the Length count (LC) latches.
At start Storage Test, the LC latch latches are all off indicating that Run 1 (Store run) has to be performed. The End Op latch turns on (T3 to T3) at the end of Run 1 and sets the LC 2 latch (T7, cycle 3, not AUX LC1).
The active LC 2 latch forces Compare Run 2. Run 2 starts and runs similar to Run 1, but with compare functions.
The End Op gate, (cycle 3, T1 to T3) caused by Increment Carry 0 at the end of Run 2, allows setting of End Op latch and AUX LC1 latch (LC 2 latch on) at T3.
The active End Op latch stops repetitions within Run 2 and the Long Time Clock advances to cycle 0.
The AUX LC 1 latch, active from T3, cycle 3, until T3, cycle 0 (reset), resets LC 2 latch (End Op latch on) and turns on LC 1 latch (End Op and not Invert Parity latch) at T7.

The LC 1 latch indicates the invert runs (remains active for Run 3 and 4) while the LC 2 indicates the compare runs (Run 2 and 4, normal and inverted).
The active LC 1 latch turns on the Invert Parity latch (Run 3, cycle 0, T3). This latch remains on during Run 3 and 4 (LC 1 active).
At the end of the Store Invert run (Run 3, End Op) LC 2 latch turns on again, while LC1 latch remains active. LC 2 and LC 1 latches on indicate Run 4 (Compare Invert run).
At the end of Run 4, the AUX LC 1 latch is active (cycle 3, T3, to cycle 0, T3), and resets LC 1 and LC 2. The dropped LC 1 latch allows resetting of the Invert Parity latch (cycle 0, T3). Thus, after Run 4 all LC latches are reset and Storage Test starts again with Run 1.

The Storage Test ends after Run 2, when the Stop key is operated. This provides correct parity in Main Storage.

Invert Parity Latch

- The Invert Parity Latch is active during Storage Test or when the Invert Parity switch (CE) is turned on.
- Invert Parity latch on cause CPU operations with even parity.
- Operating with inverted parity turns on all CPU check latches, but a Process Check (CPU stop) occurs only if a check latch fails to turn on.

The Invert Parity latch allows CE operations with inverted (even) parity. Beside Storage Test it is activated for Storage Fill or Scan if the Invert Parity switch on the CE console is turned on. Thus, the scan or fill operation is executed with the incorrect parity and all CPU check latches have to turn on. Only the Process Check latch remains off, but it is set (Attention indicator on) when not all CPU checks are turned on.
Thus, Process Check during inverted parity operations paints either to lost data bits or failing CPU checks.

LONG TIME CLOCK

- The core storage Read/Write time defines the CPU cycle time.
- The Read/Write time is devided into eight Timing (T) pulses.
- A core storage cycle lasts from T6 to T5, while CPU (data flow) cycle runs from T1 to T8.
- The complete CPU processing period (delta cycle and cycle) to T8 (end cycle).

The period the core storage needs for a read and write operation is a cycle. Thus, the cycle time depends on the core storage processing speed.
Each core storage cycle (Figure 4-65 FEM DM 2020 CPU) is devided into eight time segments named timing (T) pulses (four T-pulses for Read and four T-pulses for Write). Thus, a core storage operation is performed by eight steps. However, the basic period necessary to process data in the CPU (Data Flow), requires 13 steps. The additional

five steps are provided by overlapping the data flow processing periods. Thus, a processing period starts before the preceding one is terminated.

The period during which processing overlaps with the preceding processing period is called delta cycle. During delta cycle all those operations are performed, which do not influence the data handling of the preceding processing period. The delta cycle lasts as long as a cycle (eight T pulses). After the fifth T-pulse of a delta cycle, the cycle (also eight T-pulses) starts. Thus, the basic CPU processing period lasts from start of delta cycle until end of cycle (13 T-pulses).

When the CPU operates in Single Cycle Mode (CE), the period from start delta cycle to end of cycle is the actual processing time.

CPU processing is a sequence of delta cycles and cycles (processing periods) to perform functions specified by micro-instructions or MANOP's.

During one delta cycle and cycle only one access to the core storage is possible. Thus, micro-instructions or MANOP's, which require more than one storage access, also require more than one delta cycle and cycle.

Note: Since delta cycle and cycle are always considered together, the term CYCLE will be used to cover both.

Micro-instructions and MANOP's use up to four cycles. These cycles are numbered from 0 to 3. Each cycle performs a basic function:

Cycle 0 = Read out micro-instruction.
Cycle 1 = Read out second operand.
Cycle 2 = Read out first operand.
Cycle 3 = Store result or data.

The cycles 1 to 3 are defined as function cycles.

Long Time Clock Latches

- The Long Time Clock consists of two delta cycle latches (1 and 2) and two cycle latches.
- The delta cycle latches are controlled by the End Op, Skip ahead, and Skip to repetition signals at T4 (start delta cycle).
- The cycle latches (1 and 2) accept the delta cycle latch settings at T1 (start cycle).
- No End OP, Skip ahead, or Skip to repetition signal causes that the Long Time Clock advances automatically at each T4 pulse from 0 to 3 and starting again with 0.
- The delta cycle and cycle numbering depends on the binary value expressed by the latches (display CE console).
- The Reset Clock signal turns of all clock latch to define the start condition (cycle 0).
- Figure 4-64 in the FEMDM shows the advancing of the long time clock.

Long Time Clock Start and Stop

- Timed by the active Start latch, the Reset Clock signal occurs for Power on, system reset, ICPL or Load.

- The Long Time Clock advances for the first time at T4 after delta cycle 0 (Process latch active).
- Normally the CPU stops by a "Halt and Display" micro-instruction.
- Since the "Halt and Display" instruction causes the End Op CY 0 signal, the Long Time Clock is prepared (delta cycle 0) for the next start.
- After CPU check stop, the Long Time Clock contains either the error cycle information or the information of the cycle after the error cycle.
- For single cycling the delta cycle latches contain the condition of the next cycle (delta) while the cycle latches show the last performed cycle.
- The clock can be manually reset after check stop or single cycling when the Single Micro-Instruction switch (CE) is on and the Stop Key is operated.
- During CS operations, the Long Time Clock is not affective.

The Long Time Clock advances at each T4 when the CPU is running (Delta Process and Process latch on).

Reset Clock

For Start of Power on, ICPL, System Reset, or Load (customer's program) the Long Time Clock is reset by the Reset Clock signal.
The reset Clock signal is active as long as the Start Latch is on (T8 to T8).
The clock is reset to cycle 0.

Normal Stop

Normally the CPU is stopped by a 'Halt and Display' (HALT) micro-instruction. This instruction requires only one cycle (cycle 0) and forces the End Op CY 0 signal. At the end of delta cycle 0 (T4) Delta Process drops. However, the delta cycle latches are advanced once more by the End Op CY 0 signal, since Process is still active until T1. The acception of the cycle latches is performed, since this function is gated by Delta Process.
Thus, for a normal stop the same clock conditions (prepared to start cycle 0) exist as after Clock Reset. When the Start Key is pressed , the clock starts as described for Clock Reset, but without resetting.

Check Stop

Any CPU check stops the CPU either immediately after the cycle in process or after the next cycle.
After check stop the delta cycle latches are already set according to the next delta cycle, while the cycle latches still contain the last cycle performed.
This fact has to be remembered when identifying the error cycle, if an CPU check has occured, using the cycle display on the CE console.

Single Cycle or Single Micro-Instruction Stop

After this stop the delta cycle latches are already set according to the next delta cycle while the cycle latches still identify the last cycle performed (displayed on CE console).

Manual Reset Clock

Resetting clock (cycle 0) either after check stop or Single Cycle or Single Micro-Instruction can be achieved by operating the System Reset key, the ICPL key, or the Load key. However, operating one of these keys destroys valuable information (register content, indicators,and so on), since the system reset routine is performed.
To allow MANOP's (Alter, Display, Scan, Fill, Test) without resetting information. The clock can be reset manually.
Manual Reset Clock is performed when the Stop key is operated (Stop Latch turns on) with the Single Micro-Instruction switch (CE console) on, while the CPU stops.
Reset Clock signals are provided at T4 until the Single Micro-Instruction switch is turned off or CPU is started again (Stop latch turns off).

Clock Functions for Cycle Stealing

During the Cycle Steal cycle (T6 to T5, Cycle Steal latch on), advancing of the Long Time Clock is prevented. The cycle condition of the clock is ineffective. Since the clock is not advanced, normal processing starts again at that point where it was interrupted by cycle stealing.

LOCAL STORE

LOCAL STORE AND LOCAL STORE CONTROL

Local Store

- The LS, mounted on two cards, consists of 64 halfword registers.
- Each of the 64 LS registers can be separately addressed.

The Local Store (LS) is an internal storage device able to store 64 halfwords.

The LS (Figure 4-70 , FEMDM 2020 CPU) consists of 64 halfword registers (LS register) which can be separately addressed. Each LS register consists of 18 bit cells, which are considered to be latches. Thus, a LS register consists of 18 latches numbered from P0 to 15.

The Local Store is mounted on two cards. One card provides the latches P0-7 of all 64 registers, while the other card provides the latches P1 to 15.

LS Addressing

- A LS register is selected by a Select LS-Y signal and a X-address.

The LS registers are arranged in eight zones. Each zone contains eight registers (8 times 8 equals 64).

The zones as well as the registers within a zone are numbered from ø to 7.

The zones are defined by Select LS Y (ø-7) signals, while the X-addresses (ø-7) specify the registers.

LS Input Control

- To write a halfword or a byte into a LS WRITE signal must be active.
- Halfwords which shall be stored into LS are provided either by the ALU or by the Modifier.
- During direct addressed I/O SENSE's the byte provided by the Data Bus is directly set into the specified LS register low order byte, while the high order byte is is filled with zero.
- ALU to LS is performed at T5 or T8, while the MAR to LS condition is present for all other T-pulses.
- I/O Bus to LS occurs only at T8 and overrides ALU to LS.
- During CPU LOG the start address /C 000/ of the system reset routine is forced by the input circuits and is set into IAR of zone 7.

Data forced during CPU LOG in

- The LS input timing depends on the LS Write signal. Only this signal active allows storing of data into LS.

LS Write is generated by circuits depending upon the micro-instruction in the OP REG.

- For details see figure 4-70 in the FEMDM.

LS Output Control

- The halfword of an addressed register is provided at the LS output (LS Sense bits) when the SA gate is active.
- The SA gate is active from end of pulse A to end of pulse B when LS to TDR, LS to FDR, LS to MAR or LS to SAR is active.
- For details see fig. 4-70 in the FEMDM.

LS ZONE SELECTION

- The LS zones are selected by the Select LS Y signals 0 - 7.
- The LS Addressing (LSA) check turns on when none or more than one Select LS Y signal is active at the same time.
- The Select LS Y signals (Figure 4-71, FEMDM 2020 CPU) are activated as defined by :
  1. the Old or New Program Level (PL),
  2. Cycle Steal (CS) LS Select, or
  3. CE LS Select.

Zone Selection depending upon Program Level

- LS New PL and LS Old PL zone gate alternating generates the select LS Y signals depending upon the present New and Old PL.
- The New PL is active according to the highest priority request saved in the PL REG.
- The Old PL is provided by the Old PL REG.
- New and Old PL are different when a trap request occurs (accepted by PL REG) forcing a New PL until this New PL is set into Old PL REG.
- During the different PL period two different LS zones are selected timed by the Old and New PL zone gates.

A trap request set into the PL REG forces a New PL according to its priority. When the New PL is different to the Old PL (Old PL REG), the unequal PL condition activates the Old PL Control and the New PL is transferred into the Old PL REG.

Controlled by the LS New PL Zone gate and the LS Old PL Zone gate the Select LS Y signals are generated depending on the New or Old PL. Two different PL's are present from setting the trap request into PL REG (T3) until the New PL is transferred into the Old PL REG (T1). During this period (T 3 to T 1) Select LS Y signals are generated depending on the Old PL.

After the New PL has been transferred into the Old PL REG (T1), both PL's are the same until the New PL changes either by a higher priority request or by resetting the PL REG.

PL REG

- The PL REG consists of 7 latches (1 to 7)
- The PL REG latches are numbered according to the request priority
- The PL REG latches can be set by trap requests and CTRL's /09/ to /0F/. It can only be reset by CTRL's /01/ to /07/.
- The Reset PL Reg signals occurs during ICPL and sets PL REG 7, while the latches 1 to 6 are reset.

The PL REG consists of seven latches numbered from 1 to 7. These latches are set at T3 either by trap requests 1 to 7 or by CTRL's /09/ to /0F/ at T3, however, only when the corresponding trap request line is inactive.

The trap requests are reset during the service phases. These service phases end with a CTRL resetting the corresponding PL REG latch. The trap request line active at the service phase end indicates that the same trap request has been set again. Thus, the service phase starts again.

The contents of the PL REG are displaced on the CE console (lamps 1 to 7) when the CE display select switch is set to OCU 2.

The Reset PL REG signal forces PL REG latch 7 (highest priority) and resets the latches 1 to 6. The Reset PL REG signal is generated by the active ICPL reset latch, which is set when operating the ICPL key and is reset by CTRL 10 after the loader card has been read.

The Reset PL REG signal can be displayed (CE console, lamp 0) when the CE display select switch is set to OCU 2 (also display PL REG). The lamp 0 is off when the signal is active (negative potential).

New PL

- The New PL's are numbered from 0 - 7.
- A New PL is specified by the turned on PL REG latch having the highest priority.
- New PL 0 (CPU PL) is operational if no PL REG latch is active.
- MANOP's force New PL 7 without setting the PL Reg.

The output of the PL REG feeds circuits, which allow, that only that trap request is active, which has the highest priority. The output of these circuits provide the New PL. Thus, the New PL (1 to 7) agrees with the highest latch turned on in the PL REG.

Allow PL Switching

- Trap request is set into PL REG only when the Sense Trap Request Line signal is active.
- Sense Trap Request Lines occur when PL Switching is allowed (Figure 4-72, FEMDM 2020 CPU)
- Allow PL Switching defines the cycle at which changing of the PL does not influence the current basic operation.

The trap requests provided by the active trap request bus lines are gated into the PL REG by the Sense Trap Request Lines signal. This signal is active at T3 when the Allow PL Switching allows setting of PL REG and, conditioned by the new contents of PL REG, changing of the New PL.

Since changing of the New PL immediately influences the zone addressing (LS New PL Zone gate), Allow PL Switching defines the moment at which the LS zone addressing can be switched without affecting the current operation.

Generally Allow PL Switching occurs during the last cycle of a basic operation, which is either cycle 3 or the cycle during which the corresponding End Op signal becomes affective. For ALC operations Allow PL Switching occurs in the last cycle before performing the skip to repetition.

Allow PL Switching and Unequal PL set the Long Time Clock to delta cycle ø at T4.

Changing of Old PL

- Unequal New PL and Old PL moves the New PL into the Old PL REG.
- Unequal PL condition turns on the Change PL latch which controls setting of the Old PL REG.
- New and Old PL are identical until the priority in PL REG changes either by a higher priority request or by resetting the current PL.

For details see figure 4-71 in the FEMDM.

Alternating Use of Old and New PL

- Alternating Use of the Old and New PL depends upon the Old and New PL zone gate.
- Alternating Use of both PL's allows completing the 'old' micro-instruction, while the 'new' instruction is already initiated using the IAR of the New zone.
- The functions which are already performed using the zone information of the New PL are : IAR to SAR (T5), IAR to MAR (T6), IAR to FDR (T1).

The Old and New PL are alternating used to generate the Select LS Y signals controlled by the LS Old PL Zone gate and the LS New PL Zone gate.

According to the gate timing the Old PL is used at T3, T4, T7, T8, T2, while the New PL is used at T5, T6 and T2.

The following functions are performed by using the New PL :

IAR to SAR at T 5 :

This function places the micro-instruction address into SAR. Since the micro-instruction read out as the next is already the first instruction of the service phase, the IAR of the zone defined by New PL has to be used. Thus, at T 5 the LS New PL Zone gate is active.

IAR to MAR at T 6 :

This function places the IAR into the MAR for updating the micro-instruction address. Thus, the new zone has to be used.

IAR to FDR at T 1 :

This function places the IAR of the new zone into FDR for branch type instructions.

Old/New PL during CPU LOG in

- CPU LOG in forces New PL 7
- Since Old PL REG is saved during LOG in two different PL's address LS zones depending upon the zone gates.

During LOG, the MANOP signal forces New PL 7. The New PL 7 cannot be transferred into Old PL REG since LOG prevents reset/set of Old PL REG. Thus, New PL 7 and the present Old PL are used alternating during the four LOG cycles.

The LOG cycles 0 to 2 perform function without using the LS. During delta cycle 3 the IAR of the zone defined by the Old PL is read out at T7 (LS Old PL Zone gate), while the start address of the system reset routine (/000/, forced by circuits) is set into IAR of the zone specified by New PL.

During LOG PL REG latch 7 is forced on by Reset PL REG. The Reset PL REG signal is generated by the ICPL Reset latch. This latch turns on by the Reset Pulse which occurs during CPU start, when the System Reset key is operated.

Thus, the New PL 7 condition remains on, even if MANOP drops at the end of LOG in. The System reset routine operates in PL 7.

Zone Selection for Cycle Stealing

Caused by Any CS request and depending on the active CS Request latch (device 1, 2, 3, or 4) two LS zones can be selected at CS LS Select time.
The CS Request latch device 1 or 2 forces selection of zone 7 (Select LS Y 7), while the CS Request latch device 3 or 4 selects zone 1 (Select LS Y 1) Any CS Request is active from T3 to T2. The CS LS Select time lasts from TS to T2.

CE Zone Selection

- For LS Disalter the zone is selected by setting Data switch 1 to a value from ø to 7.

- For LR or STR the instruction bits 9 to 11 (R1-field) define the LS zone.

For MANOP LS Disalter the zone is selected by setting Data switch 1 (Customer console) to a value ∅ to 7. The value set up in the Data switch 1 is placed into OPREG bits 9 to 11. These bits feed the R1 Decode circuits.

The R1 Decode signals ∅ to 7 generate the corresponding Select LS Y signals.

For LR and STR micro instructions the R1 field (bits 9 to 11) specifies the zone, LS register 1 of which is used as first operand. The value in the R1 field forces the R1 Decode signals ∅ to 7. The R1 Decode signals are used during cycle ∅ either at T3 (LR) or at T8 (STR) to generate the corresponding Select LS Y signal.

LS REGISTER SELECTION

- The X-Addresses, which select any register in a specified LS zone, can be fixed or variable.

- X-Addresses are checked that not more or less than one address is active at a time (LSA Check).

- When none or more than one X-address is active at the same time, the LSA check latch turns on (indicator on CE console).

Variable X-Addresses

- Variable X-addresses are provided by the R1 or R2 field in the micro-instruction

- The R1 or R2 Select signal defines which field is used for addressing.

- If R1 (To-operand) is 7 (IAR addressed) LS WRITE is suppressed.

According to the R1 and R2 field of the micro instruction placed in the OP REG (T1 cycle0) the R1 and R2 Decode signals 0 to 7 are generated. These signals are active from cycle 0 T1 (set instruction into OP REG) to the next cycle 0, T1 (change OP REG by setting the next instruction).

The R1 Select and R2 Select signals (Fig. 4-73 and 4-74, FEMDM 2020 CPU) generated at specified T-pulses depending upon the present micro-instruction (OP Code), define which decode signals (R1 or R2) have to be used as X-addresses at which time.

When the R1 field specifies the LS register 7 (IAR, R1 Decode 7) writing data into LS at T8 (store result) is prevented.

Note: For LR and STR, R1 Decode 7 do not prevent LS Write, since for these instructions the R1 field specifies a zone. For STR writing into IAR is allowed. The LS register 7 is then specified in the R2 field.

Fixed X-Addresses

Fixed X-addresses (Figure 4-75, FEMDM 2020 CPU) are provided which select LS registers containing additional processing informations (e. g. , field length).

Fixed X-Address ∅

Packed Decimal instructions with ALC:
For these instructions LS register ∅ contains the field length of the second operand.

MANOP`s:
For MANOP`s LS register ∅ contains the data address.

CS operations, request device 1 or 3:
During CS operations, forced by device 1 or 3 requests, LS register ∅ contains the data address.

Fixed X-Address 1

Note: Fixed X address 1 for other than ALC or CS operations is a `Don`t Care` function.

All ALC instructions:
LS register 1 contains either the field length or the field length of the first operand (Packed Decimal).

CS operations, request device 1 or 3:
LS register 1 contains the field length for the CS operations.

Fixed X-Address 2

The LS register 2 contains the data address for CS operation forced by device 2 or 4 requests.

Fixed X-Address 3

The LS register 3 contains the field length for CS operations forced by device 2 or 4 requests.

Fixed X-Address 4

BST:
After BST, LS register 4 contains the address of the Main storage halfword into which the BST instruction address (IAR) incremented by 2 has been stored.

ICPL:
For ICPL LS register 4 contains the field length (defines the number of columns to be read from the loader card).

Fixed X-Address 5

No X-address 5 is forced by circuits.

Fixed X-Address 6

For SENSE or CTRL with ALC LS register 6 contains the data address.

Fixed A-Address 7

LS register 7 always contains the IAR.

LS WRITE

The LS Write signal active allows setting of data provided by A LU or Modifier output or by the data bus input (sense data) into the specified LS register (specified either by

variable or fixed addresses). The signal (Figure 4-76, FEMDM 2Ø2Ø CPU) is generated either depending on the micro-instruction or forced by circuits (MANOP or CS).

MAR AND MODIFIER

The MAR and the associated circuits are shown in Figure 4-80, FEMDM 2ø2ø CPU.

MAR

The MAR is a halfword register. The 18 MAR latches are numbered P ø to 15. The MAR can only be set by data received from the LS . The MAR output is provided to the Modifier, the test circuits, and the Address check circuits.

The MAR contents are displayed on the CE console, when the CE Display Select switch is set to MAR.

The MAR is not parity checked.

MAR Input Control

The LS Sense Bits P ø to 15, provided by the LS output from end of pulse A until end of pulse B (SA gate), are accepted by MAR at pulse B (reset/set), when the LS to MAR signal is active.

The LS to MAR signal can occur at T3, T5, T6 and T8.

MAR setting at pulse B ensures that the LS Sense bits are stable at LS output (end of SA gate).

Branch Go

- For branching the branch address (ALU output) is set into IAR and SAR.
- Branching is performed unconditional (TRBS, TRBL, B) or conditional (BZ, BM, BP, BAC).
- Branching is performed when the Branch Go latch turns on.
- For conditional branch instructions, the halfword to be tested is set into MAR.

At T3, during cycle ø, of the branch micro-instructions BZ (zero), BM (minus), BP (plus, but not zero), BAC (address check) the contents of a specified R1) LS register is placed into MAR.

The MAR output feeds test circuits, the output of which is gated by the corresponding micro-instructions (OP REG Decode, OP REG bits). If the MAR contents match with the branch condition the Set Branch Go signal is activated, which turns on the Branch Go latch at T4 (reset/set), cycle ø. The active Branch Go latch forces branching (Branch address to SAR and to IAR at T5).

The Branch Go latch is turned on unconditionally for B, TRBS and TRBL. The Branch Go latch is reset at T4 (reset/set) during cycle ø of the next following micro-instruction.

CS operations block Branch Go.

Test for MAR zero

For BZ the Set Branch Go signal is activated if the MAR bits ø to 15 are all zero.

Test for MAR minus

For BM, only the MAR bit Ø is investigated for being on, since this bit is the sign bit for a binary number placed in MAR.

Test for MAR plus

For BP, MAR bit Ø is investigated for being off. However, in addition any other MAR bit must be on (not zero).

Test for Address Check

- Address Check is caused either by addressing the protected area (/ ØØØØ / to / ØØ8F /) or by an address which exceeds the customer's Main storage area.

- Since the customer's storage area is variable in size, the address check circuits must be prepared by plugging jumpers according to the rented storage capacity.

  For CS operations, MAR bits 0-13 zero force the CS interface signal Counter zero.

When an address is set into MAR, it can be checked, that it does not exceed the available customer's area (Address check high) or specify the protected area (address check low).

Address Check High: Depending upon the available customer storage, the high order MAR bit 0 to 3 are investigated. See figure 4-80 in the FEMDM.

Address Check Low: The protected area addresses /0000/ to /008F/ are not directly accessible to the customer.

Thus, addresses programmed within the customer program cause an address check low if they are below /0090/.

The address check low circuit investigates the MAR bits 0 to 11 (three high order hexadecimal values). For details see figure 4-80 in the FEMDM.

For CS operation, MAR bits 0 to 13 zero forces the corresponding signal which is sent as Counter Zero interface signal to the CS devices (attachments).

Address and Halfword Boundary Check

- The operand addresses, used during micro-instructions, are checked for being in the customer's area when the AC bit (bit 7) is on.

- Either the Upper or Lower Address Check Latch turns on.

- Halfword operand addresses are unconditionally checked to be even (bit 15 off).

- An odd halfword address turns on the Halfword Boundary Latch.

- Any of the three check latches turned on forces Trap Request 2.

- The three latches are sensed and reset by a SENSE /16/.

Besides checking an address, currently present in a LS register, by a BAC micro-instruction the addresses used within a micro-instruction are checked for validity when the AC bit (bit 7) in the micro-instruction is on.

During halfword micro-instruction, like MVH, MVHS, AH, AHSC, SH, and SHSC, the used addresses are checked to be for halfword boundaries (even addresses, MAR bit 15 off).

The three check conditions. address check high. low, and boundary check, turn on latches named accordingly. Setting of the latches is prevented when any CS request occurs or when Storage Use drops (addressing of not available storage positions, including control area).

Setting of the latches is timed by T6 during delta cycles 1, 2 and 3. Since during these delta cycles operands are processed, only the operand addresses are checked. Note that the addresses are already placed into SAR at T5.

The upper or lower Address Check latch active forces Regen Halfword. This signal causes that the halfword read out by the faulty address is re-generated.

During Storage Fill or Alter (not CE mode switch on) the procedure allows the customer to display the control area but not to alter any data in it.

Any address check latch on forces trap request 2 (not affective during MANOP Storage Alter or Fill, PL 7 is forced).

This allows servicing the address check by a special micro-program routine.

During this routine a SENSE /16/ can be used to decide which check caused the trap request 2.

The check latches are reset either at T8 during SENSE /16/ or when the CPU is stopped (not Process, end Storage Alter). For Storage Fill or Alter the High and Low address check latches are reset each time at T6 during delta cycle 1, 2, or 3.

MODIFIER

- The Modifier increments or decrements LS halfword by 1 or 2.
- For parity checking the Modifier parity (P0, P1) is predicted.

The Modifier can increment or decrement data by one or two. The result returns to LS register. During ICPL data pass the Modifier unchanged (start address from Address switches in LS Reg 0 moves to IAR, zone 7).

The Modifier output is parity checked. The Modifier parity bits PØ and P1 are predicted according to the parity in MAR, the function to be performed, and bit pattern in MAR.

Modifier Control

- Increment by 1 and Decrement by 1 or 2 are control signals generated depending

upon the operation to be performed.

- Increment by 2 is active when no other Increment or Decrement signal is generated.

- During CS operations the Modifier is controlled by the CPU CS unit.

The Modifier control circuits provide the Increment or Decrement by 1 or 2 signals (Figure 4-81, FEMDM 2ø2ø CPU). The signals are generated depending upon the basic operation to be executed and are timed by delta cycles (T4 to T3) or half delta cycles (T4 to T7 or T8 to T3). Increment by 1 and Decrement by 1 or 2 are controlled signals, while Increment by 2 is forced if no other Modifier control signal is active. Thus, Increment by 2 is active generally. To move data through the Modifier unchanged , this common control signal is suppressed (no Increment or Decrement).
During Cycle Stealing all four Increment and Decrement possibilities are controlled by the CS Modifier Control. The result is present at Modifier output only as long as the corresponding Increment or Decrement signal is active.

Modifier Circuits

- The Modifier consists of two seperate circuits, able to process one byte.

- The low order byte provides a carry (Increment or Decrement Carry 8) to the high order byte.

- The carry out of the Modifier halfword is defined as Increment or Decrement Carry ø.

The Modifier is mounted on two similar cards. One card contains the Modifier circuits for the loworder byte (P1 to 15), while the Modifier circuits for the highorder byte (P ø to 7) are on the second card.

The carry information from the loworder byte to the highorder byte is provided either by the Increment Carry 8 or Decrement Carry 8 signal.

An overrun out of both bytes is indicated either by the Increment or Decrement Carry ø.

The Carry 8 is a internal Modifier signal, while the Carry ø is used as CPU control signal.
For details of the modifier circuits see fig. 4-80 in the FEMDM.

Modifier Parity Correction

- MAR bits P0 and P1 predict the Modifier parity.

- The predicted parity is changed when a defined MAR bit pattern indicates that the Modifier operation changes the status (on to off, off to on) of an odd number of MAR bits.

Modifier Check

- Even parity in either the highorder or loworder Modifier byte turns on the Modifier Check Latch (indicator on CE console).

The Modifier output is parity checked. An even number of bits (including parity bit) either in the highorder or loworder byte turns on the Modifier Check latch (indicator on the CE-console).

LOGICAL UNIT

- The Logical Unit consists of FDR-Invert Switch, TDR-Shift Unit, and ALU.

The ALU is able to perform logical as well as arithmetical operations using operands provided by FDR and TDR. The logical operations are ANDing, ORing and Exclusive ORing. ADDing is the only arithmetical function which can be executed.

Before the operands, provided by FDR or TDR enter the ALU, they can be modified either by the Invert Switch (FDR operands) or by the Shift and Suppress Unit (TDR operands).

FDR AND INVERT SWITCH

- FDR keeps one operand during ALU operations until the result is saved.

- The Invert Switch provides FDR bits to the ALU either true or inverted.

- The Invert Switch can force zeros or ones.

FDR and Invert Switch are shown in figure 4-9Ø, FEMDM 2Ø2Ø CPU. The FDR is a short time storing device, which save one operand during ALU operations. This operand is provided either by LS, ALU output (preceding result) or by sensed data from the I/O Bus.

FDR data are fed through the Invert Switch before entering the ALU.

The Invert Switch can be controlled, so that FDR data passes unchanged (True) or inverted. If the True and Invert control signals are active at the same time the Invert Switch forces the FDR data to zero, while at a time, when both signals are absent, the output of the Invert Switch is fixed to ones.

FDR Set

- LS to FDR, ALU to FDR or I/O Bus to FDR selects the halfword which is set into FDR.

- ALU to FDR causes the Reset FDR signal (T6), while the other conditions reset/set the FDR.

- Retain FDRØ-7 causes that during TRBS or TRBL, cycle Ø, LS to FDR only set the loworder FDR byte. The highorder byte is retained.

- For I/O Bus to FDR the Address Bus parity bit (OP REG bit P1) directly sets the FDR bit PØ (Address Bus to FDR bits Ø-7).

- Sense data of CPU SENSE /14/ or /15/ are one halfword in length.

- For CPU SENSE's parity checking is inhibited since CPU sense data are provided without parity bits.

LS to FDR

LS to FDR signal generated at T1, T3, T4, and T7 depending upon basic operations allows the LS Sense Bits P0 to 15 to enter the FDR.

For TRBS and TRBL, T3, cycle 0, the Retain FDR signal is activated. The active Retain FDR signal prevents resetting of FDR bits P0 to 7 and blocks the FDR input for the LS Sense Bits P0 to 7. Thus, only the LS Sense Bits P1 to 15 are placed into FDR while the FDR bits P0 to 7 remain unchanged.

ALU to FDR

When the ALU to FDR gate is active, the ALU to FDR signal gates P0 to 15 from ALU into FDR. The FDR was reset at T6 by the Reset FDR signal.

I/O Bus to FDR

The I/O Bus to FDR signal sets the Sense Bits 0 to 7 and P1 to 15 into FDR. The parity bit P0 derives directly from the OP REG bit P1. The bits 0 to 7 are provided by the Address Bus while the bits P1 to 15 are the sensed data byte provided on the Data Bus.

CPU sense data, which are provided either when performing a CPU SENSE, an ICPL SENSE, or when the MANOP inbus to FDR signal is active, is set into FDR at T4 (cycle 0 or 1). Since CPU sense data is provided without parity the Prevent ALU and SU Check signal is generated.

FDR Parity correction

- To correct FDR parity bits for CPU sense data the signals Force or Turn off P1 or PØ are generated by the ALU parity correction circuits.

- For parity correction the FDR bits 8-11 are moved to the Shift Unit (INT SU bits 4-7) during TRBS. cycle Ø. only.

The FDR contains 18 latches numbered PØ to 15. The latches are reset when new data shall be placed in (reset/set). The FDR contents are displayed on the CE console when the CE Display Select switch is set to FDR.

To provide correct parity for CPU sense data the parity latches PØ and P1 can be set or turned off controlled by the ALU Parity Correction circuits which generate the CPU sense data parity when the Prevent ALU and SU Check condition exists. Both parity bits are affected since CPU sense data may be a halfword in length

For TRBS. CY Ø. the FDR bits 8 to 11 are transferred to the Shift Unit (becomes INT SU bits 4 to 7). Transferring these bits to the SU saves the parity information since the bits are forced to zero by the Invert Switch (TRBS only). If an odd number of bits is forced to zero the FDR parity bit P1 is wrong

for the Invert Switch output. This would also cause wrong parity P1 of the ALU result. However, the saved parity information in SU can tell, that either an even or odd number of bits has been forced to zero and a Change Parity (P1) signal is generated accordingly within the ALU Parity Correction circuits.

Invert Switch

- The Invert Switch is controlled by FDR True or FDR Invert.
- FDR True affects the complete FDR halfword, while three FDR Invert signals (Ø - 7, 8 - 11, 12 - 15) can affect bit groups.
- True and Invert force the FDR bits to zeros (parity bits on).
- Not True and not Invert force ones (parity bits on).
- Possible control combinations are: True and Force zeros or Invert and Force ones.

The Invert Switch consists of AND-OR combinations which are activated by the invert switch control signals `FDR True` or `FDR Invert` (Figure 4-91, FEMDM 2Ø2Ø CPU). When the FDR True signal is active the FDR bits PØ to 15 run through the Invert Switch and become the Invert Switch bits PØ - 15 (Invert Switch output) unchanged (true).

For invert operations the FDR bits 0 to 7, 8 to 11 and 12 to 15 can be selected by separate FDR Invert signals.

An active FDR Invert signal causes the corresponding FDR bits to be inverted at the Invert Switch output.

Beside the true or invert function, the Invert Switch is able to force zeros or ones at the output.

To force zeros the FDR True and all three FDR Invert signals must be active at the same time.

It is possible to force either the FDR bits 0 to 7, 8 to 11, or 12 to 15 to zero when only one of the three Invert signals is active. The other FDR bits remain unchanged.

The Invert Switch output is forced to ones when both control signals (True and Invert) are absent at the same time.

The Invert Switch Control provides the True and Invert signals generated depending on the basic operation in process.

Invert Switch Parity

- Invert Switch functions are normally performed on byte boundaries.
- True or Invert on byte format does not influence the parity.
- Forcing a byte to zeros or ones always results in even parity and, thus, the parity bit has to be turned on.

The Invert Switch functions are performed either on halfword or byte boundaries, except during TRBS when the bits 8 to 11 are forced to zero and the bits 12 to 15 are transferred true.

When a byte moves true (unchanged) through the Invert Switch, the FDR parity becomes the Invert Switch parity.

For FDR Invert on byte boundary the FDR parity also becomes Invert Switch parity since the number of turned on bits is not changed by inverting.

When a byte is forced to zeros or ones, the corresponding parity bit is turned on at the Invert Switch output. However, during operation with Invert Parity mode (Invert parity latch on) setting of the parity bit of a byte forced to zeros or ones is inhibited to provide even (incorrect) parity.

TRBS - Invert Switch Parity

- Two different Invert Switch functions in a byte cause undefined parity.
- The Invert Switch parity bits are used to predict the ALU parity bits and the prediction is corrected if an odd number of turned on bits (8 to 11) is forced to zeros.

TRBS is the only basic operation during which Invert Switch operations are not performed on byte boundaries.

During cycle ∅ the bits 8 to 11 are forced to zeros while the bits 12 to 15 as well as the bits ∅ to 7 are transferred true.

Since the bits 8 to 11 are forced to zeros the parity of the loworder byte at the Invert Switch output (Invert Switch bit P1) is undefined. The FDR bit P1 becomes Invert Switch bit P1

The Invert Switch parity bits are used for ALU parity prediction. Thus, an information is needed which tells wether the Invert Switch bit P1 is correct or not. In case of incorrectness the P1 information must be changed to get correct ALU parity.

Invert Switch bit P1 is incorrect when the FDT bits 8 to 11 (forced to zeros) provide an odd number of turned on bits.

During TRBS cycle ∅ the FDR bits 8 to 11 are transferred to the INT SU bits 4 to 7. The Shift Unit parity circuits generate the signal `Odd SU Bits 4 to 7` if the number of turned on bits is odd. The active Odd SU Bits 4 to 7 signal causes changing of the ALU parity P1 which is predicted depending upon the Invert Switch bit P1 (can be incorrect) and the INT SU bit P1.

Invert Switch Display

The Invert Switch output is displayed on the customer console in Data Register (DR) display E (P0, 0 to 3), S (4 to 7), T (P1, 8 to 11), and R (11 to 15).

During TRBS the byte displayed in DR-T and R may show incorrect parity (P1),

since the change parity function (bits 8 to 11 forced to zeros) affects only the parity of the ALU output (result).

TDR AND SHIFT UNIT

TDR and Shift Unit (Figure 4-100, FEMDM 2020 CPU) represent the second input of the Logical Unit. The TDR is set by data which derives either from the LS or from the SDR.

The TDR data enters the ALU via the Shift Unit (SU).
The SU can shift TDR data by eight, four or two bits to left or to right. The SU can also suppress (set to zero) the complete TDR halfword or selectable bit groups of the TDR halfword.
The SU contains circuits to perform sign or packed byte testing and circuits which are able to normalize signs according to the EBCDIC or ASCII standard.

For CTRL's, TDR data is provided to the Data Bus after the eight shift circuits.

TDR

The TDR is a halfword register consisting of 18 latches numbered P0 to 15. The TDR contents is displayed on the CE console when the CE Display Select switch is set to TDR. The TDR contents is changed only when new data is set (Reset/Set).

The TDR can be entered from the SDR or the LS always one halfword in length.

The TDR is not parity checked.

Eight Shift

The TDR output directly feeds the eight shift circuits. After the eight shift circuits the bits are defined as Internal (INT) SU bits P0 to 15.

Four eight shift functions are possible:

Cross Shift
Eight Shift Left or Right
No Shift Halfword
No Shift Byte

The four shift functions are controlled by four eight shift control signals:

No Shift - TDR 8-15 to SU 8-15
TDR 0-7 to SU 0-7
Shift - TDR 0-7 to SU 8-15 (right shift by eight)
TDR 8-15 to SU 0-7 (left shift by eight)

The Eight Shift output is parity checked (SU Check).

Cross Shift:
Cross Shifting, exchange of highorder and loworder byte, is performed when both shift signals (TDR Ø-7 to SU 8-15, TDR 8-15 to SU Ø-7) are active during the same cycle.

Eight Shift Right or Left:

Only one shift signal, TDR Ø-7 to SU 8-15 or TDR 8-15 to SU Ø-7 is active. The not affected INT SU bits (Ø to 7 or 8 to 15) are set to zeros.

No Shift Halfword:

When the two No Shift signals, TDR Ø-7 to SU Ø-7 and TDR 8-15 to SU 8-15, are active at the same time, the halfword in TDR is moved to the INT SU bits (PØ to 15) without shifting.

No Shift Byte:

Only the No shift signal TDR 8-15 to SU 8-15 is active. The loworder byte in TDR (P1 to 15) is moved to the INT SU bits P1 to 15. The highorder byte (INT SU bits PØ to 7) is set to zeros (TDR remains).

No Shift Halfword and Cross Shift are the basic function of the Eight Shift circuits. No Shift Byte and Eight Shift Right or Left are sub-functions which are achieved by additionally controlling the No Shift Halfword or Cross Shift control signals.

For details of shift control signal generation refer to figure 4-101 in the FEMDM.

Eight Shift Parity

Eight Shift functions do not affect the parity. Thus, the TDR parity bits are transferred with the corresponding byte and become the INT SU parity bits.

The parity bit of a byte set to zero by not No Shift and not Shift is forced on by circuits to provide odd parity. However, for inverted operations (Invert Parity Latch active) forcing of the parity bit is suppressed (even parity). Incorrect parity (even) of the INT SU bits (highorder or loworderbyte) activates the SU Check latch and the corresponding indicator on the CE console turns on. The check is inhibited during CPU Senses (Prevent ALU and SU Check).

Shift Unit

- The internal (INT) SU bits, provided by the Eight Shift, can be shifted by 2 or 4 bits to the left or to the right.
- Shift is performed on halfword basis.
- The SU output bits can be suppressed in groups of 4 (highorder byte) or 2 (loworder byte) bit groups.
- Normalize Sign allows standardization of the packed decimal signs (/A/ to /F/ valid) according to the required standard code (ASCII or EBCDIC).
- The SU output is displayed in Data Register display P, I, U, and L.

The internal (INT) SU bits (P0 to 15) provided by the Eight Shift circuits are used as input data for the Shift Unit (SU).

Data is fed through the SU either shifted by 2 or 4 bits to the left or to the right, or unshifted. Data is shifted on halfword basis. Bits shifted out of the halfword are lost, while the additional bits are set to zeros.

The output of the SU provides data to one ALU entry. The output of the SU can be suppressed (set to zero) either partially or completely controlled by the Suppress circuits. Also circuits are provided to standardize signs to test for correct sign bit pattern and to test for packed decimal data. The output of the SU is constantly displayed in the Data Register display (Customer console) E (PØ to 3), S (4 to 7), T (P1 to 11), and R (12 to 15).

Shift by 2 or 4

- Shift by 2 or 4 is performed during SLM, SRM and MVHS only.

- A shift by any amount of bits to the left has the effect that the respective binary high order bits are lost and an equal number of free binary low order bit positions are created. These low order positions are set to zero. A reverse situation occurs with the right shift.

Shifting by 2 or 4 bits left or right is performed by AND switch combinations which are activated by one of the four shift control signals (Figure 4-1Ø1, FEMDM 2Ø2Ø CPU):

Shift Left by 2
Shift Left by 4
Shift Right by 2
Shift Right by 4

Only one of these signals is active at a time. However, the shift by 2 or 4 can be combined with shift by 8, since Eight Shift and SU operate independently.

The shift control signals are only generated for SLM, SRM, or MVHS (1, 2, 3).

For SLM and SRM, the shift which shall be performed is specified in the micro-instruction bits 5, 6, and 7 (OP REG bits 5, 6, or 7). Bit 5 on activates the Eight Shift control.

Details of shift control signal generation are shown in fig. 4-101 in the FEMDM.

For example the MVHS 1 causes generation of the Shift Right by 4 signal during cycle 2.

No Shift Control for ADDI

- Suppressing the No Shift 0-7 signal causes forcing of ones into the high-order SU byte.
- The highorder SU byte forced to ones allows extension of a negative immediate data byte to a negative binary halfword number.
- A positive immediate data byte causes suppressing of the highorder byte set to ones.

For ADDI the control signal 'No Shift Ø to 7' is suppressed. Since also no Shift by 2 or 4 signal is active during ADDI no control signal affects the bits Ø to 7. This condition forces turning on the bits Ø to 7 (hexadecimal /FF/).

The SU bits Ø to 7 (highorder byte) forced to ones extend the immediate data byte (low-order) to a negative binary number one halfword in length (/FF../). Since for ADDI the highorder byte in SU is generally forced to

ones the suppress circuits are controlled according to the sign bit (OP REG bit8) of the immediate data byte. Sign bit off, indicating a positive binary value, causes suppressing of SU bits Ø to 7. Since suppressed bits are set to zeros the immediate data byte is extended by zeros which is true for a positive binary number.

Sign bit on, indicating a negative binary value, allows that the bits Ø to 7 (forced to ones) are provided to the ALU. The immediate data byte extended by ones (/FF/) represents a negative binary number one halfword in length. Addition of a negative value is similar to a subtraction.

Normalize Sign

- During SDS the packed decimal sign (/A/ to /F/) provided by bits 12 to 15 of the loworder INT SU byte is standardized according to the selected code (ASCII or EBCDIC).

The Normalize Sign circuits are used to convert a packed decimal sign provided by the INT SU bits 12 to 15 into the standard sign according to the selected code (EBCDIC or ASCII).

The circuits are active during the SDS micro-instruction (Normalize Sign signal). Within the CPU the hexadecimal values /A/ to /F/ are accepted as valid signs. The values /A/, /C/, /E/ and /F/ are considered as positive signs, while /B/ or /D/ is accepted as a negative sign.

Standard Sign EBCDIC:

+ = /C/ (11ØØ)
- = /D/ (11Ø1)

Standard Sign ASCII:

+ = /A/ (1Ø1Ø)
- = /B/ (1Ø11)

The three highorder bits of a standardized sign are similar for minus or plus. Thus, for standardization the three highorder bits (SU bits 12, 13, 14) can be forced directly depending upon the selected standard code (EBCDIC = 11Ø. ; ASCII = 1Ø1.). The selected code is defined by the ASCII latch turned on (ASCII) or off (EBCDIC). See figure 4-100 in the FEMDM.

The loworder bit (SU bit 15) has to be turned on or off to define a negative or positive sign.

The turning on or off of the loworder sign bit is controlled by the Normalize Sign Bit 15 signal, which is active (negative) when the sign is negative.

Since all sign representations have the bit 12 turned on, only the bits 13 to 15 have to be investigated to define the sign polarity.

The Normalize Sign Bit 15 signal is active when INT SU bit 15 is present and either the bit 14 (/D/, 11Ø1) or the bit 13 (/B/, 1Ø11) is off. This condition defines the characteristics of both negative signs, /B/ and /D/, only. The bit patterns of the positive signs cannot activate the Normalize Sign Bit 15 signal and, thus, SU bit 15 is set to zero for standardized positive signs.

Normalizing Signs does not affect the parity (number of turned on bits in the loworder byte) with the only exception: Normalizing sign /E/.

Normalizing the sign /E/ (three turned on bits, 111Ø) to the positive standard sign /A/ or /C/ (two turned on bits, 1Ø1Ø or 11ØØ) causes parity changing (P1) in the ALU parity correction circuits.

Note that Normalize Sign forces the SU bits 12 to 15 to ones by preventing the No Shift functions (No Shift 8-15 signal active) for these bits. The turned on bits 13 to 15 are provided to AND switches which are controlled by the Normalize Sign circuits while the turned on bits 12 (common for all signs) directly becomes SU bit 12.

Suppress

The SU output can be suppressed in groups of 4 or 2 bits.

These groups are:

SU Bits Ø 1 2 3
4 5 6 7
8 9
10 11
12 13
14 15

Thus, six suppress signals are available, named according to the bit groups affected (eg. Suppress 4-7, Suppress 12-13 and so on). Suppressed bits (groups) are forced to zeros.

The six suppress signals are provided by the Suppress Control (Figure 4-1Ø1, FEMDM 2Ø2Ø CPU). They are generated depending on the basic operation to be executed and correspond with the cycle periods (T1 to T8).

Test Sign

- During SDS the INT SU bits 12 to 15 are tested for representing a sign (/A/ to /F/).
- An invalid sign (/Ø/ to /9/) forces trap request 2 by turning on the Data Error latch.
- The Data Error Latch is sensed and reset by a CPU SENSE /16/.

During SDS (cycle Ø if DD type, cycle 1 if DX type) the INT SU Bits 12 to 15 are tested for representing a valid sign (any hexadecimal value from /A/ to /F/).

An invalid sign (any value /Ø/ to /9/) prevents activating the signal Bits 12-15 Sign A to F. This signal inactive during cycle Ø or cycle 1 if the SDS instruction is in process turns on the Data Error latch at T 6 (delta cycle 1 or 2 overlaps with cycle Ø or 1).

The active Data Error latch forces a trap request 2. Trap request 2 forces a circuit controlled branch (change of LS zone addressing) to a micro-program routine. During this routine a SENSE /16/ investigates for the trap request cause. The active Data Error latch sets the bit 11 in the sensed byte. During the SENSE /16/ the Data Error latch is reset at T8. Beside SENSE /16/ the Data Error latch can be reset by stopping the CPU (Process drops).

Test Packed Byte

During packed decimal operations (AP, SP, ZAP, PPC) the two packed decimal digits provided in the loworder byte (INT SU bits

8 to 11 and 12 to 15) are investigated to be valid (hexadecimal values /Ø/ to /9/). If the two digits are valid the signals Bits 8 to 11 Sign A - F and Bits 12 to 15 Sign A - F are inactive.

When one of these signals is active during cycle Ø (DX or XX type), cycle 1 (DD or XD type), or cycle 2 the Data Error latch turns on and forces a trap request 2.

Data Bus Output

During a CTRL instruction the INT SU bits P1 to 15, which derive either from TDR highorder byte (after shift by 8) or from the TDR loworder byte (No shift) are provided to the Data Bus (P1 to 15). The Data Bus provides the CTRL data to the attachments, channels or control features. Note that the INT SU bits P1 to 15 are provided to the Data Bus as long as the CTRL instruction is placed in the OP REG. (T1, cycle Ø until T1, cycle Ø of the next micro-instruction) .

Shift Unit Parity

- To ensure correct SU input parity the INT SU bits PØ to 15 are checked (SU check)
- Shift by 2 or 4 and Suppress require changing of the input parity when an odd number of turned on bits in the highorder or loworder byte are affected.
- Standardizing Sign /E/ (plus) always changes the parity of the loworder byte.

SU shift or suppress functions affect only data bits (Ø to 7, 8 to 15). The SU parity is predicted in the ALU parity correction circuits depending on the executed function (Shift Suppress). The SU parity is used for SU display (DR - E, S, T, R) and to determine the ALU parity.

Remember the general parity rule:
Changing the status (on to off, off to on) of an odd number of bits in a byte requires changing the parity bit.

The status of bits can be changed in the SU either by shifting or by suppressing. For Normalize Sign operations only, standardizing the sign /E/ affects the parity (P1).

The INT SU bits PØ to 7 and P1 to 15 (Eight Shift result) are parity checked to ensure correct SU input parity. If the INT SU bits are of correct parity the signals Shift Unit Ø - 7 odd parity and Shift Unit 8 - 15 odd parity are active (negative).

Incorrect (even) parity of any INT SU bytes turns on the SU Check latch except for CPU SENSE instructions, MANOP's, or ICPL SENSES.

The SU Check latch on causes stopping the CPU one cycle after the cycle in which the check occured. The active SU Check latch illuminates the SU Check indicator on the CE console.

During TRBS, cycle 0, the signal which indicates correct parity for the highorder byte (Ø to 7) is forced to prevent SU check caused by the transferred FDR bits 8 to 11

(forced to zeros by Invert Switch) which are only needed to correct the Invert Switch parity P1.

The XOR-logic combinations which provide the SU check signals (Shift Unit ø - 7 or 8 - 15 odd parity) also provide signals which define the number of turned on bits of specified bit groups.

These signals are:

highorder INT SU byte	loworder INT SU byte
Odd SU Bits ø-3	Odd SU Bits 8-11
Odd SU Bits 4-7	Odd SU Bits 12-15
Odd SU Bits 6-7	Odd SU Bits 8- 9
Odd SU Bits ø-3 x or 8-11	Odd SU Bits 4- 7 x or 12-15
Odd SU Bits ø-1 x or 8- 9	Odd SU Bits 6-7 x or 14-15

This signals are used according to a shift by 2 or 4 or a suppress function to decide wether the parity must be changed (odd number of turned on bits affected) or not (even number of turned on bits affected).

ARITHMETIC - LOGIC UNIT

- The ALU performs ADD, AND, OR, or XOR using the SU and invert Switch halfwords.
- XOR is the basic operation the ALU is prepared for if no Adder, AND, or OR gate is active.
- The ALU consists of 16 ALU cells which perform the ALU operations for one bit position.
- ALU Carry bit 8 indicates a carry for byte operations, while ALU carry bit 0 indicates a carry for binary halfword operations.
- An ALU carry turns on the Carry and AUX Carry latch.
- AUX Carry Latch on causes Additional Carry into the unit bit cell (15) during the next Add operation.
- A carry out of the high order or low-order byte during logical operations is fed back to the units position of the corresponding byte for parity corrections only.
- During arithmetic or logical operations a odd number of carries out of the bit positions within a byte changes the parity predicted by the SU and Invert Switch parity.
- The ALU output (Sum bits) is parity checked to ensure correct ALU operations.
- Depending upon the ALU result the signals ALU Zero or ALU Not Zero turn on.
- For binary, logical and packed decimal instructions the four condition code latches are set according to the ALU result condition.

The Arithmetic-Logic Unit (ALU) can execute four different functions using the half-

word operands provided by the SU output and the Invert Switch output.
The four functions are :

Adding (Add)
ANDing (AND)
ORing (OR)
Exclusive ORing (XOR)

The function to be performed is defined by a control gate generated in the ALU control circuits.

Only three gates, Adder gate, AND gate, and OR gate, are generated depending upon the basic operations to be executed. Exclusive ORing is performed when no gate is active (general condition). The gates are timed by cycle periods (T1 to T8).

Basically the ALU consists of 16 similar circuits named ALU Cells 0 to 15. Since the ALU cells operate similar it is only necessary to understand the function of one of them.
An ALU cell performs the same functions (Add, AND, OR, XOR) as the complete ALU does, but on bit basis.
The three inputs of cell are :

SU bit
Invert Switch bit
Carry

The two outputs of a cell are :

Sum bit
Carry

The ALU Cell operating principles for Add, AND, OR and XOR are shown in figure 4-111, 2020 CPU FEMDM.
The ALU cells are chained to process halfwords or bytes by connecting the carry output of a cell to the carry input of the next highorder cell.

Take into consideration that an ALU cell can provide a carry also during AND, OR, or XOR, however since the Adder gate is inactive the next cell does not accept this carry.
The carries out of a byte or an halfword are provided by the carry output of ALU cell 8 or ALU cell 0 respectively. These carries are defined as ALU Carry bits 8 or 0. ALU Carry bit 8 is used during packed decimal instructions (AP, SP, ZAP, PPC), while ALU Carry bit 0 is used during binary halfword instructions (AHSC, SHSC).
The ALU Carry bit 8 and 12 are used to control six correct functions, which are necessary to correct hexadecimal values (/A/ to /F/) during packed decimal operations to obtain valid packed decimal values (/0/ to /9/).
The carry input of the ALU Cell 15 is activated by the Additional Carry.
The Additional Carry occurs as complement correction (inverted data plus one) during the subtract operations of SH or CLC (Inject one to ALU) or caused by the active AUXiliary Carry latch. The AUX Carry latch turns on at T2 (Reset/Set) during cycle 0 or 1 when the Carry latch has been turned on either by ALU Carry 8 or 0 during the previous instruction or operation (repetitions if ALC).

During the logical operations (AND, OR, XOR ; Adder gate inactive), the Additional Carry as well as the Carry into Position 7 are activated by feeding back the carries

(ALU Carry 8 or 0) to the unit position of the loworder or highorder byte. Since the Adder gate is inactive, the feed back carries do not influence the result (Sum bits) but they are necessary for parity correction during logical operations.

During the binary arithmetic instructions (AH, AHSC, SH, SHSC), the carry into the sign position (ALU Carry bit 1) are used to recognize a Binary Overflow. A Binary Overflow activates Condition Code (CC) latch 3.

The parity bits P0 and P1 of the ALU result bytes (Sum bits 0 to 7 and 8 to 15) are predicted according to the Invert Switch (display I, P, U, L) and SU parity (display E, S, T, R ; INT SU parity changed according to the performed SU functions).

The predicted ALU parity is changed when the signals Change Parity Sum Bits 9 to 15 or 1 to 7 are active. The signals are activated if any ALU operation changes the status of an odd number of bits in one of the ALU bytes. The status change information is provided by the carry output of the ALU cells. Thus, the carry outputs are connected to a XOR combination (one for each byte), the output of which provide the Change parity signal if the status changes are odd.

At the ALU output the signals Sum Bits 0 to 7 Even and Sum Bits 8 to 15 Even are activated depending upon the number of turned on bits in the highorder or loworder result byte. The Sum Bits Even signals are compared with the corresponding corrected parity bits and the ALU Check latch turns on if even parity is recognized. The active ALU Check latch illuminates the ALU Check indicator on the CE console and stops the CPU after the next cycle.

During CPU SENSES (provide SENSE data without parity) the Prevent ALU and SU Check signal is active. This signal blocks the parity bits provided by the correction circuits and allows generation of the ALU bits P0 and P1 depending upon the Sum Bits Even signals only. Thus, the ALU operations are not checked, since the parity is generated depending upon the result and the parity is correct even if the ALU as well as the SU and Invert Switch circuits function erroneously. The parity bits of the SENSE data in FDR are corrected retroactive depending on the generated parity bits at the ALU output. Corrections are performed by either the signals Force FDR bit P0 or P1 or the signals Turn off FDR bit P0 or P1 active.

The complete ALU result is displayed on the CE console when the CE Display Select switch is set to ALU.

Depending upon the present ALU result the ALU Zero or ALU Not Zero signal is generated.

The halfword at ALU output is provided to the input of SAR LS, FDR, and Inhibit.

The Condition Code latches (CC latches 0 to 3) are set according to the result condition either unconditional (AP, SP, ZAP, PPC) or controlled by the CC bit (OPREG bit 6) in the micro instruction.
(AH, AHSC, SH, SHSC, AND, OR, XOR, CLC).

The condition code is accessible by a SENSE instruction.

Six Correction Circuits

- Six Correction effects only the two four-bit digits of the loworder ALU result byte.

- Six Correction can be considered as a binary adder which adds the constant '-6' to the digit to be corrected.

- Six Correction influences the ALU parity correction.

The Six Correction Circuits are activated for packed decimal instructions (AP, SP, ZAP, PPC), however, for the subtract-type instructions (SP, PPC) only if the Adder gate is active (cycle 2 or 3).

The Six Correction affects the two half-bytes of the loworder ALU byte. Each half byte is corrected by a separate circuit.
The Six Correction circuit can be considered as a four bit binary adder which adds the constant '-6' to the ALU result (two hexa-decimal digits of the loworder byte). Six Correction is performed when no carry occurs out of the highorder bit of the corresponding halfbyte (No ALU Carry bit 12 or 8).

The constant '-6' expressed as a half byte binary number is 1010. This binary number is added to the halfbyte value represented by Sum Bits 12 to 15 or 8 to 11. Since the loworder bit of the constant is zero and adding of zero does not influence the input bit, the loworder bits (15 and 11) are not affected by the Six Correction.
A performed six correction influences the parity of the loworder ALU byte (Change Parity Sum Bits 9 to 15).

ALU Parity Prediction and Correction

- The ALU result parity is predicted depending upon the parity of both ALU input operands.

- ALU operations which change the number of turned on bits odd, force a change of the predicted parity.

The ALU operations ADD, AND, OR, or XOR affect the data bits 0 to 7 (highorder byte) and 8 to 15 (loworder byte). The number of turned on data bits in both ALU result bytes (Sum bits 0 to 7 and 8 to 15) can be even or odd. Thus, the Sum bits P0 and P1 must be turned on or off to provide always odd byte parity (odd number of turned on Sum bits including parity bit).
The ALU result bytes are tested for proper parity. Even parity in the highorder or loworder byte turns on the ALU Check latch and the CPU stops (Process Check).
To allow testing of ALU operations the Sum parity bits P0 and P1 are predicted. That means, the parity result is generated according to the parity of the ALU input operands and predetermines the odd or even number of turned on data bits in the highorder or loworder result byte.
The predicted Sum parity bits are correct

when all turned on bits of the input operands occur in the ALU result.

Example: 1 1 1 1 0 0 0 0 Operand 1
0 0 0 0 1 1 1 1 Operand 2

1 1 1 1 1 1 1 1 Result if ADD

Each operand provides four turned on data bits. To provide odd operand parity, both parity bits must be on. The parity bit on indicates an even number (four of turned on data bits in the corresponding byte. The total number of turned on bits of both operand bytes (even or odd) is specified in Table 2-1:

Since the Sum parity is predicted upon the total number of input data bits, the prediction has to be changed (corrected) when an odd number of turned on data bits is lost caused by the executed ALU function (Add, AND, OR, XOR)

Examples:

ADD 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1

Result 0 0 0 0 0 0 1 0

The total number of input data bits is two (even) ; thus, the predicted parity bit has been turned on to provide odd parity. However, adding the two turned on bits causes a zero in the corresponding result position. A carry, which can be assumed as one turned on bit, occurs and turns on the bit in the next higher position. One turned on input bit (odd number) has been lost and the predicted parity bit (on) has to be changed (off).

During Adding a carry provided to an ALU cell indicates the one input bit has been lost. The predicted parity bit must be changed when an odd number of bit carries occurs.

AND 0 1 0 1 0 1 0 0
0 0 0 0 0 1 0 0

Result 0 0 0 0 0 1 0 0

The total number of turned on input bits is even (odd plus odd). The predicted parity bit is on. However, an odd number of turned input bits is lost (three) and the predicted parity bit has to be changed.

OR 0 1 0 1 0 1 0 0
0 0 0 0 0 1 0 0

Result 0 1 0 1 0 1 0 0

The predicted parity bit is on (even number of input bits). One of the opposite turned on bits is lost (odd number). The predicted

Table 2-1 Turned On Bits of Even and Odd Operand Bytes

Operand 1	Operand 2	Total of turned on data bits
P=1 (even)	P=1 (even)	Even (even plus even = even)
P=0 (odd)	P=0 (odd)	Even (odd plus odd = even)
P=1 (even)	P=0 (odd)	Odd (even plus odd = odd)
P=0 (odd)	P=1 (even)	Odd (odd plus even = odd)

parity bit has to be changed.

XOR	1 0 0 1	1 0 0 1
	0 0 0 0	1 0 0 1
Result	1 0 0 1	0 0 0 0

The predicted parity bit is on (even number of turned on input bits).
No change of the predicted parity is required since four (even) bits are lost.
Note that for XOR the predicted parity is always correct since XOR operations affect only even numbers of turned on input bits.
During Add operations changing of the total number of input bits is indicated by a carry into the next higher ALU cell. Note that a carry into the unit position (cell 15 or 7) of one of the ALU bytes (Additional Carry, Carry into Position 7) also affects the number of turned on bits of the corresponding byte. The ALU cell circuits are designed to activate the carry output (input of the next cell) even for the logical ALU functions (AND, OR), when bits are lost. A logical carry is not accepted by the next ALU cell
For XORing no carry is generated since this function always affects an even number of turned on bits and no correction of the predicted parity bits is necessary.
Figure 2-19 shows the carry conditions of the four possible ALU functions.
The predicted parity bit must be corrected when an odd number of bit carries occur within a byte. To detect an odd number of carries the eight carry inputs within an ALU byte are connected to a XOR combination.
The XOR combination provides the Change Parity signal (Change Parity Sum Bits 9 to 15, Change Parity Sum Bits 1 to 7).

ALU Cell Input						
SU Bit	Invert Sw Bit	Carry	ADD	AND	OR	XOR
ϕ	ϕ		—	—	—	—
ϕ	⊥		—	CARRY	—	—
⊥	ϕ		—	CARRY	—	—
⊥	⊥		CARRY	CARRY	CARRY	—
ϕ	ϕ	⊥	—			
ϕ	⊥	⊥	CARRY			
⊥	ϕ	⊥	CARRY			
⊥	⊥	⊥	CARRY			

Note: A carry always indicates that one turned on input bit has been lost.

Figure 2-19 ALU Cell Carry Conditions

Note that for logical ALU functions (AND, OR) the carries out of the highorder ALU cell of each byte are fed back to the unit positions. This feed back is necessary to recognize the carries of all eight cells for parity correction.

Correction of the Predicted Parity if Six Correction

Six Correction is required for packed decimal operations. The constant '-6' (1010) is added to the ALU result. Six Correction affects only the two four bit (hexadecimal) digits in the loworder byte (8 to 11 and 12 to 15).

Correction of the predicted parity bit (P1) is necessary when the six correction changes the number of turned on Sum bits uneven. The total number of turned on Sum bits is changed from odd to even or even to odd only when adding the constant 1010 (-6) to the hexadecimal values /2/, /3/, /A/ and /B/.

Adding the constant 1010 affects only the three highorder bits of the hexadecimal digit since the loworder bit of the constant is zero. Thus, six correction can be considered as an addition of three bit digits (constant 101). The three highorder bits of /2/ and /3/ are 001, while the three highorder bits of /A/ and /B/ are 101.

Both three bit digits show the bit combination identifies the only hexadecimal digits which cause an odd change of the number of turned on Sum bits when adding the constant. Thus, a signal is generated depending upon 01, which influences the Change Parity XOR combination of the loworder ALU byte.

Condition Code Latches

- The Condition Code Latches are set to reflect the ALU result conditions.
- For binary and logical micro-instruction, the condition code is set when the CC bit in the instruction is on.
- The condition code is set unconditionally for packed decimal micro-instructions.

The Condition Code (CC) latches are shown in figure 4-113 , 2020 CPU FEMDM.

The CC Latches are set during Binary, Logical and Packed Decimal micro-instructions (CC-instructions) according to the final operation result (ALU result).

The saved result condition can be tested by SENSE /10/. A turned on CC latch (0 to 3) turns on the corresponding bit (8 to 11) in the sensed data byte set into FDR.

Beside CC-instruction the CC latches are set by CTRL /10/ controlled by the INT SU bits 12 to 15 turned on.

Set CC and Carry

The Set CC and Carry signal, active for cycle 2, cycle 3, or delta cycle 0, is generated according to the type of the CC-instruction (DD, DK, XD, XX) and specifies the cycle (delta cycle) during which the

final result is present at ALU output.

Allow CC Setting

The second condition, necessary to activate the CC latches, is the Allow CC Setting Latch turned on. This latch is set at T7 (reset at every T6) either for packed decimal micro-instructions (AP, SP, ZAP, PPC) or for binary or logical instructions (AH, AHSC, SH, SHSC, AND, OR, XOR, CLC) but for the last instructions only if the CC bit (bit 6) in the instruction is on (1).

Set CC and Carry and Allow CC Setting (Latch) active enable turning on the CC latches at delta T8 (reset/set).

Allow CC Setting is prevented for CC-instructions with ALC when the final condition is saved in the CC Latches (e.g., AP, ALC, result not zero ; CC1 active) while the ALC operation is still not terminated (field length not decreased below zero).

I/O BUS OUT AND IN

- The I/O Bus connects the CPU with all present attachments.

The I/O control units are attached to the CPU by bus lines.

The cycle steal control lines, and the bus lines are the only way to send or receive informations from the I/O control units.

The bus lines can be divided into three categories :

1. Address Bus
2. Data Bus
3. Control bus lines

There are two types of bus lines specified by the bus termination in the attachments. Bus lines terminated by a line sense amplifier (LSA) are defined as type A. Bus lines type A are used to provide CPU information to the attachments with a minimum of delay time. This bus type cannot use the sent informations from the attachments to the CPU (e.g., Address Bus).
Type B bus lines are those which accept informations either in the attachments or in the CPU (e.g., Data Bus). These type B bus lines are terminated by an Inverter-OR-Inverter Logic which allows activating of the bus lines even in the attachments. Since the Inverter - OR - Invert Logics of all attachments are connected in series the bus signal is delayed.
The CPU (I/O bus entry) can be considered as an additional attachment in respect to the bus termination.

I/O BUS OUT

A comprehensive representation of all bus lines leaving CPU is given in figure 2-20
The following sections describe the I/O Bus output control shown in the FEMDM, 2020 CPU, Figure 4-120. This figure shows the Address and Data Bus control as well as the generation of the control strobes for SENSE and CTRL functions. The other control Lines leaving CPU are shown in the circuits where they are generated.

Address Bus Out

- The Address Bus provides the Device Address for CTRL or SENSE.
- A Device Address can be applied from the micro-instruction or from switches.
- For I/O Display the Device Address is set up in the CE Select switches 1 and 2.

The Address Bus, consisting of eight single bus lines (bits 8 to 15), provides the Device Addresses to the attachments for SENSE and CTRL.
A device address consists of two hexadecimal digits (one byte, eight bits) which are provided either by the device address field in a SENSE or CTRL micro-instruction (OP REG bits 8 to 15) or, especially during

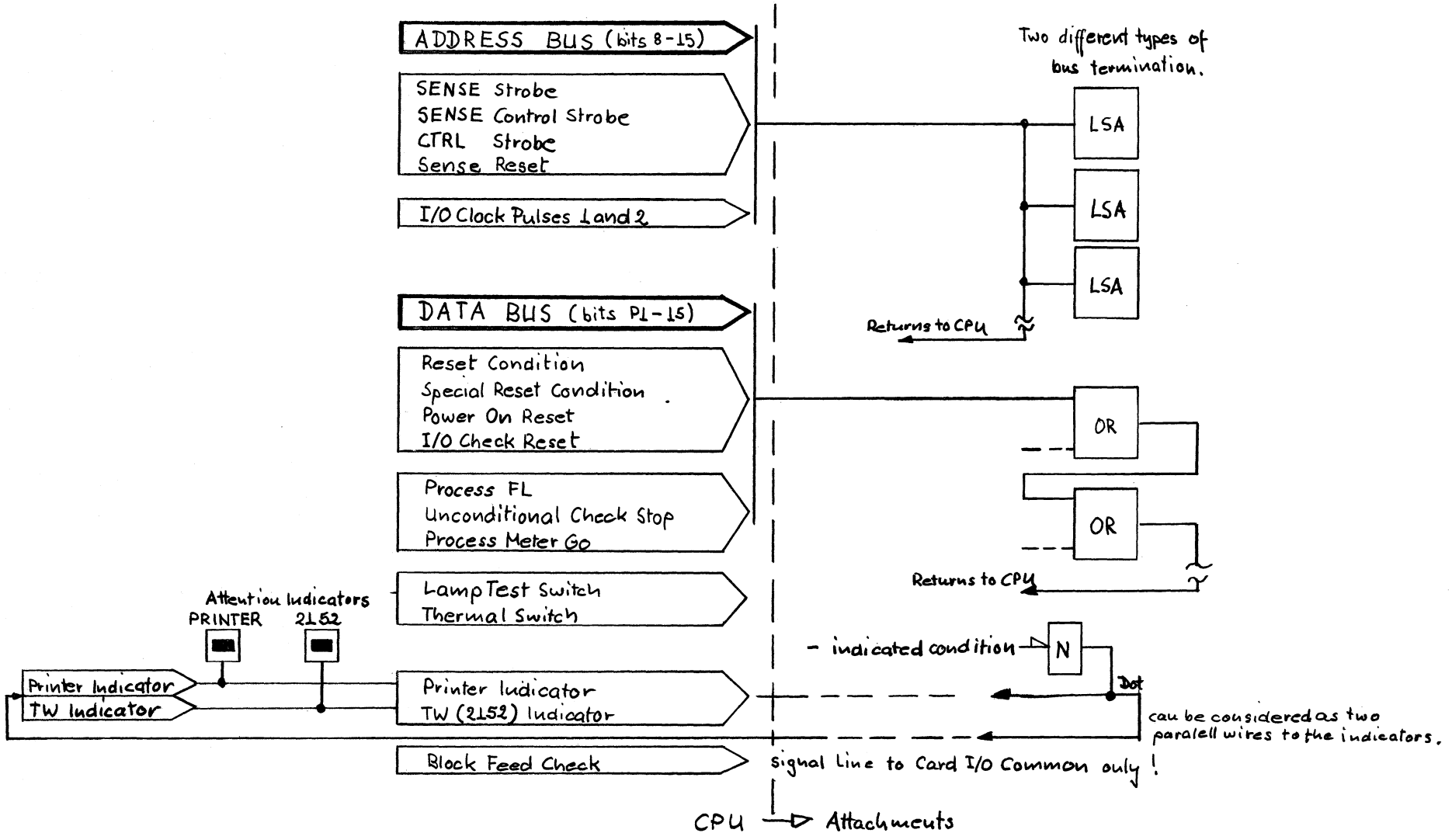


Figure 2-20 Output Bus Lines

ICPL, by the values set up in the Data switches 1 and 2 on the customer console (also placed into OP REG 8 to 15).
If no SENSE/CTRL and no ICPL operation is being processed the Address Bus is activated according to the values set up in the CE Select switches 1 and 2.
The device address , consisting of an high-order and loworder hexadecimal digit can be interpreted as follows :
The highorder digit (8 to 11) defines the called attachment. Since the number of possible attachments is limited, only the values /0/ to /A/ are valid.
The loworder digit (12 to 15) can vary between /0/ and /F/ without changing the high-order digit. Thus, it can be said, that each highorder digit (attachment) address can be extended by 16 sub-addresses.
However, each of the 16 addresses (device addresses) relating to one attachment allow controlling or sensing of eight different functions or conditions in the specified attachment (highorder digit) by means of the eight Data Bus bits. Thus, each attachment address (highorder digit) provides the possibility to test (SENSE) or to control (CTRL) 128 (16 times 8) different attachment functions.
The assignment of the possible highorder digits in relation to the attachments is given in figure 2 - 21.

OP REG to Address Bus
The OP REG bits 8 to 15 are gated to the Address Bus when the OP REG to Address Bus signal is active. This signal is activated when either a SENSE or CTRL microinstruction (OP R Decode E or F) is placed in the OP REG or when the ICPL Latch is turned on during Initial Control Program Load.Note that for ICPL the device address has to be /20/ and the Data switches must be set up accordingly.

I/O Display Address Out
The contents of CE Select switches 1 and 2 are used as Device Address (gated to the bus) when the I/O Display Address Out signal is active. The I/O Display Address Out signal is active during T4 to T7 when no SENSE or CTRL function is required (no OP REG to Address Bus signal active). Note that this signal is active even if the CPU has been stopped and allows the CE to display (CE console) I/O sense information by selecting any required device address using the CE Select switches 1 and 2. Depending upon the timing signal T4 to T7, the signals I/O Display Address Out and OP REG to Address Bus are generated alternating. [redacted]. The undefined address which derives from the OP REG bit 8 to 15 is ineffective since the I/O display functions are timed by T4 to T7 only.

Data Bus Out

- During CTRL operations the Data Bus carries the control bits.
- During all CPU operations other than CTRL's and during CPU stop the Data Bus parity bit P1 is forced.

Highorder device address digit	SENSE	CTRL
/0/	Test for Trap Request Source (I/O)	Reset or Set PL REG (CPU)
/1/	CPU	
/2/	Serial Read/Punch 2501,2520/2560, 1442 (Card I/O Common)	
/3/	2520 / 2560	
/4/	1403 / 2203	
/5/	Reserved	
/6/	IOC	
/7/	Storage Control Feature	
/8/	Reserved	
/9/ (/0...3/) (/9...F/)	Typewriter SIOC	
/A/	BSC A	

Figure 2-~~11~~ 21 Assignment of the Highorder Device Address Digit

The only operations which require to send out data from the CPU to the attachments are the CTRL micro-instructions and ICPL. However, during ICPL the CTRL functions are performed during cycle 0 and 1 only.

The OPR Decode F signal, generated depending upon the four highorder bits of the CTRL instruction in the OP REG, gates the INT SU bits 8 to 15 to the Data Bus. The parity bit P1 is provided by the SU Parity Correction circuits and is similar to that bit P1 displayed in Data Register (DR-) U on the customer console.

During ICPL, cycle 0 and 1 (ICPL CTRL) the Data Bus bits are forced by circuits as shown in figure 2-**22**.

Since the Data switches are set to /20/, the forced Data Bus bits control the same attachment functions as for a CTRL /20/ micro-instruction.

Data Bus Parity

For CTRL micro-instructions the bus parity bit P1 is provided by the SU parity bit P1, which is similar to the INT SU bit P1, since during CTRL no shift unit functions are performed.

During ICPL CTRL the parity bit P1 is also forced by circuits according to the generated bit pattern.

During all other operations (also CPU stop) the Data Bus bit P1 is forced unconditionally to provide always correct parity on the Data Bus.

Since the Data Bus is also used to transport sensed data from the attachments to the CPU (see I/O BUS IN), the forced Data Bus bit P1 has to be changed according to the parity of the sensed data byte. The principles of this parity change is given in figure 2 -**23**.

Control Strobes for SENSE and CTRL

- During SENSE operations, functions within the attachments are controlled by the SENSE Strobe (T3) and by the SENSE Control Strobe (delta T8).

- During CTRL operations, functions within the attachments are controlled by the Sense Reset pulse (T3) and by the CTRL Strobe (delta T8).

Four control signals (strobes) are generated within the CPU which are sent out to the attachments via bus lines. These signals are timing signals which control the SENSE or CTRL functions in the attachments in relation to CPU processing time.

These four signals are:

Sense Reset	(CTRL only)
CTRL Strobe	(CTRL only)
SENSE Strobe	(SENSE only)
SENSE Control Strobe	(SENSE only)

These control signals are gated to the attachments by Process not CS (CPU running but not cycle stealing) and the active Allow Strobe signal.

Allow Strobe is active during SENSE, CTRL, and ICPL only. However, Allow Strobe is prevented during cycle 1 of indirect addressed SENSE or CTRL instruction. During

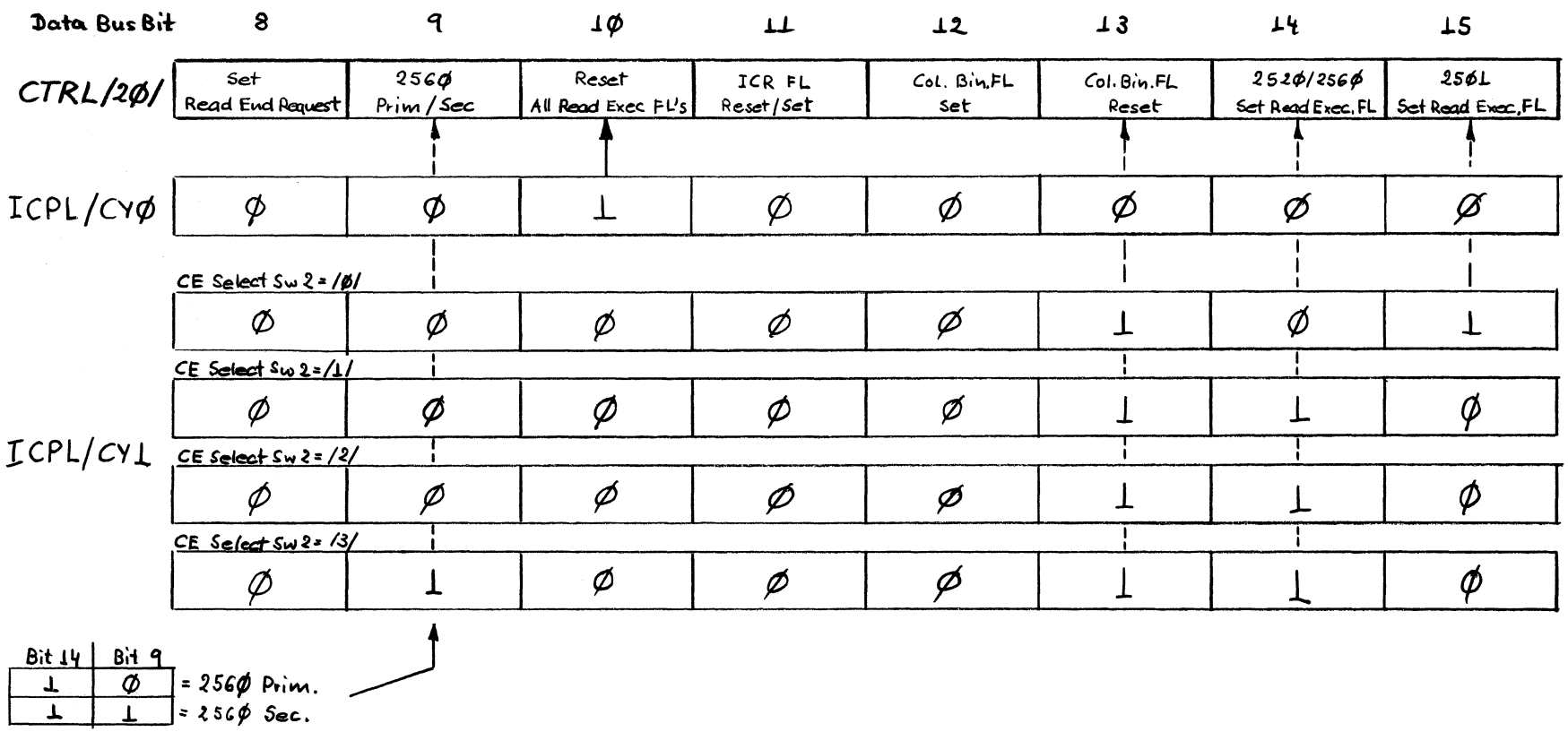


Figure 2-~~Ø~~22 Device Address /2Ø/ Functions during ICPL-CTRL

ATTACHMENT

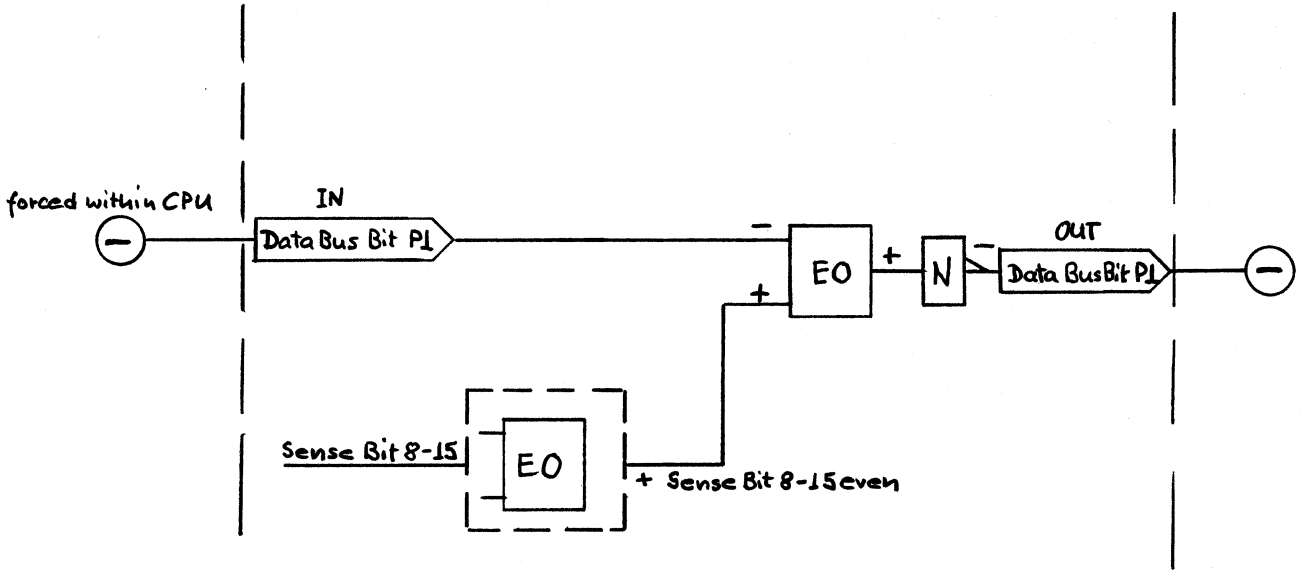


Figure 2-28 Principles of Data Bus Parity Changing during I/O SENSE

this cycle 1 the IAR updating of indirect addressed (ALC) I/O instructions is performed.

In addition, the Allow Strobe condition is inhibited during indirect addressed CTRL instructions when delta CY2 overlaps with CY 0. No Allow Strobe during the overlapping period prevents the CTRL strobe (delta T8) during CY 0. This is necessary since the control bits are read out and set onto the Data Bus not earlier than during CY 2. Thus, the CTRL strobe is prevented and occurs at the end (delta T8) of CY 2 only.

SENSE Strobe and SENSE Control Strobe

Both strobes are generated for SENSE and ICPL CY 2 (ICPL SENSE) only. The SENSE Strobe, generated at T3 after the SENSE micro-instruction has been transferred into the OP REG (OPR Decode E, CY0, T1) or at T3 during ICPL SENSE (CY 2), allows setting/resetting of the Sense Registers by sense data. The source of the sense data is defined by the device address on the Address Bus.

After the Sense Register has been set, the sense data is available on the Data Bus and, thus, also at CPU Data Bus Entry. This sense data is placed into FDR when the SENSE Control Strobe (delta T8) becomes active. Beside the Data Bus Entry Control, the SENSE Control Strobe is used within the attachments to perform reset functions (resets latches the conditions of which are just have been sensed).

The SENSE Strobe affects the Sense Registers of all present attachments, but sense data is provided to only one of them (Sense Register in the attachment specified by the highorder digit of the device address). All other Sense registers are set to zero since no sense data is provided.
Sense data remains in the Sense Register until the next SENSE or CTRL instruction is executed.

Sense Reset and CTRL Strobe

Both signals are generated for CTRL and ICPL CTRL (CY 0 and CY 1) only.
The Sense Reset signal, generated at T3, resets all Sense Registers in the attachments to clear the Data Bus for the control bits.
The CTRL Strobe, generated at delta T8, is the timing signal which controls the functions in the attachment defined by the device address and the control bits on the Data Bus.

I/O Display

- I/O Display allows to select and display any SENSE data source in the attachment during CPU run as well as during CPU stop.

The I/O Display allows the CE to select and display any required I/O sense data source by a device address set up in the CE Select switches 1 and 2. This I/O Display functions when the CPU stops as well as during CPU run, but not during SENSE or CTRL. The Display Sense Strobe generated at T6, activates the SENSE Strobe which controls the Sense Registers in the attachments.
Since the Display Sense Strobe occurs for each cycle (T6 to T6) the sense data source is tested in a rate which corresponds with the CPU processing speed (every 2 or 3.6 microseconds).
The I/O Display Latch, which is only reset for SENSE, CTRL, ICPL and CPU LOG in, gates the I/O Bus Entry (Data Bus entry) to the display (P1 to 15) on the CE console, but only when the CE Display Select Switch is set to I/O Display.
The sense information, which illuminates the indicators can also be used to generate the Equal Synch signal (Test Hub) or, if the CE Compare Stop switch is turned on, to stop CPU depending upon I/O conditions.
For Equal Synch or Compare Stop the sensed data byte is compared against the byte setup in the CE Select switches 3 and 4.

I/O Bus Powering

The bus lines type A (LSA termination) require additional powering. For this powering, the bus lines enter the CPU and leave it again after powering. This amplification divides the complete bus into halves, which are identified with the letters A (bus before powering) and B (bus after powering).

I/O BUS IN

A comprehensive representation of all bus lines entering the CPU is given in figure 2 -24 . The following sections describe the I/O Bus input shown in the FEMDM, 2020 CPU, figure 4-121. This figure shows the Data and Address Bus input controls as well as the CPU SENSE circuits.

Bus in Control

- Data and Address Bus are set into FDR by I/O Bus to FDR signals.
- I/O SENSE, direct addressing, places the Data Bus Information directly into the loworder byte of the specified LS register.

Address and Data Bus are gated into FDR by the signals I/O Bus to FDR 0-7 (Address Bus) and I/O Bus to FDR 8-15 (Data Bus). These signals are generated for CPU SENSES (I/O Address 1 x, MANOP Inbus to FDR) at T4. During I/O SENSES (Device Addresses 0 x, 2 x to A x) and all CTRL operations (CPU, I/O ICPL) the Address Bus is gated into FDR at T4, but the Data Bus is accepted at delta T8. The later acceptance of Data Bus in relation to Address Bus (T4) is based on the Data Bus delay caused by the bus termination type B.
Especially for direct addressed I/O SENSES the Data Bus is gated simultaneously into FDR and into the loworder byte of the specified LS register at delta T8. To provide correct parity in the highorder byte of the LS register, the parity bit P0 is forced.

CPU SENSE

- CPU SENSES are specified by Device Addresses with the highorder hexadecimal digit = /1/.
- SENSES /14/ and /15/ are the only halfword SENSES.
- CPU SENSE data is applied without parity bits and, thus, the Prevent ALU and SU check signal is generated.

CPU SENSES are identified by Device Addresses (on Address Bus), the highorder hexadecimal digit of which is /1/ (Address Bus bits 8 to 11 = 0001). This condition causes the I/O Address 1 x signal.
The I/O Address 1 x signal specifies in connection with the Address Bus bits 13 to 15 any SENSE between /10/ and /16/.

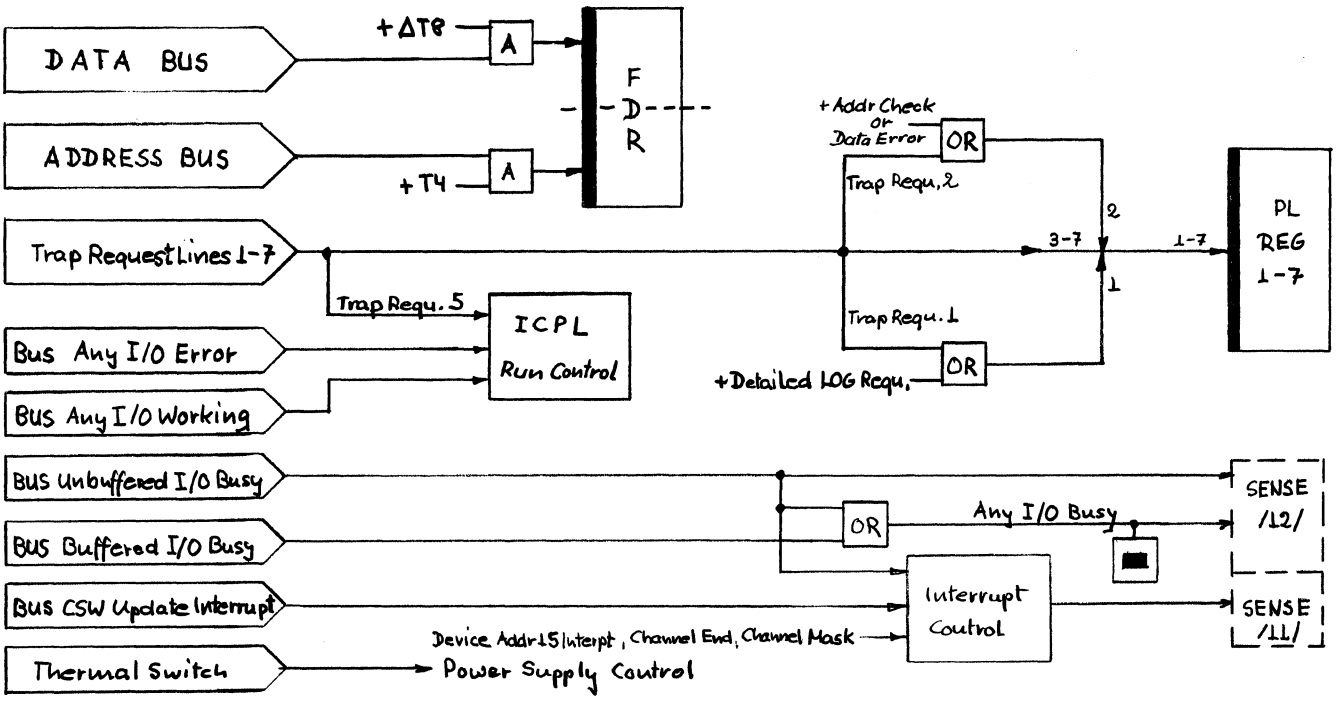


Figure 2-24 CPU Input Bus Lines

During MANOP's (Storage DASF, ICPL) the SENSES /13/, /14/ and /15/ (switch SENSE's) are forced by circuits. Generally SENSE data are of byte format. The only exception are CPU SENSES /14/ and /15/, which sense the four Address or CE Select switches at once. The sensed data of these two SENSE's is of halfword (two bytes) format. Since the Data Bus can only carry one byte the second (highorder) byte is set on the Address Bus. To allow use of the Address Bus for data transport during halfword SENSE's, the Address Bus input is blocked.

The Sense Bits 8 to 15 (Data Bus) can be provided by four separate circuits. These circuits are :

1. I/O Bus in (Data Bus entry)
2. CC and Carry Latches
3. Address Check (Modifier) - I/O LOG
4. CPU SENSE

The outputs (Sense bits) of the four circuits are dotted. To allow dotting of the Sense bit lines, the outputs of the circuits are forced to positive (Sense bit active) by the corresponding SENSE signal inactive.

Any CPU SENSE activates the signal Prevent ALU and SU Check since CPU data is sensed without parity bit.
This signal inactive indicates that the Sense bits 8 to 15 can be activated according to the Data Bus bits. This condition is defined as I/O SENSE.

- However, the Sense bits are gated to FDR only for SENSE and CTRL.
- The Bus Check tests the Data Bus for correct parity.

I/O SENSE

- I/O SENSE can be considered as the common I/O Bus condition which is inhibited during CPU SENSE's only.
- I/O SENSE condition allows the Data Bus to enter CPU (activates the Sense bits 8 - 15).

The I/O SENSE condition is present during all operations, except CPU SENSE, and also if the CPU stops. During I/O SENSE condition, caused by not Prevent ALU and SU Check, the Sense bits 8 to 15 reflect the Data Bus bit pattern. However, the Sense bits enter FDR only during SENSE or CTRL when the signals I/O Bus to FDR 0-7 or 8-15 are active.

SPECIAL PURPOSE LATCHES

The special purpose latches are shown in FEMDM, 2020 CPU, figure 4-130.
These latches are activated by CPU CTRL micro-instruction (CTRL /10/ and /11/) during micro program routines only.

Detailed LOG Request Latch

This latch is turned on by micro program when a data error occurs during a serial read or punch operation. The latch active forces Trap Request 1 and is tested by a SENSE /16/.

ASCII Latch

- This latch on indicates that the CPU has to provide data in the ASCII code.

The ASCII Latch is turned on by micro program when the ASCII-Code is used instead of the EBCDI-Code. The active ASCII latch controls the Normalize Sign circuits (SU) during the SDS micro-instructions, so that the positive or negative signs are normalized according to the ASCII-code.

The latch condition is displayed on the CE console (P0), when the CE Display/Compare Select Switch is set to OCU 2.

Interrupt Control

- The latches Channel End, Device Address 15 Interrupt, and Channel Mask Control the interrupts.

- Interrupts are enabled only when the Channel Mask Latch is on.

An interrupt is a branch to special program routine (exchange PSW) which is performed when a I/O operation has been terminated. An interrupt can be caused by Channel End, Device Address 15 Interrupt or by CSW Update Interrupt. The CSW Update Interrupt condition derives via a bus line from the corresponding control feature while Channel End or Device Address 15 Interrupt conditions are recognized by micro-program, which turns on the corresponding latches.
The Channel Mask Latch, which is also set and reset by micro-program, enables or disables the interrupt condition.
The interrupt condition is tested by a SENSE /11/ within micro program routines.

CPU CYCLE STEAL CONTROL

The CPU Cycle Steal Control circuits, shown in FEMDM, 2020 CPU, figure 5-190, are activated by interface signals from the CS devices.

CS devices are : Storage Control feature, IOC, and BSCA. The CS devices are numbered in ascending sequence from 1 to 4 (up to four CS devices can be attached). The numbering sequence defines the control priority of the four possible device requests. A device request 1 has the highest priority, while a request from device 4 has the lowest priority. Note that this priority sequence (1 to 4) is opposite to the priority sequence of the Trap requests (7 to 1).

The CPU Cycle Steal Control provides circuits which control the CPU functions during cycle stealing.

Since cycle steal operations are performed during one CPU cycle, only a few number of functions are executed.

These CPU functions are :

1. Transferring of Data Address from LS to SAR.

2. Incrementing or Decrementing the Data Address by 1 or 2 depending upon CS byte or halfword operations.

3. Decrementing the field length (located in LS) by 1 or 2 depending upon CS byte or halfword operations.

4. Storing a byte or halfword into main storage position defined by Data Address (CS Read).

5. Reading a byte or halfword out of Main storage position defined by Data Address (CS Write).

Cycle Steal operations interrupt any current micro-instruction for one CPU cycle. This interruption is allowed for each cycle during a micro-instruction.

During cycle stealing all important data and control informations of the current micro-instruction are frozen in and the interrupted operation is continued after the stolen cycle as normal.

A micro-instruction, interrupted by a CS request,can be considered to be prolonged in its execute time for one cycle (2 or 3.6 microseconds).

MANOP's prevent CPU CS operations.

CYCLE STEAL INTERFACE

The cycle steal interface (figure 2 - 25) connects the CPU with the Cycle Steal Units in the Storage Control feature, IOC, and BSCA.

Take into consideration that, beside the CS data bus and CS control bus lines, the above mentioned I/O control units are also controlled by the I/O Common Bus Lines (Address Bus, Data Bus, and associated control lines).

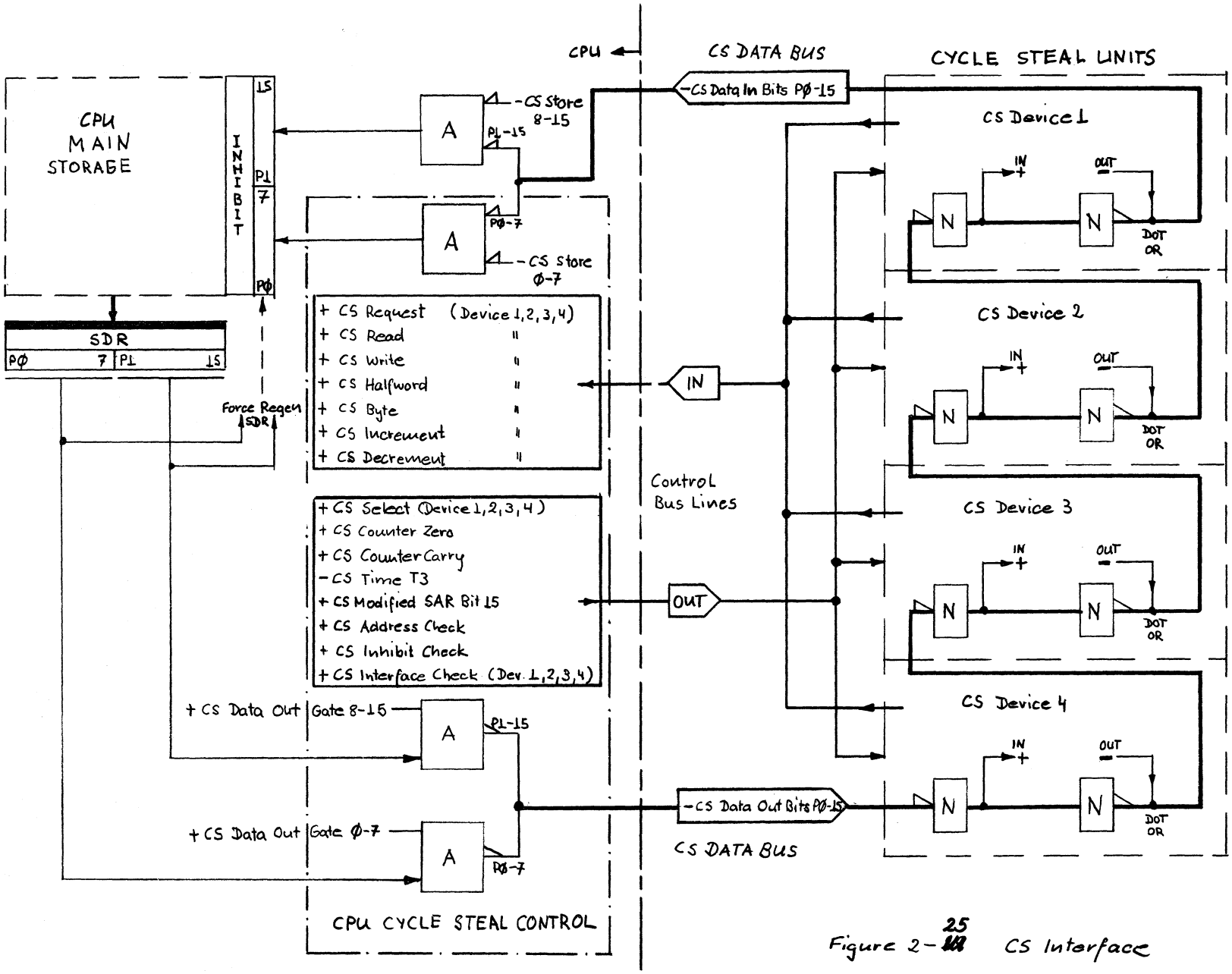


Figure 2-25 CS Interface

The CS Data Bus, able to transport one halfword (two bytes including parity bits), connects the cycle steal units directly with the Main Storage output (SDR) or input (Inhibit). The CS Data Bus is used either to provide data from the CPU to the cycle Steal units or vice versa. The data direction is controlled by the CS Data Out gates or by the CS store signals. Gates and signals are generated within the CPU CS control. Especially for byte transfer from or to the CS units the byte not affected is directly regenerated (Force Regen) by connecting the SDR to the Inhibit. The CS control bus lines entering CPU control the CPU CS control circuit according to the CS operation ordered by the CS unit.
The CS control bus lines leaving CPU inform the requesting CS unit about the present CPU conditions.

CS REQUEST

The CS Request Lines Device 1 to 4, activated within the CS units, turn on the CS Request Latches 1 to 4 gated by the Sense CS Request signal. This signal is identical with T2 and is also used to reset the CS Request Latches. Thus, a CS Request latch remains active for the period T2 to T1.
If more than one CS Request is active at the same time, all corresponding CS Request latches are set. During the following CS cycle only the highest priority request is effective and the request source (latch) of the corresponding CS unit is reset. At next T2 all CS Request latches are reset, but they are set again depending upon the still present lower priority requests.
During CS operations data address and field length are available in fixed LS registers. Each CS device has its own data address and field length register (figure 2-26). These registers are selected by X-addresses and Select LS-Y signals which are generated depending upon the active highest priority CS Request Latch.

Any CS Request

Any active CS Request Latch forces the main control signal Any CS Request. This signal is active from T3 until the end of T2. The offset for one T-period between the active CS Request Latch (T2 to T1) and the Any CS Request signal (T3 to T2) is achieved by controlling the signal during T2 depending upon the Cycle Steal latch. Any MAN-OP blocks Any CS Request.
Within the CPU CS control circuits Any CS Request controls resetting and setting of the control latches.
Within the CPU circuits Any CS Request controls the following main function and signals:

1. Addressing of fixed CS registers in Local Store (depending upon requesting device) at T5 and T7 , T8 and T2.

2. Generation of LS read signals LS to SAR (T5) and LS to MAR (T5, T8) as well as the LS Write signal at T7 and T2.

3. Blocking of delta cycle advancing (Long Time Clock) at T4.

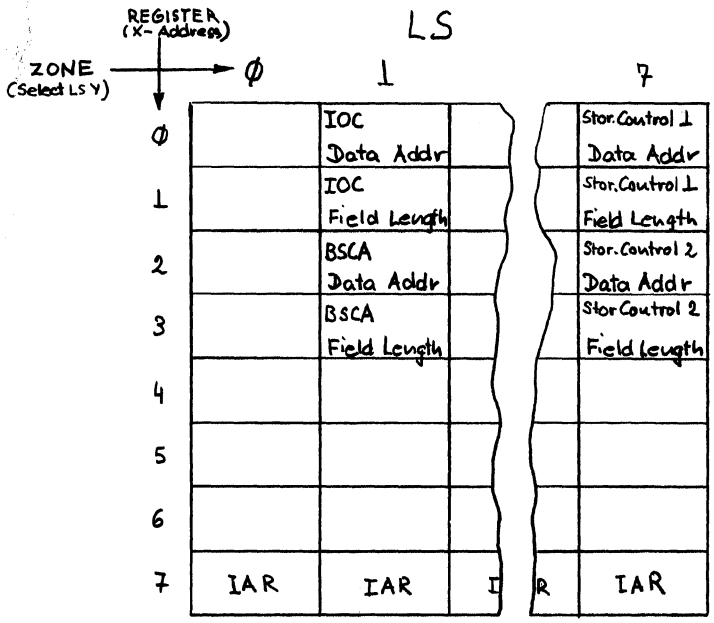


Figure 2-26 CS Data Address and Field Length Registers

4. Delta Process not CS Request.

5. Delta Process without Check and CS Request.

6. Activating of the Set up Run Condition gate to start CPU if a CS request during CPU stop.

CYCLE STEAL LATCH

The Cycle Steal Latch is active if Any CS Request occurs from T6 (pulse B) until T6 (pulse B). This period agrees with the core storage cycle (stolen cycle) (T6 to T6) and defines the time at which CS data transfer is possible. In main, the Cycle Steal Latch is used to block CPU functions by dropping the Process not CS and the Process without Check and CS signals.

DEVICE SELECTION

The Device Selection consists of two latches (CS Device 3 or 4, CS Device 2 or 3) which are reset/set at T6, according to the highest priority CS request when Any CS Request is active.

The outputs of the two latches, gated by the active Cycle Steal Latch, are decoded and generate the CS Select (Device 1, 2, 3, or 4) signal, which is sent to the requesting CS unit.

The two device selection latches save the CS request information of the CS Request Latches, which are already reset at the next

T2 (Sense CS Request), to be available during the cycle steal period (figure 2-27).

CS CONTROL

A CS operation initiated by a CS Request is defined in detail by the CS control lines activated in the requesting CS unit.
The six CS control lines are as follows:

CS Read
CS Write
CS Halfword
CS Byte
CS Increment
CS Decrement

Each CS unit (device) provides its control signals via separate bus lines. Only the six control bus lines of the highest priority CS unit , causing a CS request, are gated to the CPU CS control circuits.

CS Read / CS Write

The CS Read or CS Write control signal defines the CS data direction . For CS Read, data is transferred from the CS unit to the CPU core storage, while for CS Write, data is transferred from core storage to the CS unit. Note that for CS operations the terms Read and Write reflect the I/O operations instead the core storage operations.

A CS operation can either be a read operation or a write operation. Both control signals active or inactive indicate an error and cause a CS Interface Check.
The CS Read latch is reset at T6 when Any CS Request is active and it is set again at T7, but only when CS Read is active.
For CS Write the CS Read Latch remains off.

CS Halfword/CS Byte

The control signal CS Halfword or CS Byte defines the data format which is transferred either during CS Read or CS Write.
Similar to CS Read/CS Write both control signals are exclusive and a CS Interface Check occurs if both signals are on or off simultaneously.

CS Halfword active turns on the CS Halfword latch at T7. The Latch has been reset, similar to the CS Read Latch, at T6 before. The CS Halfword Latch remains off when CS Byte is active.

CS Increment / CS Decrement

The control signal CS Increment or CS Decrement define the processing direction of the CS data field in the CPU core storage. Depending upon CS Halfword or CS Byte, the data address (LS register) is either incremented or decremented by 2 or by 1.

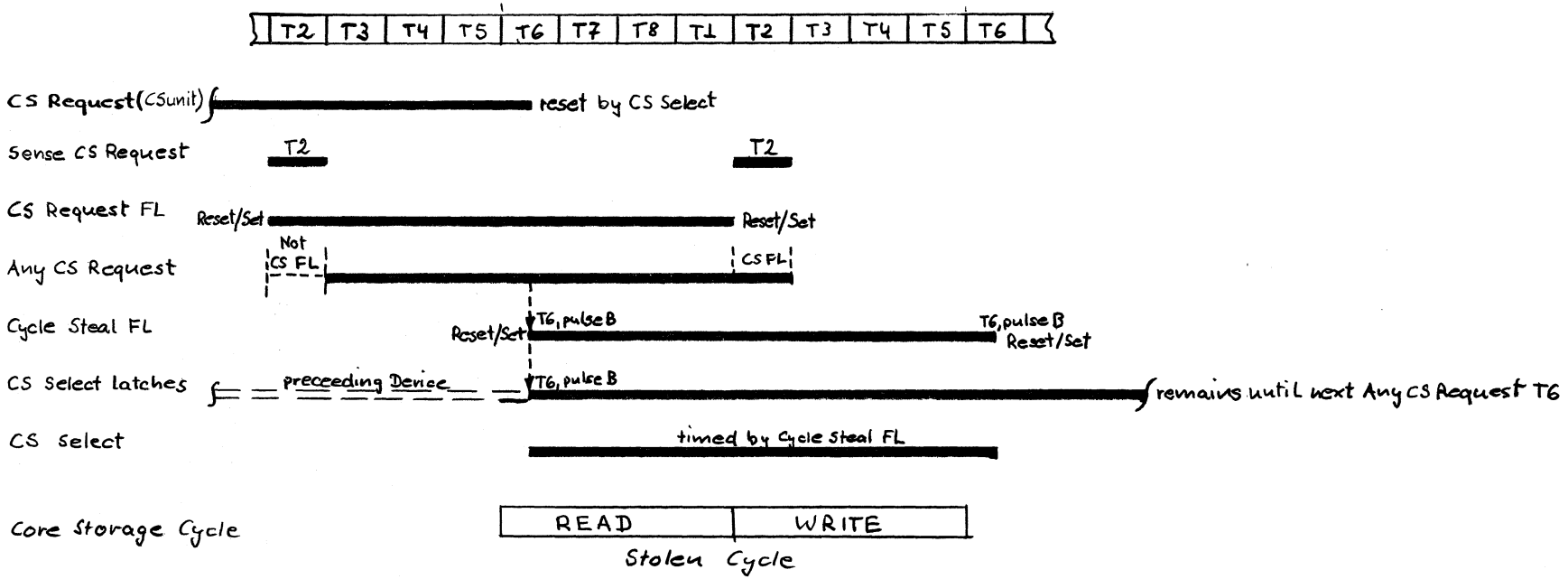


Figure 2-~~44~~ 27 CS Select

As the other control signals CS Increment and CS Decrement are also exclusive and a CS Interface Check occurs when both signals are on or off simultaneously.
CS Increment and CS Decrement do not require to be saved in Latches as CS Read or CS Halfword, since the function (updating data address), which needs the signals, is performed before the signal lines are de-activated in the CS unit.

CS DATA IN

The CS data, provided as byte P0 to 7 and byte P1 to 15 on the CS data bus, is gated to CPU Inhibit when the CS Read Latch is active during the cycle steal period (defined by the active Cycle Steal Latch, T6 to T6). The other sources (ALU, SDR), which are also able to provide inhibit data, are blocked by cycle steal condition.
The two gate signals, which allow the CS Data in bytes (P0-7, P1-15) to activate the Inhibit circuits, are CS Store 0 to 7 and CS store 8 to 15.
For Halfword CS Read (CS Read and CS Halfword Latch active) both CS Store signals are active (T6 - T6) and the complete bus halfword is set into core storage during core storage write time (T2 to T5).
When the CS Halfword Latch is off during CS Read, byte transfer is specified and the CS Store signals are activated depending on SAR (data address) bit 15.

SAR bit 15 = CS Store 8-15
No SAR bit 15 = CS Store 0-7

For CS byte operations the inhibit halfword is completed by regenerating the second byte from SDR. This regeneration is controlled by the signals Force Regen SDR 0-7 and Force Regen SDR 8-15.
These signals are generated during cycle steal (T6-T6) by the inactive CS Store signals.

CS DATA OUT

The SDR bytes P0-7 and P1-15 are sent out to the CS units via CS data bus when the CS Read Latch is off, indicating CS Write. The SDR, reset at the begin of the core storage cycle (T6), is set by the core storage data at T8.
Since CS Write inhibits both CS Store signals (see CS Data in) the Force Regen SDR 0-7 and 8-15 signals are active and cause regeneration of the complete SDR halfword during the write time of the current core storage cycle.
The CS data out bytes are gated to the CS data bus by the CS Data Out gate 0-7 and CS Data Out gate 8-15.
During Halfword CS Write (CS Halfword latch on) both CS Data Out gates are active.
For Byte CS Write the CS Data Out gates are activated depending upon SAR (data address) bit 15.

SAR bit 15 = CS Data Out gate 8-15
No SAR bit 15 = CS Data Out gate 0-7

The CS Data out control operates independently from the cycle steal condition. Since

the CS Read and CS Halfword Latches are only reset/set according to the required CS operation when Any CS Request is active, the Latch conditions are maintained until reset/set during the next following Any CS Request time. Thus, CS Write (as well as CS Read) condition is maintained until the next CS request occurs and the SDR activates the CS data bus even if the CS operation has been terminated.
Since the CS units accept data on the CS data bus only when the corresponding CS Select signal is active (timed by Cycle Steal Latch, T6-T6), the data on the CS data bus outside the cycle steal period are ineffective.

CS MODIFIER CONTROL

During CS operations the modifier is controlled by Increment or Decrement signals generated in the CS Modifier Control circuits. The CPU internal modifier control is blocked by Any CS Request. The CS Modifier Control generates the Increment or Decrement signals, necessary to update CS data addresses and field lengths, depending upon the transferred data format (CS Halfword/Byte) and the CS interface control signals CS Increment and CS Decrement. A comprehensive timing chart is given in figure 2 - 28 . The control bus signals CS Increment and CS Decrement, generated in the requesting CS unit, are exclusive and a CS Interface Check occurs if both signals are on or off simultaneously.

Data Address Updating during CS

The Data Address defines the core storage position (Byte or Halfword) which provides (CS Write) or accepts (CS Read) CS data. The data address is updated for each CS request to provide the core storage address for the next following CS Request forced by the same device.
For updating, the data address is placed into MAR at T5 (LS to MAR, Any CS Request). The updated data address (Modifier output) is placed back into its LS register at T7 (LS Write). Thus, the modifier control signal must be active from T5 until T7.

The four modifier control signal, which are generated for data address updating, are :

CS Increment by 1
CS Increment by 2

CS Decrement by 1
CS Decrement by 2

Incrementing or Decrementing is defined by the control bus signals CS Increment or CS Decrement active.
The control bus signal CS Byte defines that either a byte (CS Byte active) or a halfword (CS Byte inactive) has to be transferred during the current CS operation. Byte transfer forces Increment or Decrement by 1, while halfword transfer (two bytes) forces Increment or Decrement by 2. The forced Increment or Decrement signals are gated to the modifier circuits by Any Request at time T4 to T7.

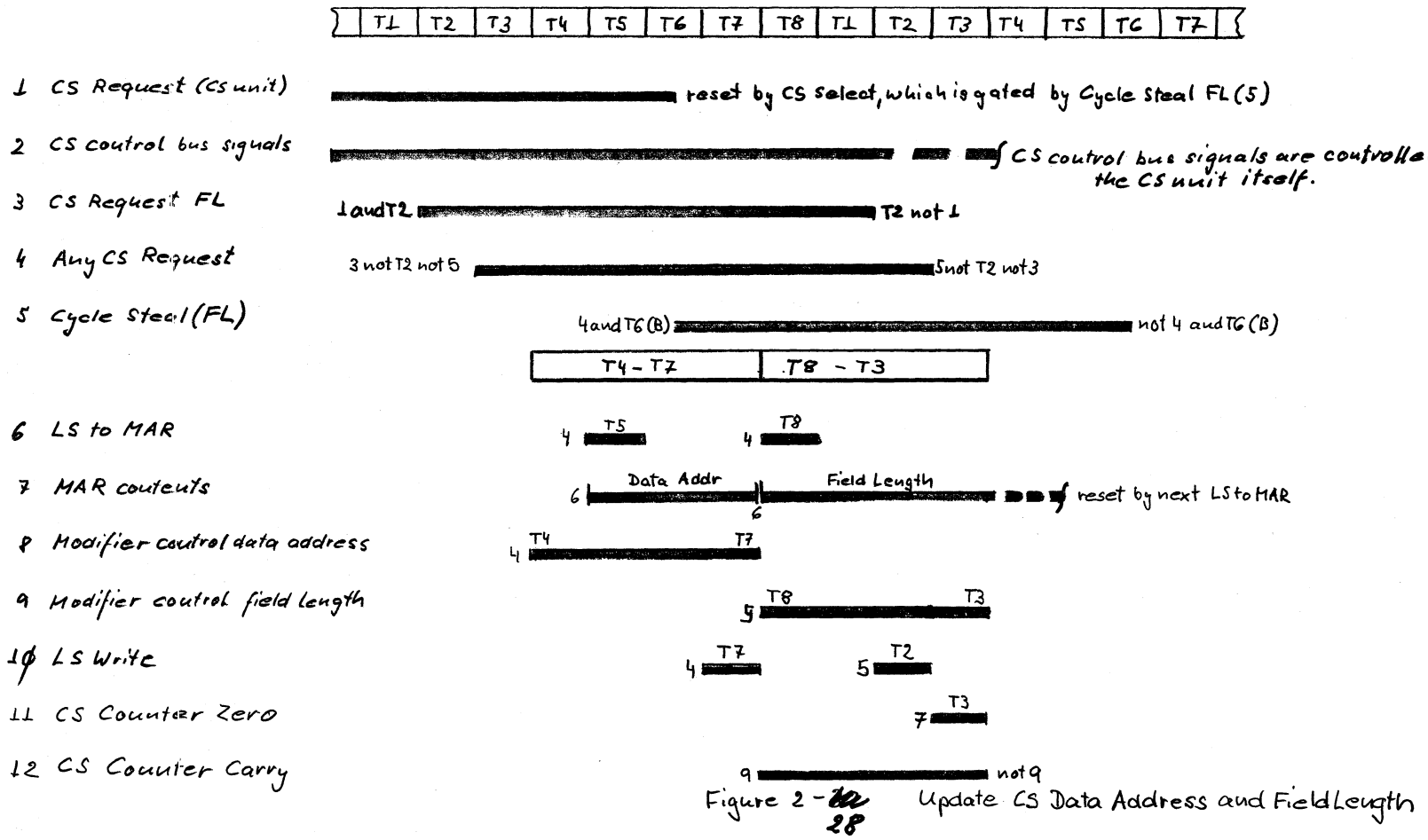


Figure 2 - ~~22~~ 28 Update CS Data Address and Field Length

Field Length Updating during CS

The CS field length , indicating the number of CS operations to be performed, is updated during a CS operation at T8 to T2. Since the field length is always decremented, the transferred data format decides whether the field length is decremented by 1 (Byte transfer) or by 2 (Halfword transfer). The control signals for field length updating are applied to the modifier gated by Cycle Steal Latch active and the time T8 to T3.

CS Counter Zero

The CS Counter Zero signal is an Interface signal sent to the requesting CS unit. This signal, active at T3, informs the CS unit that the field length has been decreased to zero during the current CS operation. The CS Counter Zero signal is generated depending upon the MAR contents (field length) and the transferred data format (indicated by CS Halfword Latch).
For Byte transfer the field length is decremented by 1 and the CS Counter Zero signal is activated when the MAR contains /0001/ (MAR bit 0-13 zero and MAR bits 14-15 = 0-1).
Note that the signal is generated depending upon the field length before updating by the modifier , which provides the decrement result /0000/ at its output.
For Halfword transfer the field length is decremented by 2 and the CS Counter Zero signal is activated when the MAR contains /0002/ (MAR bits 0-13 zero and MAR bits 14-15 = 1-0).

CS Counter Carry

The CS Counter Carry signal is activated during a CS operation when the field length is decreased below zero. This signal is active as long as the modifier control signal (CS Decrement by 1 or 2, T8 to T3) is applied.

CS Modified SAR Bit 15

The signal CS Modified SAR bit 15, sent out to the CS units, is generated during a CS operation depending upon the transferred data format and SAR bit 15. The data format, byte or halfword, is defined by the CS Halfword latch on or off. During halfword operations the modified SAR bit 15 is active when SAR bit 15 is on.
During byte operations the modified SAR bit 15 is active when SAR bit 15 is off.

Checks during CS Operations

The checks which can occur during CS operations are :

- CS Address Check
- CS Inhibit Check
- CS Interface Check

These checks do not stop CPU (Process Check) immediately. The check condition

indicated by an active check latch is sent to the requesting CS unit, which saves the check condition in latches and forces a Trap Request.
The Trap Request forces the CPU by PL switching to branch to a micro program routine which services the check condition.

CS Address Check

The CS Address Check Latch is activated at T7 (MAR contains data address, T5 to T7) when the MAR contains an address which specifies a core storage position outside the customer area (Address Check Condition High or Low). CS Address Check turns off CS Read Latch to prevent writing into a core storage position outside customer area.

CS Inhibit Check

The CS Inhibit Check Latch is activated at T3 during the write period of the CS core storage cycle (Cycle Steal Latch active), but only when the CS Read Latch (data from CS unit to core storage) is turned on.
The check is forced when the inhibit bytes P0 to 7 or P1 to 15 are even at T3. The erroneous data is written into core storage.

CS Interface Check

The CS Interface Check latch is turned on during the period T4 to T7 when both or none of the CS control bus signals (CS Read/CS Write, CS Halfword/CS Byte, CS Increment/CS Decrement) are active. The CS Interface condition (latch active) is sent to the four possible CS units (devices 1, 2, 3, and 4) via separate bus lines. Which bus line has to be activated is decided by the active CS Request latch. CS Interface Check turns off the CS Read Latch to prevent writing into core storage.

CS Test

The CS Test circuit allows unconditional activating of the three CS check latches for test purposes (in the CS units).

The CS Test latch, which forces turning on of the CS Address and CS Inhibit Check Latches at T8, is set during the first cycle of any I/O read operation. The CS Test Latch is activated by no CS Increment and CS Decrement control bus signal and the other control bus signal valid.
No CS Increment and CS Decrement turns on the CS Interface Check Latch and forces the signal CS Re-Address, which prevents data address updating by blocking the CS Modifier control signal (data address is moved through the modifier unchanged). However, the field length is decremented according to the indicated (CS Byte/CS Halfword) data format. The Test and check latch are reset during the next CS request (Any CS Request active).

cycle 0 (End Op signals), which is out of the normal sequence (o to 3).

Depending upon the instruction in process, the following signals are generated :

End OP CY 0
End OP CY 1
End OP CY 2

Skip CY 0 to CY 2
Skip CY 0 to CY 3
Skip CY 1 to CY 3

See figure 4-61 in the FEMDM.

CYCLE CONTROL FOR OPERATIONS WITH AUTOMATIC LENGTH COUNT

- ALC operations can be micro-instructions or MANOP's.
- ALC operations repeat required functions until an End Op condition occurs.
- The function cycles (1 to 3) are repeated controlled by Skip to repetition signals.
- ALC operations require indirect addressing (operands in Main Storage).
- The number of halfwords or bytes to be processed is specified by a field length.
- The field length is stored in specified LS registers.
- Normally the ALC operations ends when the field length is decreased below zero (Decrement Carry).
- Packed Decimal Instructions provide two field lengths ; one for each operand.
- For Packed Decimal Instructions the Decrement Carry is stored in Length Count Latches and the operations end when both field lengths are decremented below zero.
- A CLC, ALC, operations ends either if the field length is reduced below zero or when the compare result is unequal.
- The IAR updating for ALC instructions is performed in the last cycle of the complete ALC operation (End Op condition)
- IAR updating in the last cycle allows recalling of the instruction after interruption by trap requests.
- The operand addresses are automatically updated.

Automatic Length Count (ALC) operations can be either micro-instructions or MANOP's. ALC operations start with cycle 0 and repeat the required function cycles 1, 2 , or 3 until an End Op condition occurs. The repetition of the required cycles is controlled by the skip to repetition signals, which switch the Long Time Clock accordingly.
The common requirement for ALC operations is that the operands are in Main storage (indirect addressing) to provide strings of data (halfwords or bytes).

The number of halfwords or bytes to be processed is specified by a field length in a fixed LS register.

Move, Binary Arithmetic, and Logical Instructions with ALC

These micro-instructions are :

1. MVH, MVB, MVN, MVZ
2. AH, AHSC, SH, SHSC
3. AND, OR, XOR (for CLC see CLC with ALC)

The prerequisites to operate using ALC are:

1. Both operands must be located in Main storage (XX type instruction; OP REG bits 8 and 12 on).
2. The R1 field must address LS register 3, which defines the start address of the To-field (first operand).
3. The R2 field must address LS register 5, containing the start address of the From-field (second operand).
4. LS register 1 must contain the field length.

The ALC instructions start with cycle 0 (delta cycle 0).
At T1 the micro-instruction is set into the OP REG and the OP REG Decode signals are active. Since no End Op condition exists, the required skip to repetition signal (Skip CY 3 to CY 1) is active, too. However, this signal is affective during cycle 3 only.
The Long Time Clock is advanced either automatically or by skip ahead signals (MVH, MVB ; Skip CY 1 to CY 3) until after cycle 3. At the end of cycle 3 the Long Time Clock is set again to cycle 1 (Skip to repetition; Skip CY 3 to CY 1).

During the function cycles (1, 2 or 3), the field length (LS register 1) is decremented by 1 or 2 (cycle 1) and the data addresses are incremented by 1 or 2 depending upon the operated data format (byte or halfword). Especially for MVH and MVB, the data addresses can be incremented or decremented depending on the required processing direction. Micro-instruction bit 6 defines either incrementing (bit 6 off) or decrementing (bit 6 on).

End Op ALC

The End Op condition for the Move, Binary Arithmetic, and Logical instructions is the carry (Decrement Carry 0), which occurs during cycle 1 when the field length is decremented below zero. The Decrement Carry 0 signal is active as long as the corresponding decrement by 1 or 2 signal (T8 to T3, CY 1). Thus, the End Op gate timed by cycle 1 (T1-T8) is active from T1 to T3.
The End Op gate sets the End Op latch at T3. The active End Op latch disables the Skip to repetition (Skip CY 3 to CY 1). Thus, the Long Time Clock advances from cycle 3 to cycle 0 (start next micro-instruction).
The End Op latch is reset at T3 during cycle 0 of the next micro-instruction.

Packed Decimal Instructions with ALC

For packed decimal instructions (AP, SP, ZAP, PPC), ALC functions only when operating in program level 0 (CPU program level).

ALC is performed when :

1. XX type instruction
2. R1 field specifies LS register 3
3. R2 field specifies LS register 5
4. Operand 1 field length is in LS register 1
5. Operand 2 field length is in LS register 0

Both field lengths are decremented by one each time a byte is processed. The operand addresses in LS registers 3 and 5 are incremented by one.

The operations start with cycle 0. The skip to repetition signal Skip CY 3 to CY 1 is active when the micro-instruction is placed into the OP REG (OP REG Decode and bits active).

The three function cycles 1, 2 and 3 are performed in sequence (the Long Time Clock advances automatically). At the end of cycle 3 the Long Time Clock is set to cycle 1 (Skip CY 3 to CY 1) and the cycles 1, 2 and 3 are performed again. The procedure is repeated until both field lengths are counted below zero (Decrement Carry 0).

Length Count Latches

The three Length Count (LC) latches (AUX•LC1, LC1, LC2) store the Decrement Carry 0 information when one of the field lengths is decremented below zero.

Storing the Decrement Carry information is necessary, since after a carry has occured the field length is /FFFF/. This value will be further decreased during the following repetitions (until the other field length is decreased below zero).

LS register 0 (operand 2 field length) is decremented by one during cycle 1, while LS register 1 (operand 1 field length) is decremented by one during cycle 2.

Thus, a Decrement Carry 0 can occur at T1 to T3 during both cycles.

The LC latches are set at T3 either during cycle 1 (LC 2 latch; field length operand 2) or during cycle 2 (AUX LC 1 and LC 1; field length operand 1) when a Decrement Carry 0 occurs.

LC 1 and LC 2 latch on generates the LC End Op Gate. The active LC End Op gate turns on the End Op latch at T3 during cycle 3 (End OP gate).

The active End Op latch prevents the skip to repetition and the Long Time Clock advances automatically from cycle 3 to cycle 0 (start next micro-instruction).

The LC latches are reset at T7 after the End Op latch has turned on. The End Op latch is reset at T3 during cycle 0 of the next instruction.

CLC with ALC

The prerequisites to perform CLC with ALC are:

1. XX type instruction (byte compare),
2. R1 field specifies LS register 3 (first operand address),
3. R2 field specifies LS register 5 (second operand address),

BASIC CPU OPERATIONS

- CPU Basic Operations are executed by CPU circuits directly.

- Basic operations are :
  1. Micro-Instruction Operations,
  2. Circuit controlled Manual Operations (MANOP's),
  3. Cycle Steal Operations,

- Basic operations (micro-instructions) execute the actions called for by the machine language (customer program) instructions.

- To do this a more or less expansive chain of individual micro instructions (micro program) is required.

MICRO-INSTRUCTION-OPERATIONS

Micro-Instruction operations are defined by the Op code of the micro-instructions. Since 43 different Op codes are available also 43 basically different operations are performed.
The number of different operations is additionally extended by direct or indirect operand addressing, which also influences the run down of micro-instructions.

Micro-instruction operations require 1 to 4 CPU cycles (cycle 0 to 3) to execute their functions indicated by the Op code. Every micro-instruction operation starts with delta cycle and cycle 0. However, depending upon Op code and direct or indirect addressing the micro-instructions can be terminated after any of the four possible cycles.
Since micro-instructions are stored in the control area (including the auxiliary storage) of the Main storage every micro-instruction operation starts with loading the micro-instruction into the OP REG (T1, start cycle 0). After the micro-instruction has been placed into the OP REG, the further operation is controlled by the OP REG decoding signals.
The control signals necessary to read out and to load a micro-instruction into the OP REG are generated depending upon the delta cycle 0 condition only. Beside loading the micro-instruction into OP REG the incrementing of the micro-instruction address (IAR) by 2 is initiated.
Flow charts and timing charts for all possible micro -instruction operations and their variations depending upon direct or indirect operand addressing are given in the FEMDM, 2020 CPU, Chapter 5 Operations.
Since the micro-instruction flow and timing charts are all similar in their representation, only some typical instruction run downs are explained in the following sec-

tions to give in principles how to interpret the charts.

Micro-Instruction Flow and Timing Charts General

Flow chart and timing chart for one micro-instruction are shown on two pages that face each other. The time basic (T-pulses and cycles) of both charts are aligned to allow referring between both charts.

Flow Chart

The upper part of the flow chart shows the instruction layout, the description of the micro-instruction, and a simplified example of the function execute by the instruction.
The lower part of the chart, the actual flow chart, is divided into rows, which are assigned to the several CPU data flow components. The CPU data flow components are represented as blocks which are connected by lines indicating the data and information flow.
The vertical lines, aligned with the time basis, define the real time at which a function is executed.
The data (halfwords) handled during a micro-instruction operation are generally expressed by four hexa-decimal digits. The hexa-decimal digits are represented either true (0 to F), or as undefined digits (N, D_0, X, Y).
Beside hexa-decimal values, terms (u-instruction, u-instr. addr, NSI) are used to represent data or information .
Take into consideration the data registers (SDR, FDR, TDR) and the address registers (SAR, MAR) as well as the local store registers are controlled by reset/set signals. On the other hand there are unlatched data flow components, like Shift Unit, Invert Switch, ALU, Modifier, and Inhibit Switch, which only provide the required data when the correct input data is available and the corresponding control signals (e. g. , FDR Invert, shift by 2, Increment by 2) are active.

Timing Chart

The timing chart shows the main control signals generated depending upon the micro-instruction bit pattern in the OP REG or generated depending upon the active delta cycle or cycle. Take into consideration that the timing chart shows the control signals, necessary to execute the micro-instructiin as shown in the flowchart, as well as 'DON'T CARE' signals , which are generated without affecting the operation.

Examples of Micro-Instruction Operations

Load Byte Immediate - LBI (FEMDM, 2020 CPU, figure 5-2)

The LBI-instruction loads one byte into a LS register. As shown in the instruction layout

the byte to be loaded is carried by the instruction itself (immediate data, bits 8 to 15), while the LS register into which the byte has to be loaded is specified in the T0 - Reg field (bits 6-to 7).

The micro instruction operation starts at T4 with delta cycle 0 (delta cycle latches of the Long Time Clock are turned off).

Micro-Instruction Addressing: At T5 the address in the IAR of the LS Zone specified by the present New PL (New PL zone gate active) is moved into SAR. This instruction step is controlled by the signal LS to SAR, which activates the SA gate (read LS) and gates the LS data out bits into SAR. LS to SAR and the X-address 7 (IAR = LS register 7) are forced by delta cycle 0, T5.
Note that the instruction address is set into SAR before start of the next core storage cycle (T6 = start read).

IAR Updating: At T6 the IAR, still containing the instruction address, is moved into MAR by the active LS to MAR signal and the forced X-address 7. LS to MAR activates the SA gate and controls the MAR input. After the address has been placed into MAR, it is applied to the Modifier, which is controlled by the active modifier control signal Increment by 2. Thus, the updated instruction address (+2 = next sequential instruction address, NSI) is available at the modifier output. Note the modifier control signal Increment by 2 is always active when no other increment or decrement signal is generated.
The modifier output (updated address, NSI) is moved into IAR at T2. This step is controlled by MAR to LS, the forced X-address 7 and the LS Write signal.
Note that LS Write generates the Bit Timing signal (pulse B) which is the reset/set signal for the addressed LS register.

Read out and Regenerate Micro-Instruction: During read time of the core storage cycle, the sense bits of the core storage position defined by the micro-instruction address in SAR are set into SDR at T8 (micro-instruction in SDR).
Note that SDR is reset at each T6.

The control signal SDR to Inhibit, active for cycle 0, provides the SDR contents to the Inhibit Switch for regeneration during the core storage write time.
Note that the address used to read out the instruction is still available in SAR during write time (until T5).

Micro-Instruction to OP REG: The micro-instruction in SDR is applied to the OP REG. The reset/set signal, which allows the instruction to enter the OP REG, is generated at T1 (start cycle 0) when the Set OP REG gate is active. The Set OP REG gate is activated by delta cycle 0 and no Process Check. Thus, in case of any process (CPU) check, the instruction, which causes the check, is saved in OP REG and can be displayed (CE) for trouble shooting.

SDR to TDR: At T2 the SDR contents (micro instruction) is placed into TDR by the signal SDR to TDR active. The signal SDR to TDR at T2 is generally generated during the cycles 0, 1, and 2.

Logical Unit: The control signals of the logical Unit, which is combined by Shift Unit, Invert Switch, and ALU, are generally timed by cycles. However, the expected result is not available at ALU output until the correct operands are placed into TDR and FDR. That means for the current example, that the Logical unit is already controlled at T1, but the correct input data are placed into TDR at T2 (SDR to TDR).

The Eight Shift is controlled by the signals TDR 0-7 to SU 0-7 and TDR 8-15 to SU 8-15, which are activated by the turned on Not Shift 8 Latch (cycle 0, T2). No Shift by 2 or 4 signal is activated and that causes the Not Shift 0-7 and the Not Shift 8-15 signal. The suppress circuits are controlled by the signals Suppress 0-3 and Suppress 4-7.

Depending upon the active Shift Unit controls only the highorder byte (Op code and To-register) of the TDR halfword is suppressed. Since suppressed data is set to zero (parity is corrected) the halfword applied to the ALU input provides the immediate data byte (XX) in the loworder byte and the highorder byte set to zero.

The Invert Switch, controlled by the signals FDR true 0-15 and FDR Invert 0-15 forces zeros at its output. This zero forcing can be considered as suppressing.

The both halfword operands are exclusive ORed (OE) by the ALU. Since exclusive ORing against zero does not affect the data of the second operand, the ALU result provides the immediate data in the low-order byte and the highorder byte zero. The ALU result is available for cycle 0 (no change of the control signals and data in TDR).

ALU to TO-Register: The ALU halfword is provided to the local store input at T5 (ALU to LS). The LS register address is provided by the Old PL (Old PL Reg selected by the Old PL zone gate), and the X-address which bases on the decoding of the TO-REG field is allowed by the TO-REG Select signal.

The reset/set signal for the addressed LS register (Bit Timing) is generated by LS Write at T8. LS Write is suppressed if the TO-REG field defines the IAR (X-address /7/) and the ALU output data is lost. After T8 the TO register (101 to 161) contains the immediate data in the low-order byte and the highorder byte is zero.

Initiating the Next Micro-Instruction :

When the LBI has been set into OP REG, the End OP cycle 0 signal is activated depending upon the Op code (OP REG Decode signal). The End Op cycle 0 signal prevents advancing of the Long Time Clock. Thus, after LBI delta cycle 0 a new delta cycle 0 starts.

During this following delta cycle 0 reading of the next micro instruction is performed. This reading starts with moving the IAR

to SAR at T5 (still shown in the flowchart) Remember that this step is performed using the Select LS-Y signal which is generated depending upon the New PL (New PL Zone gate). If the New PL has been changed in the meantime by an incoming higher priority trap request , or by resetting a PL another LS zone is defined and the IAR of this new zone is moved to SAR instead of the IAR containing the NSI (old zone).

Note the next micro instruction starts (delta cycle 0)functions of the next instruction use the zone addresses of the New PL while the current instruction uses still the old zone address. This simultaneously operating in two different zones is controlled by the Old and New PL zone gates.

ADD Packed Byte - AP, XX-Type
(FEMDM, 2020 CPU, figure 5-130)

The AP instruction, XX-type, adds two packed bytes decimal. Both operand bytes are in core storage and are specified by addresses available in the LS register defined by the TO-and FROM field in the instruction. The AC bit (Address Check bit) on allows testing of the operand addresses to be within the customer area.

An Address Check forces Trap Request 2 which causes a circuit controlled branch (new IAR used) to the Address Check micro-program routine.

An occuring carry is saved in the Carry Latch and the Condition Code is set unconditionally.

Every time an operand byte has been read out from core storage the corresponding address is increased by 1.

The AP instruction, XX-type, is performed as Automatic Length Count (ALC) operation when the To-register field is /3/ and the From-register field is /5/. For ALC the LS register 1 contains the field length for the To-operand and LS register 0 contains the field length for the From-operand. The ALC operation is terminated (End Op) when both field length are counted below zero (Decrement Carry 0). If the field lengths are different the following happens:

Field Length of To-operand larger than field length of From-operand:

When the field length of the From-operand is decreased below zero, the remaining bytes of the To-operand are added against zero. This is achieved by preventing Storage Use (no storage operation) for the cycles which read out From-operand bytes.

Field Length of To-operand smaller than field length of From-operand:

Storage Use is prevented when the field length of the To-operand has been decreased below zero. This prevents reading and storing of To-operand bytes. The From-operand bytes are added against zero. The overflow latch of the Condition code is turned on when the remaining bytes of the

From-operand are not all zero. Thus, overflow latch active after the operation indicates that the To-field was to small to accept all bytes of the From-field.

During the AP instruction, the packed bytes are tested to contain valid decimal digits. Invalid decimal digits turn on the Data Error Latch, which causes a trap request 2. However, the data error handling is only defined when CPU operates in PL 0.

Delta Cycle and Cycle 0: The IAR is moved to SAR to address the micro-instruction. The addressed micro-instruction is read out (SDR, T8), set into OP REG (T1), and regenerated during write cycle (SDR to Inhibit). No IAR updating is performed, since LS Write at T2 (Updated instruction to IAR) is inhibited by the instruction in OP REG.

Delta Cycle and Cycle 1: The common function of cycle 1 is reading the From-operand (valid for all micro-instructions). At delta CY 1, T5, the address is moved from the From-Reg to MAR and SAR. The modifier is controlled by Decrement by 1 (T4-T3, delta CY 1). The address in MAR is checked at T6 when the AC bit is on. At T7 the decremented From-address is set into the From-Reg (LS Write).

At delta CY 1, T8, LS register 0 (field length From-operand) is moved to MAR. The modifier is still controlled by Decrement by 1 (active during delta CY 1). At T2, the decremented field length returns to LS register 0, but only when ALC is specified. No ALC suppresses LS Write at T2.

When the field length is decremented below zero , a Decrement Carry 0 occurs. This decrement carry is generated as long as the field length is available in MAR and the modifier control signal (Decrement by 1) is active.

The Decrement Carry 0 turns on the Length Count Latch 2 at T3. For ALC operations LC Latch 2 active prevents Storage Use during the following repetitions of cycle 1 (prevents further reading of From-operands). No Storage Use force the From-operand bytes to zero (parity is corrected).

Assumed that the LC latch 2 is off , a core storage cycle is performed. At T8, the halfword, addressed by the From-address in SAR is read into SDR. This halfword is regenerated during write time of the current core storage cycle (SDR to Inhibit From-address still in SAR).

At T2, cycle 1, the SDR halfword is moved to TDR. To select the operand byte out of the TDR halfword, the Eight Shift is controlled by SAR bit 15 (SAR still contains the From-address).

The active signals Suppress 0-3 and 4-7 suppresses the byte not specified by the address in SAR (highorder). Within the Shift Unit the operand byte (loworder) is tested for being packed (Data Error, Trap

Request 2). The highorder byte of the halfword, provided by the SU to the ALU, is zero (Suppress 0-7), while the loworder byte carries the operand byte.
The Invert Switch is forced to ones, since no invert switch control signal (True or Invert) is generated.
Invert Switch halfword and SU halfword are exclusive ORed (not ALU control gate active) by the ALU. This exclusive ORing inverts the SU halfword.
The Six Correction circuits activated (AP and no Carry bit 8 and 12 during OE operation) and the constant '6' is subtracted from the inverted, loworder operand byte. The highorder byte is not affected by the six correction.
The resulting ALU halfword is set into FDR (ALU to FDR) at T8, end of CY 1.
The FDR has been reset before at T6, CY1, by the FDR Reset Signal, generated depending upon the active ALU to FDR condition.

Delta Cycle and Cycle 2: The common function of cycle 2 is reading the To-operand (valid for all micro-instructions).
At T5, delta CY 2, the address is moved from the To-Reg to MAR and SAR. The modifier is controlled by Decrement by 1, however, the updated operand address is not set back into the To-Reg, since at T7, delta CY 2, no LS Write is generated. This ensures that the To-address is unchanged for the store cycle 3, which needs the origin To-address to store the result.
The To-address in MAR is tested at T6 for being valid when the AC bit is on.
For updating the To-operand field length the halfword in LS register 1 is moved into MAR at T8. The modifier is controlled by Decrement by 1. LS Write is activated at T2, but only when an ALC operation is specified, and the decremented field length is set into LS register 1. A Decrement Carry 0, which occurs when the field length is decreased below zero turns on the LC Latch 1 which prevents Storage Use for the To-operand cycles (CY 2 and 3) of the following ALC repetitions. The halfword, defined by the To-address in SAR, is read out into SDR and regenerated. At T2, the halfword is set into TDR. The Eight Shift selects the To-operand byte depending upon SAR bit 15 (Not Shift 8 Latch).
The SU suppresses the highorder byte (Suppress 0-4 and 4-7). The SU halfword applied to the ALU shows the true To-operand byte in the loworder byte while the highorder byte is forced to zero by suppressing.
The Invert Switch, controlled by FDR Invert 0-15, provides the modified From-operand (Inverted and six corrected) in FDR inverted to the ALU.
The ALU is controlled by the Adder gated. If the Aux Carry Latch has been set by a carry in a previous AP operation, a one is additionally added (Additional Carry) to the ALU result. The six correction circuits are activated when during the ALU operation no carry (no Carry bit 8 and 12) occurs out of the two four-bit positions of the loworder byte. (A carry, which occurs out of the loworder byte (ALU Carry bit 12, before six correction) during adding.)

The input condition of the Logical Unit (FDR, TDR) and all necessary control signals remain also during CY 3.

Delta Cycle and Cycle 3: The common function of cycle 3 is storing the result into the core storage position of the To-operand (valid for all micro-instructions).
At T5, delta CY 3, the To-address, not updated during delta CY 2, is moved to MAR and SAR. The address is tested at T6 (AC bit on) and decremented by 1. The updated To-address is set into the To-Reg at T7 (LS Write active).
At T8, delta CY 3, the IAR updating, prevented during delta CY 0, is initiated by moving IAR to MAR. The modifier is controlled by Increment by 2. The LS Write signal , generated at T2, sets the updated micro-instruction address back into IAR. This LS Write signal is only activated when both field lengths are decremented below zero.
This IAR updating depending upon End Op conditions allows recalling of ALC operations (updated operand addresses and field lengths are available) after interruptions by trap requests.

During CY 3, the halfword, defined by the not updated To-operand address in SAR is read into SDR. During write time the loworder ALU byte is moved into Inhibit highorder or lower byte depending upon SAR bit 15. (ALU 8-15 to Inhibit 0-7 or 8-15). The second Inhibit byte is provided by SDR (SDR 0-7 or 8-15 to Inhibit).

A carry , which occurs out of the loworder byte (ALU carry bit 12, before six correction) turns on the Carry Latch at T1.
The condition code latches are set according to the result condition.
The three possible condition codes are :

Result Zero
Result not Zero
Overflow

When the condition code has once been set for Result not Zero, further Condition code settings are suppressed, since this condition is true for the complete operation.
Overflow indicates set the To-operand field was to small to accept all coincide bytes of the From-operand field.

Repeating of cycles during ALC operations is controlled by the Skip to repetition signals (AP - ALC = Skip CY 3 to CY 1), which are activated depending upon the instruction in the OP REG. These skip to repetition signals prevent Long Time Clock advancing to cycle 0 after cycle 3. However, Skip to repetition signals are prevented when the End Op condition occurs (AP-ALC, both field lengths below zero = LC latches 1 and 2 turned on) and the next micro-instruction operation is initiated by moving the NSI (IAR) to SAR (delta CY 0, T5).

CIRCUIT CONTROLLED MANUAL OPERATIONS (MANOPS)

Information on the following circuit controlled manual operations was not available at time of printing:

1. Storage alter
2. Storage fill
3. Storage display
4. Storage scan
5. Storage test
6. LS register alter
7. LS register display
8. Initial control program load
9. CPU log in

CYCLE STEAL OPERATIONS

Information on the cycle steal operations was not available at time of printing.

MICRO-PROGRAM OPERATIONS

- Micro-program operations execute machine language instructions (customer program) as well as special functions like System Reset, Program Load, and micro-program controlled manual operations (Instruction Step, Address Stop, etc.)
- For micro-program operations concerning CPU refer to FEMDM, 2020 CPU, Appendix B.
- For micro-program operations concerning I/O's refer to the appropriate attachment FEMDM or FETOM.

CHAPTER 4. FEATURES

I/O attachments, channels, and control features allowing I/O devices to be connected to the CPU are considered as features.

For details of these features see the appropriate IBM Field Engineering Theory of Operation manuals (FETOM's).

CHAPTER 5. POWER SUPPLIES AND CONTROL

POWER SUPPLY AND POWER DISTRIBUTION

- The CPU power supply provides all necessary ac and dc voltages.
- The main line power is applied to the power supply via a filter when the main circuit breaker is closed.
- For power on or off, the contactors are switched in a sequence controlled by the sequence control circuits.

The CPU power supply circuits distribute the ac voltages and generate and distribute the necessary dc voltages.

To distribute the ac and dc voltages to the several I/O units, female plugs are provided at the CPU power tower to connect I/O power cables.

Power supplies for 60 or 50 Hz differ in ac components like transformers, overload switches, etc., but the general control is similar for both power supply types.

The main line enters the power supply via a filter which suppresses all frequencies others than the required main line frequency (50 or 60 Hz). When the main circuit breaker (CB) is closed, 24 volts ac are provided to the power sequence control circuits. Beside this control voltage, 115 volts ac are provided to all convenient outlets (also I/O's) and the 41 volts ac for use meter control are generated (transformed). Contactors, operated by the sequence control circuits during power on, provide the main line voltage to the I/O's (via cable) and to the dc generating components (Mid Pack transformer ; DC power supplies, PS).

The main (Mid Pack) transformer has a 3-phase primary winding. In 60 Hz machines the primary winding can be adjusted by tapping for 230 volts or 208 volts. In 50 Hz machines, the primary winding can be adjusted to 380 v, 220 v, 235 v, or 195 v supplies by jumpers on a switch block (Y - Δ) and/or by tapping.
Five secondary windings are wound in 6-phase start configuration. A sixth winding provides the 13 volts dc (after rectification), used for line failure detection and missing phase detection (50 Hz only).
Each secondary voltage is rectified by six diodes ; therefore, a low ripple dc voltage is obtained which requires little filtering.

The obtained dc voltages are supplied to the regulation circuits in the Mid Pack modules (PS).
The Mid Pack transformer and the ac distribution are protected by circuit breakers (CB's) with thermic and/or magnetic overload tripping.

Similarly, the dc voltages are also protected by CB's (in the appropriated PS).

POWER ON/OFF SEQUENCES

- To initiate Power On, the under voltage, overcurrent, and overtemperature conditions must be normal.
- The Power On and Off sequence charts are given in FEMDM, 2020 CPU, fig. 6-00 to 6-10.
- EC-controlled timing charts for Power On or Off are given ALD pages YB 147 (60 Hz) and YB 047 (50 Hz).

EMERGENCY POWER OFF

- Operating the EPO switch either on the CPU customer console or on the I/O attached to the SIOC (IBM 1419, 1259) removes all power except 24 volt ac (power supply control) and 41 volts ac use meter power simultaneously without sequencing.

POWER FAILURES

- Power failures initiate power off sequencing.
- The fact that a power failure has occurred can be recognized when the Power Ready Light turns off without operating the Power Off Key.
- The type of power failure is indicated by the Attention indicators.
- Power failures can be caused by:
  Main line failure conditions,
  circuit overloads,
  logical voltages too low,
  overtemperature in any system area.

Line Failures

Line failures are recognized by the line failure detection circuit or by the missing phase circuit (50 Hz only).
Both circuits are supplied by the 13 volts dc, generated by the Mid Pack transformer. This voltage is rectified, but not filtered.

The line failure detection circuit contains a reed relay, which is picked when the 13v dc are applied. This relay drops, initiating power off sequencing, when the 13 v dc input is decreased to approximately 8 volts for more than 500 microseconds. These types of line failures can be caused e. g., by static on the line or net switchings.

The missing phase circuit is activated (a reed relay picks) in a three phase power System, when one phase is missing or too low in relation to the other two line phases. This missing phase condition can occur on the main line (street lightning control by dropping one phase) or within the power supply itself (failing contacts).

The missing phase detection is performed depending upon the increasing ripple (13v dc) after the six-phase (diodes) rectification. The picked missing phase relay generates

the logical signal Missing Phase which is sent to the CPU circuits.
This signal turns on the Missing Phase Latch, when a Process Check occurs. The active Missing Phase Latch is sensed by micro program (LOG).

The picked missing phase relay also allows current flow through a delay action circuit breaker (mounted on the CE console). About 5 to 10 seconds later this circuit breaker operates (push-button out) and initiates power off sequencing. Power can be turned on again by operating the power on key only. If the phase is still missing, power turns off again after the thermal delay time. However, power stays on if the phase has returned in the meantime. If the push-button of the delay action circuit breaker (CE console) is out, it indicates to the CE that missing phase had occurred. After recognizing this fact the CE must restore the push-button.

Line Failure or Missing Phase, causing a power off switching, turns on the Power Failure and the Line Failure attention indicator on the customer console.
The CPU has been stopped by the active Storage Protect signal (drops Hold Run Condition).

Circuit Overload

Circuit breakers (CB's) in the ac circuits and dc power moduls trip when the current exceeds the defined limit.
Any tripping CB initiates the power off sequence and stops the CPU after the current cycle by the active Storage Protect signal.
Circuit Overload turns on the Power Failure attention indicator.

Undervoltage

The Logical dc voltages, +6v, +3v, and -3v are applied to a undervoltage sense circuit which initiates power off when one of the logical voltages exceeds its lower limit. The Storage Protect signal is generated which stops the CPU after the current cycle. Undervoltage conditions illuminate the Power Failure attention indicator only.

Note: Undervoltage condition can also be caused by missing +20 v dc (no logical voltage).

Overtemperature

Within the CPU housing and in several I/O boxes thermal switches are mounted at temperature-sensitive positions (e.g., top of gates). All thermal switches, including I/O's, are connected in series and the power off sequence is initiated if one of the contacts opens. The Storage Protect signal is generated to stop CPU after the current cycle. Any overtemperature condition turns on the Power Failure and the Thermal attention indicator on the customer console.

SJ 4 POWER SWITCH

The SJ 4 power switch controls the Storage Protect signal in relation to the +12 volts core storage voltage.
During Power, on the SJ 4 Power Switch ensures that the Storage Protect signal (Prevents Storage Use and CPU run) drops after

the + 12 volts are present (1 to 100 milliseconds).
During Power off , the SJ 4 Power Switch ensures that the Storage Protect signal is de-activated 40 to 258 microseconds before the + 12 are below their operating limit.

SJ 2 POWER CONTROL

The SJ 2 Power Control circuit controls the special negative voltages (-15v, -9v to -12v) required for SJ 2 memories in cooperation with relays of the power sequence control.
In main, the SJ 2 power control enables rapidly discharging of load capacitors of the -15 volts and -9 to -12 volts power supply modules if the power is turned off by emergency or by any power failure.
This rapid voltage drop prevents loosing of memory information.

CHAPTER 6. CONSOLE AND MAINTENANCE FEATURES

CUSTOMER CONSOLE

- The customer console is shown in FEMDM, 2020 CPU, figure 4-20.

The customer console provides the switches, keys and indicators which are necessary to operate and control the system.

Additional switches and indicators are located on the various input/output devices which are included in the system. These switches and indicators control functions or indicate conditions peculiar to the input/output unit on which they are located, and are not considered as system controls.

KEYS

Power On Key

- Operating the power on key initiates power on sequencing in the power supply control circuits.
- Power on sequencing is prevented if a circuit breaker (CB) or circuit protector (CP) has been tripped and not restored or if a line voltage failure exists.
- Operation of the key has no effect if power is already on.
- The power on key is a backlighted push-button which is illuminated when power on sequencing is terminated.
- Power on causes System Reset.

Power Off Key

- Operating the power off key initiates power off sequencing.
- If the power off key is operated during CPU run, the Storage Protect signal, generated within the sequence control, stops CPU after the current processing cycle (delta process and process drop).

Start Key

- Operating the Start key initiates CPU run.
- Operating the start key is ineffective when CPU already runs or when a process check exists.
- Nevertheless, the CPU can be started in case of process check, when the CE process check override switch is on.
- Storage or LS Register Alter or Display operations are repeated until the start key is released (Allow Continuous Display or Alter; CE Single Cycle and Single Micro-Instruction switch off).

Depressing start key turns on the start latch and start interlock latch if the CPU has been stopped. The start interlock latch ensures that the start latch turns on only once , even if the key contact bounces.
During the period at which both latches are on,the setup run condition gate is activated. The set up run condition gate allows turning on of delta process and process latch. Delta process and process active allow to perform processing periods (micro-operations, MANOP's, cycle stealing).

Stop Key

- Operating the stop key, which is a backlighted push button,turns on the stop latch.
- The turned on stop latch illuminates the indicator in the stop key.
- The stop latch is tested during micro-program by a SENS instruction.
- For normal processing (execution of machine language program), the CPU stops by HALT micro-instruction at the end of the current machine instruction (delta process and process turns off).
- Operating stop key during Storage Scan or Fill stops CPU after the single scan or fill operation (handling of one byte) just in progress.
- During Storage Test the CPU stops after the true compare run, when the stop key has been operated.

System Reset Key

- Operating system reset key forces the system into a basic start situation.

Operating system reset key starts CPU.
The run and cycle control is set into an initial status by Reset pulse (System Reset pulse), and Clock Reset (Long Time Clock to delta cycle 0 and cycle 0).
The Reset pulse initiates CPU LOG in (four cycle operation).
The check latches are reset after being logged. The bus lines Reset Condition and Special Reset Condition are activated to reset the I/O control in the attachments.

After CPU LOG in the micro-program controlled System Reset routine (start address /C002/) is started.

The System Reset routine starts with saving the logged information and increasing the process check counter if a process check is up. After this step the local store is reset (LS registers 0 to 6 to zero; IAR's are set to start addresses of the different PL routines ; exceptions : LS registers used for System Reset).

Within the protected area (first 144 bytes) the interrupt priority table, several indicators, and the highorder halfword of the PSW (Condition Code, Channel Mask, ASCII mode bit, Overlap mode bit) are reset to zero.

At the end of the System Reset routine PL 7, which has been forced at system reset start, is turned off and the CPU stops by a HALT micro-instruction in PL 0. After CPU stop the E, S, T, R - and P, I, U, L - display shows zero (parity bits on).

Operating the System Reset key while CPU is running, stops CPU after the current process cycle. However, one cycle period later , the CPU starts again (start latch turns on) to execute System Reset. This restart is necessary to generate reset signals gated by start latch turned on.

Load Key

- Operating the load key initiates reading the customer program into main storage.

- To read a punched card the main card reader is started by an XIO machine instruction, which is internally combined by a micro-program routine.

- The Load Latch is turned on.

- For Load Key functions during Control Program Load refer to the Mode Switch description.

Loading starts with System Reset (CPU LOG in, System Reset routine). However, after the PL 7 is turned off, the active load latch causes a branch to the load routine.

Within the load routine an XIO machine instruction is combined by constants (OP Code D 0, device address, function specification), an address set up in the four Register Data or Address switches (B1, D1 field), and the field length 80 (/50/), which is also available as a constant. This XIO instruction is treated like a normal machine instruction and causes reading a punched card.. After the card (loader card) has been completely read the first machine instruction, just read from the loader card, is executed (addressed by the NSI in PSW, which was set according to the Address switch).

Operating the Load key during CPU run causes the same stop-start procedure as already described for System Reset key with the exception that loading is performed.

The following conditions must be established before the program load function is operable:

1. The mode switch on the console must be in either the process position, address stop position, or instruction step position.

2. Card I/O device no. 1 must be in the ready condition.

The Model 20 may include one or two card I/O devices capable of reading cards. In order that common program load routines may be employed on systems with different card I/O equipment, device 2 responds to the device 1 (2501 Card Reader) instruction when device 1 is not included in the system.

The device 1 instructions to which device 2 responds are:

Device 2 Response	Device 1 Instruction
Read (primary feed if 2560)	Read card
Branch if reader busy	Branch if reader busy
Branch if reader error	Branch if reader error

A device 2 instruction in a system which does not include device 2 is treated as a no-operation.

I/O Check Reset Key

- This key has no logical functions within the CPU circuits.
- Operating the I/O check reset key activates a bus line which applies the I/O Check Reset signal to all attachments.
- For the attachment functions depending upon the active bus line refer to the corresponding attachment FETOM's or FEMDM's.

SWITCHES

Mode Switch

Process

In the process mode, the CPU operates under control of the customer program.

If the mode switch is turned from process or address stop to any other position (except instruction step) while the CPU is running, operation of the CPU continues as if in the process mode until a programmed stop or an error stop occurs, or until the stop key or system reset key is operated. If the mode switch is turned to instruction step, the CPU stops as if the stop key had been operated.

Address Stop (ADR STOP)

The CPU stops when the customer program has reached the machine instruction located at the main storage address indicated by the register data or address switches ; otherwise, operation is the same as in the process mode.

Instruction Step (INSN STEP)

In this mode of operation, the CPU completely executes one machine instruction for each operation of the start key.

Storage Display (STOR DPLY)

The byte located in main storage (at the address indicated by the register address or data switches) is displayed in the U, L display when the start key is operated. The address of this byte is displayed in the E, S, T, R display.

Data in the protected area, customer area, and control area can be displayed. Addressing a not available storage position , the U, L display is zero with parity bits on.

Storage Alter (STOR ALTER)

To alter a main storage byte the register data or address switches are set according to the required address and the byte to be entered is set up in the two data switches.

When the start key is operated the set up byte is moved into the selected main storage position. The byte which was entered is displayed in U, L.
The address is displayed in E, S, T, R.
Normally alter operations are allowed within the customer area only. However, turning on the CE mode switch (CE console) enables the CE to alter bytes in the protected area or in the control area.

Display Register (DPLY REG)

The content (halfword) of the eight General Registers and of certain address registers, which are located in the protected area but are accessible for the customer, may be displayed in this mode.

The register to be displayed is selected by data switch 1 as follows:

Data Switch (Hex)		Register Selected
0	(0000)	First two bytes of previous instruction
1	(0001)	I-recall address register
2	(0010)	PSW (bits 0-15)
3	(0011)	PSW (bits 16-31)
4	(0100)	
5	(0101)	
6	(0110)	
7	(0111)	
8	(1000)	General register 8
9	(1001)	General register 9
A	(1010)	General register 10
B	(1011)	General register 11
C	(1100)	General register 12
D	(1101)	General register 13
E	(1110)	General register 14
F	(1111)	General register 15

Operating the start key displays the selected register in E, S, T, R.
The hexadecimal number of the selected register is indicated in P display.
The first halfword of the last machine instruction executed (op code plus byte 2) may be displayed in E, S, T, R, when register /0/ is selected (see foregoing table).

In this mode the CE can display local store registers when the CE mode switch is turned on. For LS register display the LS zone is selected by data switch 1 (/0/ to /7/), while the required register in the zone is selected by data switch 2 (/0/ to /7 /).
The halfword of the selected LS register is displayed in P, I, U, L.
The E, S, T , R, display shows the values

set up in the register data or address switches which are not used for LS Display.

Alter Register (ALTER REG)

Alter Register mode operates similar to the Display Register mode. However, the content of the selected register is alter according to the halfword set up in the register data or address switches.
The CE mode switch on causes altering of a selected LS register also according to the halfword set up in the register data or address switch. However, during the LS Register Alter operation, the halfword just set into the LS register is read out again and compared against the halfword in the register data or address switches. An unequal compare causes Process Check without turning on a CPU check indicator on the CE console.

Storage Scan (STOR SCAN)

In this mode, the CPU starts at the address indicated on the register data or address switches when the start key is operated.
It scans through main storage (from low to high) until a parity error is detected or until the stop key is operated.
Since during each single scan operation, the main storage address is incremented by 1, the scan continues with address /0000/ after the highest possible address, which can be expressed in halfword (/ FFFF /) has exceeded. Storage Scan operates in the protected area, customer area, and control area. Note that also not available storage positions are scanned.

Storage Fill (STOR FILL)

In this mode a byte set up in data switches 1 and 2 is moved into all positions of the customer area in main storage. Storage Fill starts at the address indicated by the register data or address switches when the start key is operated.

The fill operation proceeds from the low-address positions to the high-address positions with all possible addresses valid (/0000/ to /FFFF/). However, filling is suppressed when the address specifies a byte position in the protected area or in the control area.
Due to the address incrementing by 1, the fill operation continues with address /0000/ after the highest possible address has exceeded.
Storage Fill stops when the stop key is operated.

Storage Fill mode and the CE mode switch on allows the CE to alter also the content of storage positions in the protected area and the control area (micro-program is destroyed and re-loading is required).

Storage Fill mode and the CE Storage Test switch on causes storage test operation, which affects the total main storage.
Storage Test is executed by four different

runs during which each core storage half-word position is loaded by its own address. These loaded addresses are read out again and compared with the origin address.

An unequal compare causes Process Check and the CPU is stopped. Since the address is loaded once true and once inverted all bits are tested. During the invert, the CPU runs operated with inverted parity which additionally tests the CPU check circuits.

When the stop key is operated, Storage Test stops at the end of the true compare run to ensure that data in main storage is stored with correct parity.

Control Program Load (CPL)

The mode switch in this position and the and the load key operated causes reading of the micro-program into the control area. The micro-program card deck is read by the main card read device which is also used for loading customer programs (see load key description).

When the Load key is operated, the procedure starts with System Reset (CPU LOG in, System Reset routine). After System Reset the Load Latch and the mode switch is tested and a branch to the micro-program (control program) load routine is performed.

This load routine is stored in a restricted part of the control area and controls loading of the micro-program. All necessary informations like load address and read device specification are provided by the micro-program load routine itself. No additional switch settings are required. Note that the described procedure, which is applicable by the customer, requires the availability of a control program in main storage.

The initial control program load (ICPL) can be performed by the CE only. For ICPL, the mode switch must also be set to CPL and the load key has to be operated. However, the CE meter key switch must be set into CE position additionally.

All necessary informations like start address of load field (register data or address switches), ICPL device address (data switches 1 and 2) ICPL control data (CE Select switch 2), and field length (CE Select switches 3 and 4) must be set up.

Operating the load key initiates the circuit controlled reading (MANOP ICPL) of the first card (control program loader) containing micro-program load routine by the selected I/O device. After the first card has been read (field length below zero), the load routine is executed as for customer control program load.

Register Data or Address Switches

- The four rotary switches are used to set up addresses or data of half-word format.
- Each of the 16-position switches can express any hexadecimal digit (half byte digit) from /0/ to /F/.

- The applications of the four switches are given in the mode switch description.

Data Switches

- The two rotary switches are used to set up data bytes, addresses of General registers (data switch 1 only), and, for LS display or alter, LS zone (data switch 1) and LS register (data switch 2) specifications.
- Each switch can be set to any hexadecimal digit from /0/ to /F/.
- The applications of the two switches are given in the mode switch description.

Time-Sharing Switch

- When this switch is on, the execution of input/output operations is time shared with other CPU operations.
- When the switch is off, each input/output operation is completely executed before the CPU continues with the next sequential machine instruction (burst mode).

Lamp Test Switch

- This switch may be employed to visually check for faulty indicator lamps.
- All indicators (CPU and I/O's) should be on when the lamp test switch is operated. (except: Power Attention Indicators and indicators on 2311, 2415, 1419 or 1259).

Emergency Power-Off Switch (EPO)

- In an emergency, this switch may be pulled to immediately disconnect all power from the system.
- The switch must be manually reset by the customer engineer before power may be restored to the system.

Meter Lock-Switch

- The meter lock-switch switches the customer meter and the CE meter.
- To perform ICPL, the switch must be set to the CE position.
- To run Machine Function Tests (MFT's) or the Impulse Check Routine (ICR), the meter lock-switch must be in the CE position.

INDICATORS

P, I, U, L Display

- This display indicates the To-operand (halfword) supplied to the ALU at the Shift Unit output.

E, S, T, R Display

- This display indicates the From-operand (halfword) supplied to the ALU at the Invert Switch output.

Attention Indicators

- The Attention indicators show the particular condition or the particular I/O device that has caused the system to stop.
- The operator is guided quickly to the respective device and an inspection of its console defines the stop condition in detail.

Process

This indicator is turned on as the comprehensive indication of any CPU check (detailed indication on the CE console) by the Process Check Latch active. All operations are halted immediately. The process check condition can be reset by operating System Reset Key, Load key, or CE CPU Check Reset switch.

Line Failure

This indicator shows the line voltage provided to the power supply failed. This can be a voltage drop or a missing phase (50 Hz only). Power can be restored by first pressing System Reset Key and than the Power On Key.

Power

This indicator shows any abnormal power supply condition like over-or under voltages, tripped circuit breakers or circuit protectors as well as line failures or thermal condition in connection with the corresponding attention indicators. Power can be restored after normal power supply conditions exists again, by pressing system Reset key and than the Power On key.

Thermal

This indicator shows , in connection with the Power attention indicator that an overheating condition exists either in one of the CPU gates or in one of the attached I/O devices. For restoring power see Power indicator description above.

Printer

This indicator shows an unusual condition in either the IBM 2203 Printer or the IBM 1403 Printer. These conditions can be mis-firing of hammers, sync check, or forms check.

SIOC

This indicator shows that either the serial I/O channel or the device that operates through this channel such as the IBM 1419 or 1259 Magnetic Character Reader has an error condition. For example, a jam or sort check in the 1419 causes the SIOC-light to come on.

Card I/O 1

This indicator shows an unusual condition in the IBM 2501 Card Reader. These conditions can be a filled stacker, an empty hopper, an open machine cover, a read check, a feed check, or a machine check. Machine checks usually require action from a Customer Engineer.

Card I/O 2

This indicator shows an unusual condition in either the IBM 2560 Multi-Function Card Machine or in the IBM 2520 Reader/Punch (whichever is attached). The condition can be a filled stacker, empty hopper, open machine cover, feed check, read check, punch check as well as machine check.

Card I/O 3

This indicator relates to the IBM 1442 Card Punch, Model 5. Conditions such as feed check, empty hopper, full stacker, etc., cause this indicator to light.

2152

This indicator shows an unusual condition in the IBM 2152 Printer-Keyboard. These conditions can be an end of forms condition, a P1 check or a P2 check, a typewriter power supply failure or disconnected cables.

ST CTRL

This indicator shows that either the storage control feature or the IBM 2311 disk storage drive has an error condition.

IOC

This indicator shows that either the input/output channel or the device that operates through this channel has an error condition. For example, a programming error that involves the IBM 2415 Magnetic Tape Units, or an interface parity error causes the IOC-light to come on.

CE CONSOLE

SWITCHES

CE Mode Switch

- The CE mode switch on and the mode switch in position Display or Alter Register allows display or alter operations of LS registers.
- CE mode switch on prevents upper and lower address check and halfword boundary check for Storage Alter or Fill operations.
- Preventing of address checks, which causes regeneration of data read out, allows alter or fill of storage positions in the protected area and control area.
- CE mode switch on illuminates the CE mode indicator.

Single Cycle Switch

- Single Cycle Switch on prevents Hold Run condition and thus, delta process and process latches drop again after one delta cycle or cycle performed.
- Only one processing period is performed for each start key operation.

Single Micro-Instruction Switch

- Single Micro-Instruction switch on allows performing of one single micro-instruction for each start key operation.
- Allow PL Switching condition, which defines the last cycle in the micro-instructions, drops Hold Run Condition (delta process and process drop, CPU stops).
- For ALC operations, Allow PL Switching occurs in the last cycle of each repetition. Thus, the start key has to be operated as much as repetitions are specified by the field length to terminate ALC operation.

Compare Equal Stop Switch

- Display data, the source of which is selected by the CE Display/Compare Select switch, are compared with the data set up in the CE Select switches.
- The compare equal (Address Equal Bits 0 to 15) and the Compare Equal Stop switch on drops Hold Run Condition (CPU stops).
- The compare equal also provides the signal ' - EQUAL SYNC' at T4 (start delta cycle) which may be used for scoping.

Storage Test Switch

- Storage Test switch on and the mode switch in position Storage Fill allow

to execute storage test, when the start key is operated.

Initial Load Loop Switch

- This switch on causes continuous reading of punched cards by the circuit controlled ICPL operation.
- CPU stop by any I/O error is prevented.
- Continuous ICPL is used for trouble shooting if control program loading fails.

Invert Parity Switch

- This switch on and the mode switch in position Storage Scan or Storage Fill turns on the Invert Parity Latch.
- CPU circuits, able to generate parity bits, operate in inverted mode, that means, parity bits are generated to provide even instead of odd byte parity.
- All parity depending CPU check Latches (SAR check, ALU check, Inhibit check, etc.) must turn on.
- A Process Check (CPU stop) occurs when not all check latches turn on either by missing or additional bits (parity correct ; odd) , or if a check latch fails to turn on.
- The inverted operations affect only the customer storage area if the CE mode switch is off.
- To prevent Process Check after Scan or Fill with inverted parity (main storage data are of even parity) run Storage Fill with invert parity switch off and with Process Check Override switch on.

Process Check Override Switch

- This switch on allows CPU to function as when no Process Check is on.
- Especially for ICPL and CPL, the control program is loaded as normal, but the data read in may be incorrect.

CPU Check Reset Key

- Operating this push button directly resets the CPU check latches.
- Since no other reset functions are performed, all check informations are maintained and can be investigated (displayed) for trouble shooting.

CPU Reset Key

- Operating this push button causes a reset operation, which can be con-

sidered as System Reset in single micro-instruction mode.

- Run control and Long Time Clock are set into their initial start situations.

- CPU checks are reset and the four cycle CPU LOG in operation is performed.

Lamps Switch

- This switch on allows to illuminate the indicators on the CE console.

Block Feed Check Switch

- This switch on prevents feed check conditions in the attached card I/O devices.

- The Block Feed Check signal is directly applied to the several I/O boxes (not used in the attachments).

1403 and BSCA CE Switches

- The functions of these switches are given in the corresponding FETOM's or FEMDM's.

Display/Compare Select Switch

- The Display/Compare Select switch selects CPU registers, several CPU control conditions (OCU 1 or 2), or the common I/O bus (data bus P1 to 15) for display on the CE console.

- The display data provided by the selected source may be used for Compare Equal Stop or Equal Synch.

I/O Display

- The Display/Compare Select switch in this position allows display of Sense data bytes during CPU run or stop by the display bits P1 to 15.

- The device address of the sense byte has to be set up in the CE Select switches 1 and 2 (Sense Address Select).

- The device address is directly (not from the address bus) displayed in display bits 0 to 7 (no parity bit P0).

- Compare Equal Stop and Equal Synch can be performed if the CE select switches 3 and 4 are set according to the sense byte wait for.

- For compare and synch all possible sense bits must be taken into consideration.

- Sense sources in the I/O attachments as well as in the CPU can be selected.

- Only the loworder byte of halfword CPU senses is displayed (SENS/14/, register data or address switches).

- Note that CPU sense data is displayed without parity.

CE Select Switches

- The four CE Select switches are used to set up data (byte/halfword) or addresses needed for CE operations (Compare Equal Stop, Equal Synch, I/O display) or during ICPL.

INDICATORS

CE Mode Indicator (red)

- This indicator is on when the CE mode switch is in on position.

- It signals the CE that any Fill or Alter operation may destroy important data in the protected area and in the control area (Storage Fill or Alter) or in the Local store (Alter Register)

CPU Check Indicators (red)

- These indicators provide detailed check information if a Process Check occurs (attention indicator).

- The indicators turn on when the corresponding check latch is activated.

- Note that a Process Check can occur without any CPU check indicator on if ALU output is not zero during Storage Test (Compare Runs) or LS Register Alter.

- During Storage Fill or Scan with inverted parity the Process Check occurs, when not all CPU check latches (except BUS and LSA checks) are activated.

- This indicator on specifies a local store addressing check caused if none or more than one X-address or Select LS-Y signal is active simultaneously.

MOD

- This indicator on specifies incorrect parity in the highorder or loworder modifier result byte.

- Modifier parity bits are predicted depending upon the halfword data in MAR and the modifier operation to be executed.

SU

- This indicator on specifies incorrect parity in the highorder or loworder byte of the TDR halfword after Eight Shift (INT SU bits P1 to 15, Shift Unit entry, and data bus output).

ALU

- This indicator on specifies incorrect parity in the highorder or loworder ALU result byte.
- The ALU parity is predicted depending upon the parity of the two ALU input operands (TDR halfword after Eight Shift and Invert Switch halfword) and is corrected according to the shift by 2 or 4 operation, the Suppress operation, and ALU operation performed.

BUS

- This indicator on specifies incorrect parity on the data bus or in the low-order byte at ALU output during I/O operations (SENS or CTRL micro-instructions.

SAR

- This indicator on specifies incorrect parity in the highorder or loworder SAR byte.

INH

- This indicator on specifies incorrect parity in the highorder or loworder byte of the halfword just written into main storage.

Delta Cycle Indicators

- The two indicators display binary the delta cycle condition after CPU stop.
- Note that always the delta cycle is display which should be performed as the next.

Cycle Indicators

- The two indicators display binary the process cycle performed before CPU stops.

Any I/O Busy Indicator

- This indicator indicates that any attached I/O device (also BSCA) is operating.

Any Feed Cell Dark Indicator

- This indicator on indicates that any feed cell in one of the attached card I/O's is covered by a punched card or that a feed cell fails to operate (defective lamp or defective solar cell).

LS Zone Indicators

- The three indicators display binary the Program Level (PL) stored in the Current PL Register.

- Note that the New PL can be displayed if the Display/Compare Select switch is set to OCU 2.

CE Display Bits P0 to 15 Indicators

- These indicators can display any half-word data, the source of which is selected by the Display/Compare Sesect switch.

- Especially for I/O Display the indicators P1 to 15 (I/O Sense Display) display the data bus conditions, set by a source which is selected by the device address (CE Select switches 1 and 2).

MISCELLANEOUS CE CONSOLE COMPONENTS

CPU Remote Control Socket

- The CE remote control box can be attached to the socket.

- The remote control box is equipped with start key, stop key, and system reset key.

- When the remote control box is attached,the start key on the customer console is blocked.

Missing Phase Button

- The missing phase button is forced out to indicate that a line phase was missing for more than 5 seconds.

- The button has only indicator functions and must be restored manually for renewed indication.

MAINTENANCE FEATURES

CPU CHECKS

- CPU Checks are :
  - Local Store Addressing Check (LSA)
  - Modifier Check (MOD)
  - Shift Unit Check (SU)
  - ALU Check (ALU)
  - Data Bus in Check (BUS)
  - Storage Address Register Check (SAR)
  - Inhibit Check (INH)
- Any of these checks causes Process Check (CPU stops).
- The check indication is described in section CE Console.
- A detailed representation of all check latches is given in FEMDM, 2020 CPU, Chapter 2, Error Conditions.

CPU LOG IN

- CPU LOG in is a four cycle operation controlled by circuits.
- CPU LOG in is always performed before the micro-program controlled System Reset Routine (System Reset, Power on Reset, Load Key operated).
- During each LOG cycle , one halfword is stored into main storage.
- The main storage addresses are forced by circuits (SAR powering).
- The halfwords to be logged (one during each cycle) are provided by the CE display bits P0 to 15.
- To select different sources of LOG in data, switching of the Display/ Compare Select switch is simulated by logical circuits depending upon cycle time (figure 6-1).
- Especially during CPU LOG in cycle 0 (OCU 1 to Display), the halfword to be logged is combined as shown in figure 6-2.
- During the last LOG in cycle (CY3), the CPU checks are reset (LOG Reset) and the start address of the System Reset Routine is forced into IAR of LS zone 7.
- The LOG area consists mainly of four adjacent ranges. Each range contains four logged halfwords.
- During System Reset Routine, the four ranges are shifted by four halfwords, but only if any CPU check condition exists in the first halfword just logged.
- Always the three last LOG informations are retained.
- The contents of the missing phase byte (additional part of the LOG area) is shifted by one bit (low to high) and the

LOG in Cycles		Simulated Switching	Remarks
CYØ		IOCU 1 to Display	Parity bits PØ and P1 are generated during Read time
CY1		IOPREG to Display	
CY2	LSA or MOD Check	MAR to Display	The provided halfword is investigated for correct parity during Read time (no INH Check). During Write time the data is replaced by /ØØ/ for the correct byte or /FF/ for the incorrect byte. OPREG to Display, SAR to Display, and MAR to Display CY3 provide the normal data. However, the normal data is replaced by /ØØ/ and/or/FF/ if the parity is incorrect.
	SU Check	TDR to Display	
	ALU or BUS Check	FDR to Display	
	SAR Check	SAR to Display	
	INH Check	SDR to Display	
CY3		IMAR to Display	During CY3 MAR contains IAR of the current LS zone

Figure 6-1 Display Switching during CPU LOG in

loworder bit is turned on if the missing phase latch in the CPU is active.

- The LOG area can be displayed (Storage Display in CE mode) or printed out by a special machine program.

CE VOLTMETER

- The CE voltmeter, mounted on a separate panel on front of the power supply, allows the CE quick checking of all generated DC voltages.
- The voltages to be tested are selected by a rotary switch.
- Beside the logical voltages, all important core storage voltages can be measured.

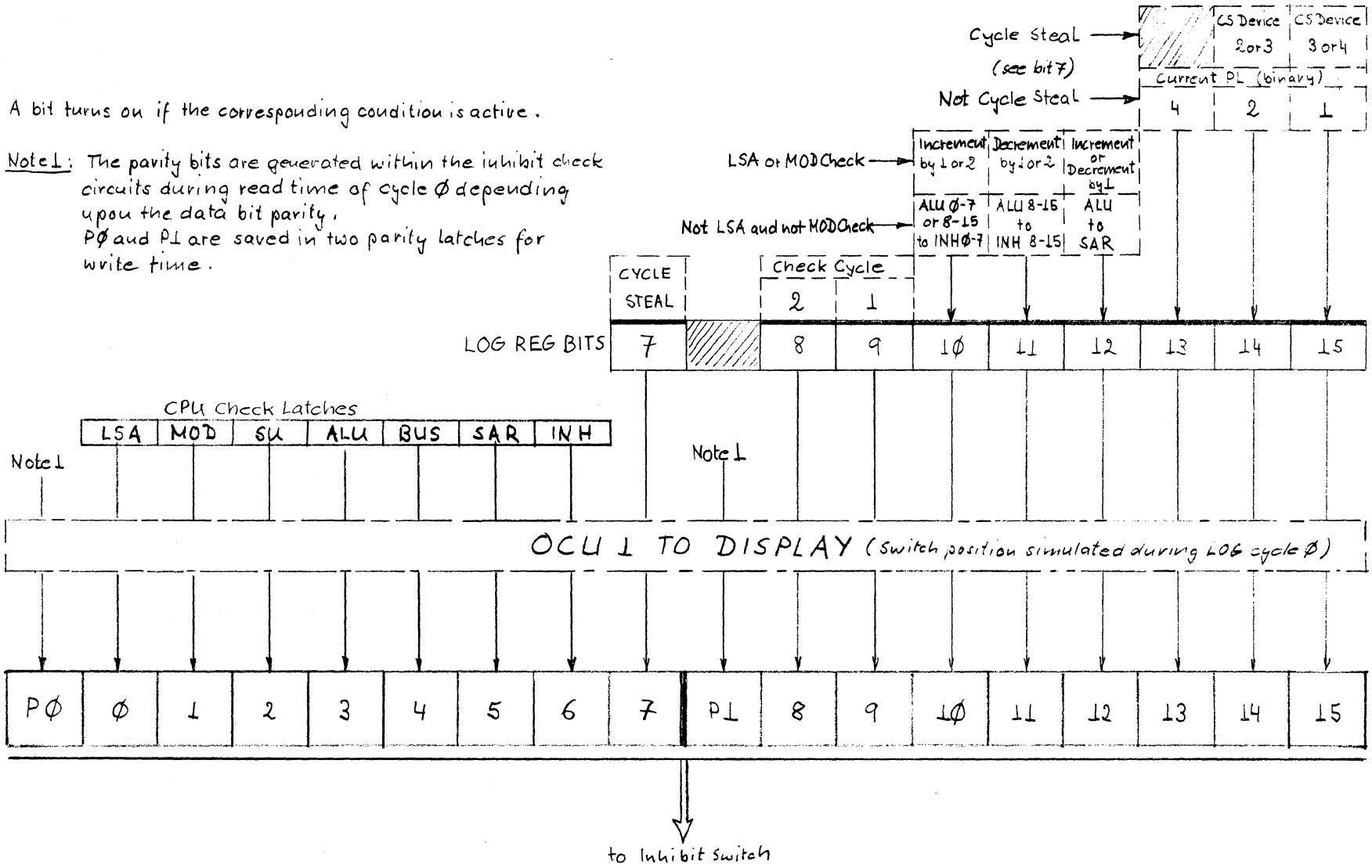


Figure 6-2 First CPU LOG in Halfword (LOG cycle Ø)

READER'S COMMENT FORM

2020 Processing Unit System/360 Model 20
(Machines with serial no. 50,000 and above)

Form Z33-1021-0

How did you use this publication?

As a reference source ☐
As a class-room text ☐
As a self-study text ☐

Based on your own experience, rate this publication:

As a reference source—Very Good ☐ Good ☐ Fair ☐ Poor ☐ Very Poor ☐
As a text—Very Good ☐ Good ☐ Fair ☐ Poor ☐ Very Poor ☐

What is your occupation?

We would appreciate your specific comments; please give page and line references where appropriate. If you wish a reply, be sure to include your name and address.

● Thank you for your cooperation. No postage necessary if mailed in U.S.A.

Z33-1021-0

YOUR COMMENTS PLEASE . . . .

This Field Engineering manual is one of a series which serves as reference sources for customer engineers. Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM. Please note: requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

fold fold

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

IBM CORPORATION

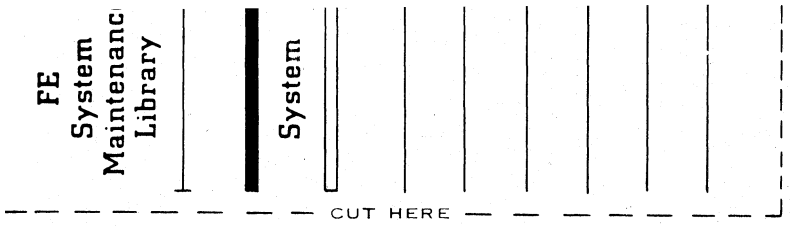
112 EAST POST ROAD,
WHITE PLAINS, N.Y. 10601.

Attention: Publishing Operations

fold fold

IBM®

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]



Z33-1021-0

Printed in U.S.A.

Z33-1021-0

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601