

2 INTRODUCTION

2.0 FOREWORD

The C-446 Service Manual has two basic objectives:

- A. To be used as a textbook under the guidance of a qualified instructor;
- B. To be used as a reference manual.

Whenever possible, pictures are used to eliminate confusion through words. **BE SURE TO READ EACH STATEMENT CAREFULLY TO AVOID MISUNDERSTANDING.**

2.1 GENERAL DESCRIPTION

2.1.1 Construction

The C-446 is an electronic billing machine consisting of four basic units: the Program Tape Reader, the Printer, the Control Keyboard and the Electronics. The machine has a thirteen word storage capacity. Each word can contain up to twelve digits, the Exponent and the Sign bit. A basic breakdown of the four units mentioned above is outlined below.

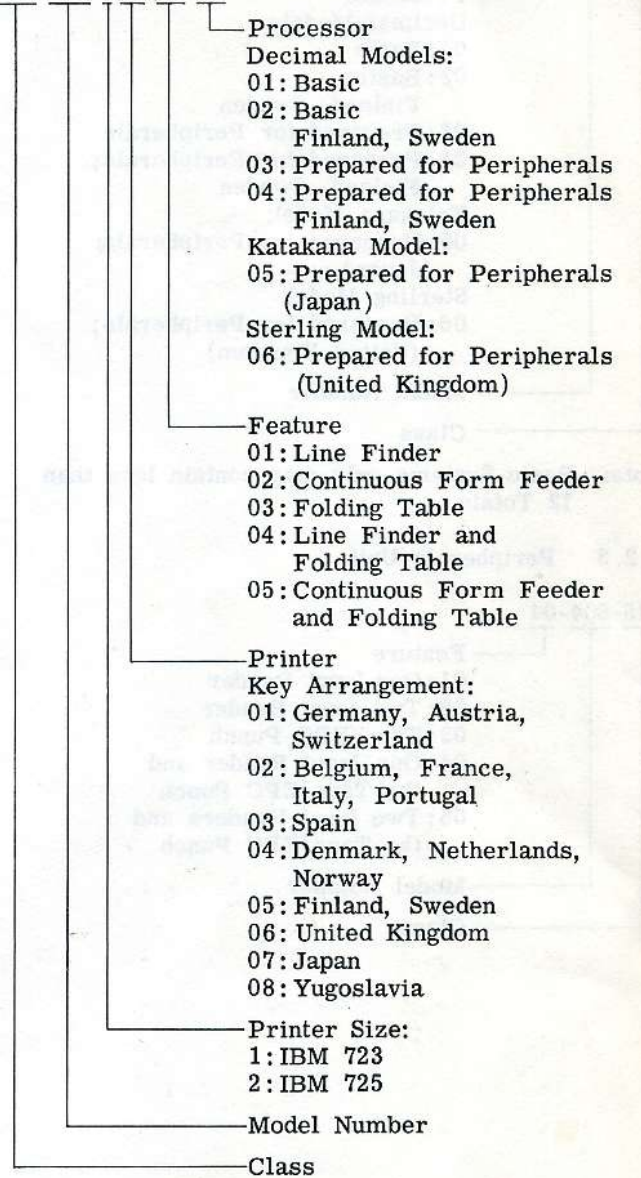
- A. Program Tape Reader
The Program Tape Reader is mounted on the front of the Electronics. This unit reads tapes and controls the Printer and Electronics. The Reader consists of a light source, a stepping motor and a transport table.
- B. Printer
The Printer is an electro-mechanical device consisting of an IBM Typewriter, an Electrical Control and Answer Back Switches. The Printer is designed to operate at fifteen characters per second.
- C. Control Keyboard
The Control Keyboard is an electro-mechanical unit with 43 keys. These keys are divided into two groups - Function Keys and Data Keys. The Power ON-OFF light and the Overflow light are mounted on this unit. There are seven read-out switches in the Control Keyboard.
- D. Electronics
The Electronics is a fully transistorized (solid state) unit. It consists of twenty-four printed circuit plug-in cards, a power supply, and a magnetic core assembly. The wire wrap method is used to interconnect these units. All computations are performed within this unit under control of the Program Tape Reader and the Control Keyboard.

2.2 SIZE NUMBER EXPLANATION

Following is the Size Number Explanation for the Class 446 System.

2.2.1 Console

446-017-202-04-03



The Core Memory Project

2.2.2 Electronics Unit

446-509-03-12

Number of Totals:

12: 12 + Register H

08: 8 + Register H

04: 4 + Register H

Processor

Decimal Models:

01: Basic

02: Basic;

Finland, Sweden

03: Prepared for Peripherals

04: Prepared for Peripherals;

Finland, Sweden

Katakana Model:

05: Prepared for Peripherals;

(Japan)

Sterling Model:

06: Prepared for Peripherals;

(United Kingdom)

Model Number

Class

Note: Basic Systems only may contain less than 12 Totals.

2.2.3 Peripherals Unit

446-804-04

Feature

01: One Input Reader

02: Two Input Reader

03: Tape/EPC Punch

04: One Input Reader and
the Tape/EPC Punch

05: Two Input Readers and
the Tape/EPC Punch

Model Number

Class

2.2.4 C-487 EPC Reader

487-101

Model Number

Class

3 PROGRAMING

3.0 INTRODUCTION

Programing of the C-446 Billing Machine is performed by reading an eight channel punched tape. The Program controls:

- A. The Logic and Registers
- B. The Keyboard (Data Input) and its functions
- C. The Printer (Data Output) and its functions
- D. The Program Reader

3.0.0 General Description

Several Main Programs may be on one Program Tape. Each Main Program includes 16 Sub-programs. Every Sub-program can be divided by up to 10 Skip programs. The Program Tape is transported step by step from left to right. To be able to search a certain Sub- or Skip-program the sequence of both programs is controlled by internal counters.

The Control Keyboard is used for Data Input (numeric) and for program selection. The Printer is used for programed and/or manual Data Output (alpha-numeric).

3.1 PROGRAM SEQUENCE CONTROL

The Program Sequence may be controlled by reading Field Symbols from the tape or by using the Control Keyboard.

3.1.1 Program Limitations

To limit the program and its sub-sections, some Field Symbols are used. A Field Symbol also contains the field modifiers (UC, FW).

- A. Begin of Field (BOF) Symbols:
 - 1. Begin of Program - BOP.
Every Program Tape, in the sample with one Main Program, has to have the following start and end:
BOP 0 IFP PSR END BOP---Main Program
---BOP 0 IFP PSL END.
 - 2. Begin of Sub-program - BOS.
Every Main Program must have 15 BOS.
For Sub-programs not used, a dummy BOS (BOS-HLT-Sprocket hole) must be programed. A BOS read by TSS will cause a TRT to the start of this Sub-program without affecting the Printer. A BOS read by TRT or TRN will update the internal SP counter and stop the Program Tape when the wanted Sub-program is found.
NOTE: SP15 always contains the 0 V routine.
 - 3. Skip Symbol - SKS.
Every Sub-program may have up to 9 SKS.

The Core Memory Project

A SKS read by TSS will be ignored but updates the internal Skip Counter. A SKS read by TRT or TRN, caused by a Skip command, will stop the Reader when the wanted Skip-program is reached.

4. **Begin of Column Symbol - BOC.**
The BOC symbols may be used to divide any Skip-program into several different FW and/or UC columns. By depressing the TAB Key, the Printhead will TAB, and due to TRN the Program Tape will be stopped at the next BOF.

B. Field Definition Symbols:

1. **Field Width - FW.**
The FW, as a part of a BOF symbol, is limited to a maximum of 12 digits (1.2 inch). Three additional spaces are required for the Decimal Point (DP), the Sign and for a 2nd symbol.
2. **Upper Case Symbol - UC.**
The UC symbol, as a part of a BOF symbol, is needed if the 2nd symbol is on the upper case hemisphere of the Printhead.

3.1.2 Control Keyboard

The Control Keyboard is used for data input (numeric) and for Program selection. It is divided into the Data Keyboard and the Function Keyboard. The green Control Light indicates power on; the red Control Light indicates an Overflow condition.

- A. The functions of the Data Keyboard Keys are:
 1. **Amount Keys:** Perform numeric data input.
 2. **Decimal Point Key:** Set the decimal point position. A programed DP position will be deleted.
 3. **Activating Bar (AB):** Resume program.
 4. **Minus Activating Bar (MAB):** Resume program and negate the input.
 5. **The % and ‰ Keys:** Affect the arithmetic operation and prints the symbol.
 6. **Clear Key:** Clears Input Register I; If an Overflow condition exists, it will be released. This key can not be locked.
- B. The Data Keyboard will be released by a Program Halt (HLT) or a Data Keyboard Release (DKR) code.

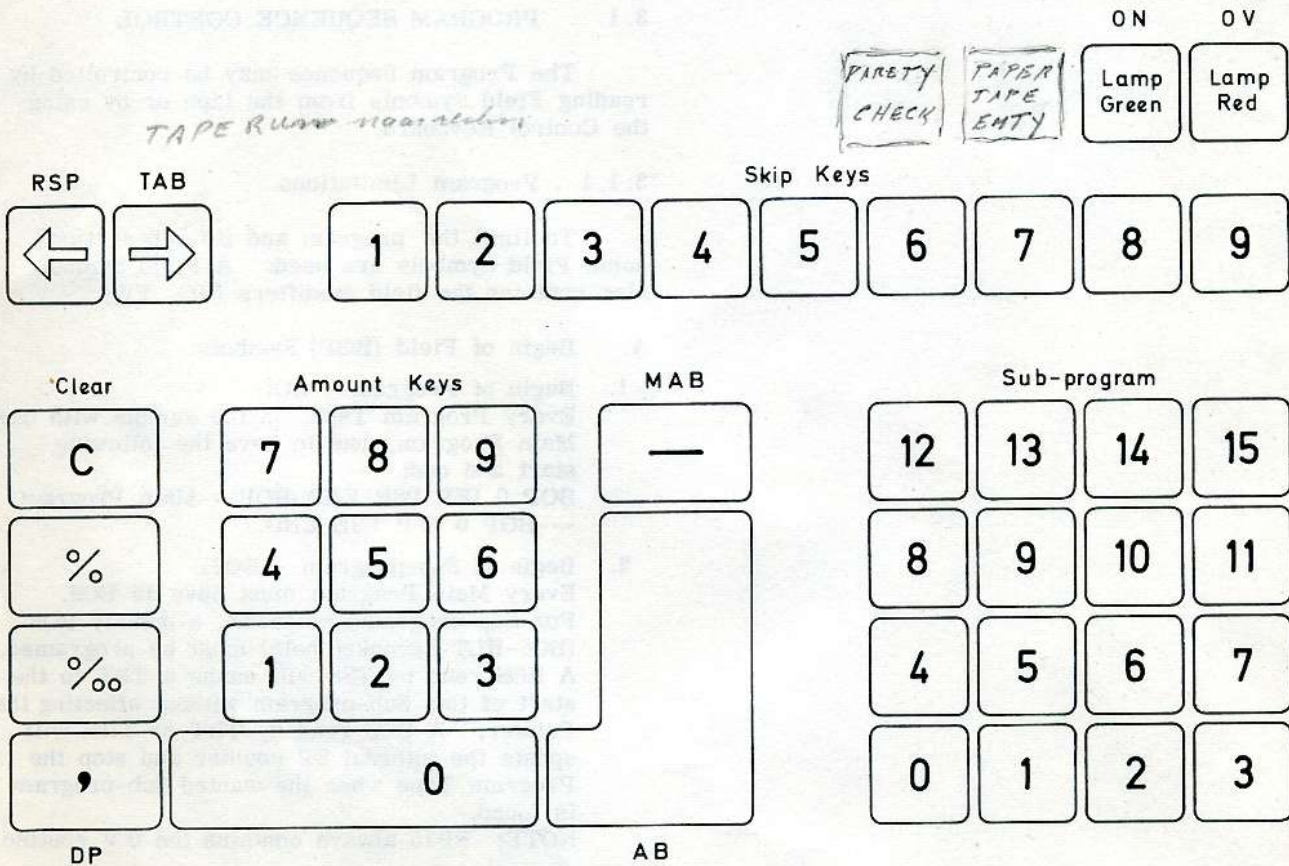


Figure 3-1. Control Keyboard Layout

The Core Memory Project

- A. If a MLT is performed, Register B includes the content of n2. After ADD, SUB and DIV the content of Register B is changed.
- B. To change the sign of an operand a SUB can be programed with a dummy address zero in n1.
- C. During DIV, the number of digits of the result (quotient) is equal to 12 minus the number of positions of the divisor after matching n2 with n1. See example.

D. Example:

| | | | | |
|-------------|---|-------------|---|-----------------|
| n1 | : | n2 | = | C |
| 3 | : | 3 | = | 1.0 000 000 000 |
| 3,0 | : | 3 | = | 1.000 000 000 |
| 300 000 000 | : | 300 000 000 | = | 1.00 |
| any figure | : | 0 | = | Overflow |

3.2.2 Copy

The content of the n1 Register will be copied into the n2 Register.

| | | |
|-----------------------------|----------|-------------------------|
| 0•B•C•I•H•S ₁₋₁₂ | CPY | B•I•H•S ₁₋₁₂ |
| n1 | Function | n2 |

- A. A zero in n1 will clear the n2 Register.
- B. Register C contains the former content of Register B after CPY is performed.

3.2.3 Compare for Magnitude

The content of the n1 Register will be compared with the content of Register B.

| | | | |
|---------------------------|----------|------------------|-----|
| B•C•I•H•S ₁₋₁₂ | CFM | any IFF function | End |
| n1 | Function | n2 | |

- A. When the content of the Register in n1 is smaller than content of Register B, the programed function (n2) will be performed. (Comparison is fulfilled).
- B. When the content of the n1 Register is equal to or greater than content of Register B, n2 will be ignored.
- C. Register B will be:
1. cleared with unequal signs.
 2. changed with unequal exponents.
 3. unchanged with positive signs, equal exponents and $n1 \geq B$.
 4. a copy of n1 Register with positive signs, equal exponents and $n1 < B$.
 5. unchanged with negative signs, equal exponents and $n1 < B$.

6. a copy of n1 Register but reversed sign with negative signs, equal exponents and $n1 \geq B$.
- D. Register C will be cleared if $n1 < B$ but contains the difference of n1 minus B, if $n1 \geq B$.
- E. Examples:

| n1 | B | C | B | n1 < B |
|-----|-----|----|-----|--------|
| 18 | 12 | 6 | 12 | no |
| 18 | -12 | 30 | 0 | no |
| -18 | 12 | 0 | 0 | yes |
| -18 | -12 | 0 | -12 | yes |
| 12 | 12 | 0 | 12 | no |
| 12 | 18 | 0 | 12 | yes |

before CFM

after CFM

3.2.4 Shift Decimal Point

The decimal Point of the n1 Register will be shifted by the programed number in n2. (Up-count of the exponent).

| | | |
|---------------------------|----------|------|
| B•C•I•H•S ₁₋₁₂ | SDP | 0-15 |
| n1 | Function | n2 |

- A. If an IFC was read and the % or ‰ Key was depressed, the programed number of shifts (n2) will be ignored and a shift to the left by two (%) or three (‰) will be performed.
- B. To shift the decimal point to the right, the number of wanted right shifts has to be subtracted from 16 and the result put into n2.
- C. Examples:

| n1 | n2 | C |
|---------|----|---------|
| 2 | 2 | 0.02 |
| 2.3 | 1 | 0.23 |
| 3.45678 | 13 | 3456.78 |

NOTE: Exponent of the n1 Register plus number in n2 = new exponent.

3.2.5 Shift

The number of decimal point positions of the n1 Register will be modified by the number in n2.

| | | |
|---------------------------|-----|------|
| B•C•I•H•S ₁₋₁₂ | SHF | 0-11 |
|---------------------------|-----|------|

- A. If the number in n2 is larger than the exponent of the n1 Register, the new decimal point positions are filled up with zeros.
- B. If n2 contains a number 16 through 27 a multiplication by 10^{n2-16} is performed.

C. Examples:

$$22 - 16 = 6$$

| n1 | n2 | C |
|----------|----|-----------|
| 2.3456 | 2 | 2.34 |
| 1.2 | 3 | 1.200 |
| 2.3456 | 0 | 2 |
| 2 | 22 | 2 000 000 |
| 2.222222 | 22 | 2 222 222 |

3.2.6 Round off and Shift

The number of decimal point positions of the n1 Register will be modified by the number in n2 and the amount will be rounded.

| B·C·I·H·S ₁₋₁₂ | RSH | 0-11 |
|---------------------------|----------|------|
| n1 | Function | n2 |

A. If the number in n2 is larger than the exponent of the n1 Register, the new decimal point positions are filled up with zeros.

B. Example:

| n1 | n2 | C |
|-------|----|------|
| 2.555 | 2 | 2.56 |
| 2.554 | 2 | 2.55 |
| 2 | 2 | 2.00 |
| 2.55 | 0 | 3 |

3.2.7 Input from Control Keyboard

The exponent of Register I will be added to the number in n2 or by the % or %₀₀ Key. This command has to be programmed after any Overlapping Input or IFP. If IFC is used to index data, a Halt (HLT) has to be programmed.

| HLT | 0 | IFC | 0-12 |
|-----|----------|-----|------|
| n1 | Function | n2 | |

- A. After IFC is performed, the Input Control Counter is set and Register C is cleared.
- B. A read HLT symbol releases the Data Keys and interrupts the program sequence to allow data indexing. Only AB or MAB resumes program.
- C. If more than 12 digits are indexed, the 0 V Light will be turned on and the Keyboard will be locked. To release the 0 V Lock, the Clear Key has to be depressed.

3.2.8 Output from Storage

The content of the n1 Register will be printed.

| B·C·I·H·S ₁₋₁₂ | OFS | Step·2nd symbol·function |
|---------------------------|----------|--------------------------|
| n1 | Function | n2 |

- A. If the exponent is zero, a space occurs first.
- B. According to the programed FW a Zero Suppression is performed.
- C. If the sign is negative, a minus will be printed after the last digit is printed. If the sign is positive, a space occurs after the last digit is printed.
- D. A "second symbol" print can be programed in n2. If no 2nd symbol is wanted a Step has to be programed.
- E. If an IFC was read and the % or %₀₀ Key was depressed, the programed n2 will be ignored and the % or %₀₀ symbol is printed.

3.2.9 Input from Program

This command allows data (constants) to be read into Register I and the performing of all Tape Control functions.

| 0 | IFP | data·function | End |
|----|----------|---------------|-----|
| n1 | Function | n2 | |

A read IFP will lock the Function Keyboard and clear the AB, % and %₀₀ bits. An End symbol has to be programmed after n2.

3.2.10 Output from Program

This command allows printing of all Printer characters, performance of all Printer functions and releasing and locking of the Keyboards.

| 0 | OPF | Printer characters · functions | End |
|----|----------|--------------------------------|-----|
| n1 | Function | n2 | |

An End symbol has to be programmed after n2.

3.2.11 Table of Commands

The C-446 performs 13 different commands. See figure 3-3.

3.3.2 Program Sheet

The Programming Sheet points out the program sequence and control, all field modifications, and inputs/outputs of Program Tape. A sample of a Program Sheet is shown in figure 3-8.

- D. During SP1, in this sample, a 2.1 is indexed (OL Input). See figure 3-9.
- Flow No. 07 : prepares Flow No. 13 (B = 1).

| NCR 446 | | PROGRAM SHEET | | | | | | Page:4/7 SP: 2 SK: 1 | | |
|------------|-------------|---------------|----|---------|----|-----|----|----------------------------|--------|--------|
| No. | Field Symb. | | | Command | | | | n2 when OFF/IFP and CFM | D K | F K |
| | Type | FW | UC | H | n1 | f | n2 | | | |
| 01 | BOS | 5 | X | | 10 | CPY | 2 | | R | L |
| 07 | | | | X | 0 | IFC | 3 | | | |
| 11 | | | | | 0 | IFP | | DKL 1000 SP4 END | L | |
| | | | | | | | | | | |

Figure 3-8. Program Sheet Sample

- The digits in the "No." column indicate the consecutive count of the code combinations (Program Flow Number).
- The "Field Symb." columns contain all Field modifications.
- Columns "H" and "U" contain a "x" if programmed.
- The content of the last two columns indicate the Data Keyboard and the Function Keyboard are locked (L) or released (R).

NOTE: The S for Storage is left out in columns n1 and n2.

3.3.3 Sample Program

This Sample Program is only a part (SP2) of a Main Program.

- Five different items, item codes .0 through .4, are selected by this Sub-program.
- Each item may have a different quantity. Both codes are indexed at once.
 - Example: Index 8.3 = Item 3 with a quantity of 8.
 - A zero for quantity is handled as a single item.
- Storage usage:
 - Storage of quantity : S1.
 - Storage of item code : S2.
 - Constant : S10 = 1.

- Flow No. 10 : manual, to wait until the Input is finished.
- Flow No. 13 : if no quantity is indexed, TAB to Flow No. 36.
- Flow No. 21, 24 : change I = 2.1 into S1 = 2 (quantity only).
- Flow No. 27, 30 : S2 = 0.1 (item code only).
- Flow No. 33 : print quantity.
- Flow No. 41 : manual to type text. (AB to resume program).
- Flow No. 46, 49 : if no quantity is indexed, go to Flow No. 53 to copy a quantity of 1.
- Flow No. 56 : I = 0.4.
- Flow No. 63, 66 : if an illegal item code (>.4) is indexed, go to SP4.
- Flow No. 74, 80 : change S2 = 0.1 into I = 1.
- Flow No. 83 : go to SK3, if item code .0 is indexed.
- Flow No. 87 : go to SK4, if item code .1 is indexed.
- Flow No. 91 : go to SK5, if item code .2 is indexed.
- Flow No. 95 : go to SK6, if item code .3 is indexed. go to SK2, if item code .4 is indexed.

Remember: Register C contains contents of n1 Register minus contents of Register

The Core Memory Project

B after a CFM is performed. Register B content is not changed until the comparison is fulfilled.

| NCR 446 | | PROGRAM SHEET | | | | | | | Page: 4/7 SP : 2 SK : 1 | | |
|------------|-------------|---------------|---|---------|----|-----|-----|-------------------------|-------------------------------|--------|--------|
| No. | Field-Symb. | | | Command | | | | n2 when OFF/IFP and CFM | P K | D K | F K |
| | Type | FW | U | H | n1 | f | n2 | | | | |
| 01 | BOS2 | 4 | | | 0 | OFF | | PKL,END | L | | L |
| 07 | | | | | 10 | CPY | B | | | | |
| 10 | | | | X | 1 | CPY | 2 | | | | |
| 13 | | | | | 1 | CFM | | TAB,END | | | |
| 17 | | | | | 0 | OFF | | DKL,END | | L | |
| 21 | | | | | 1 | SHF | 0 | | | | |
| 24 | | | | | C | CPY | 1 | | | | |
| 27 | | | | | 2 | SUB | 1 | | | | |
| 30 | | | | | C | CPY | 2 | | | | |
| 33 | | | | | 1 | OPS | PHT | | | | |
| 36 | BOC | | | | 0 | OFF | | PKR,END | R | | |
| 41 | | | | X | 0 | OFF | | PHT,PKL,END | L | | |
| 46 | | | | | 1 | CPY | B | | | | |
| 49 | | | | | 0 | CFM | | SKI,END | | | |
| 53 | | | | | 10 | CPY | 1 | | | | |
| 56 | SK 1 | | | | 0 | IFP | | CLK,DPT,4,END | | | |
| 63 | | | | | 2 | CPY | B | | | | |
| 66 | | | | | 1 | CFM | | SP4,END | | | |
| 70 | | | | | 0 | IFP | | CLK,END | | | |
| 74 | | | | | 2 | CPY | I | | | | |
| 77 | | | | | 10 | CPY | B | | | | |
| 80 | | | | | 1 | SDP | 15 | | | | |
| 83 | | | | | C | CFM | | SK3,END | | | |
| 87 | | | | | C | CFM | | SK4,END | | | |
| 91 | | | | | C | CFM | | SK5,END | | | |
| 95 | | | | | C | CFM | | SK6,END | | | |
| 99 | SK 2 | 5 | | | 0 | IFP | | CLK,END | | | |

Figure 3-9. Sample Program Sheet

3.3.4 Flow Chart Symbols

The following list shows the Flow Chart symbols used by the programmers when programming the C-446. See figure 3-10.

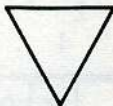




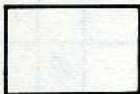






| | |
|---|---|
|  | HALT Manual Decision |
|  | Operation of: AB or MAB, Sub-program or Skip key |
|  | Begin of Sub-program |
|  | Skip Symbol |
|  | Input from Data Keyboard |
|  | ADD, SUB, MLT, DIV, CPY, SDP, SHF, RSH |
|  | Compare for Magnitude |
|  | Input from Program |
|  | Manual Printer Operation (after Printer Keyboard Release) |
|  | Output from Program (to Printer) |
|  | Output from Storage (Print) |
|  | Connectors |

Figure 3-10. Flow Chart Symbols

3.4 PROGRAMING FOR PERIPHERALS

3.4.0 INTRODUCTION

Programing of the C-446 Basic and Peripheral Models is basically identical. Several additional functions and/or command modifiers are used to control the peripheral Units:

- A. Input Reader A or B (Data Input).
- B. Tape and Edge Punched Card Recorder (Data Output).
- C. Continuous Form Feeder.
- D. Line Finder.

NOTE: The EPC Reader has been designed C-487.

3.4.1 GENERAL DESCRIPTION

To select peripheral units, some of the input/output commands are modified and three new commands are added.

3.4.1.1 Modified Commands

- A. Input from Program (IFP):
To read-in data from Tape or Edge Punched Card by the Data Reader A or B.
- B. Output from Program (OFP):
 - 1. To print, or print and punch, or punch data read by the Program Reader.
 - 2. To print, or print and punch, or punch data read by the Data Reader A or B.
- C. Output from Storage (OFS):
 - 1. To print, or print and punch, or punch (C-446 code).
 - 2. To print (C-446 code) and/or punch the selected code.

3.4.1.2 New Commands

- A. Block Selection Reader A (BSA) and B (BSB).
To select any "Block of Data" from the Data Tape or Edge Punched Card. *255 data block*
- B. Punch from Printer Keyboard (PPK).
To generate an Edge Punched Card (E.P.C.) or a Data Tape by using the Printer Keyboard.
- C. Continuous Form Feeder
To activate the C.F.F. an additional Printer function is used.
- D. Line Finder
There are 4 functions to control the Line Finder with Program Tape.

3.4.2 PUNCHED DATA

The Tape/EPC Punch will punch 8 channel Tape or Edge Punched Cards (E.P.C.). The data recording media may be read in by the two Data Readers. *allen B*

3.4.2.1 Data Tape and E.P.C. Limitations

To allow selection of a certain "Block of Data" and to define the start of a Data Tape or E.P.C. two Field Symbols are used.

- A. Begin of Tape Symbol (BTS)
 - 1. Every Data Tape or E.P.C. has to be started with a BTS.
 - 2. A BTS read by a high speed Tape selection will stop the Data Reader.
- B. Begin of Block Symbol (BBS)
 - 1. Every Block of Data has to be started with a BBS. (Block # 0 through # 255).
 - 2. A BBS read by a high speed Tape selection will update the internal Block counter and stop the Data Tape or E.P.C. when the wanted Block is found.
 - 3. To avoid erroneous Block selection (high speed) a second END must be punched in front of a BBS.
 - 4. For unused Blocks in front of the highest used Block, a dummy Block (BBS - END - END - BBS) must be programmed.
 - 5. The start of an E.P.C. is defined by the Positioning Hole.

3.4.2.2 Numeric Punch Codes

Due to the OFS variations the content of n1 Register will be either printed or printed and punched or punched. The punch of the numeric NCR C-446 Code or one selected numeric code depends on the type of OFS Command. Parity check is performed for the numeric C-446 Code and selected numeric codes. To release a parity check failure (white Control Light on) the Machine has to be switched off and on again.

The Core Memory Project

A. Non Selected Code

1. NCR C-446 Code, odd parity punch (channel 7) and check is performed.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
|------|---|---|---|---|---|---|---|---|------------------------|
| | | | | | | | | | 0 |
| | | | | | | | | | 1 |
| | | | | | | | | | 2 |
| | | | | | | | | | 3 |
| | | | | | | | | | 4 |
| | | | | | | | | | 5 |
| | | | | | | | | | 6 |
| | | | | | | | | | 7 |
| | | | | | | | | | 8 |
| | | | | | | | | | 9 |
| | | | | | | | | | minus |
| | | | | | | | | | DP |
| | | | | | | | | | %o |
| | | | | | | | | | %oo |
| | | | | | | | | | FSP (Leading zeros,37) |
| step | | | | | | | | | FSP (1.Symbol,33) |
| step | | | | | | | | | FSP (2.Symbol,36) |

Figure 3-11

B. Selected Codes

If the % and/or %o symbol is not available in certain codes another special code will be punched.

1. NCR C-315 Code, odd parity punch (channel 7) and check is performed.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
|------|---|---|---|---|---|---|---|---|------------------------|
| | | | | | | | | | 0 |
| | | | | | | | | | 1 |
| | | | | | | | | | 2 |
| | | | | | | | | | 3 |
| | | | | | | | | | 4 |
| | | | | | | | | | 5 |
| | | | | | | | | | 6 |
| | | | | | | | | | 7 |
| | | | | | | | | | 8 |
| | | | | | | | | | 9 |
| | | | | | | | | | minus |
| | | | | | | | | | %o |
| | | | | | | | | | %oo |
| | | | | | | | | | FSP (Leading zeros,37) |
| plus | | | | | | | | | FSP (1.Symbol,33) |
| | | | | | | | | | FSP (2.Symbol,36) |

Figure 3-12

2. IBM Code 046, with parity punch and check.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
|--|---|---|---|---|---|---|---|---|-------------|
| | | | | | | | | | 0 |
| | | | | | | | | | 1 |
| | | | | | | | | | 2 |
| | | | | | | | | | 3 |
| | | | | | | | | | 4 |
| | | | | | | | | | 5 |
| | | | | | | | | | 6 |
| | | | | | | | | | 7 |
| | | | | | | | | | 8 |
| | | | | | | | | | 9 |
| | | | | | | | | | minus |
| | | | | | | | | | Punch (%o) |
| | | | | | | | | | Punch (%oo) |

Figure 3-13

3. ISO Code (used for C-615), even parity punch and check is performed in channel 8.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
|--|---|---|---|---|---|---|---|---|--------|
| | | | | | | | | | 0 |
| | | | | | | | | | 1 |
| | | | | | | | | | 2 |
| | | | | | | | | | 3 |
| | | | | | | | | | 4 |
| | | | | | | | | | 5 |
| | | | | | | | | | 6 |
| | | | | | | | | | 7 |
| | | | | | | | | | 8 |
| | | | | | | | | | 9 |
| | | | | | | | | | minus |
| | | | | | | | | | %o |
| | | | | | | | | | M(%oo) |

Figure 3-14

3.4.2.3 Special Punch Codes (C-446)

Only channel 1 through 7 code combinations are duplicable (ORA1-2 or ORB1-2). A channel 8 hole, a sprocket hole only, an Ignore Code and STEP will be ignored.

Reading the following codes will cause a conversion:

- "Punch END Symbol" (PES) - an END will be punched.
- "Punch Block Symbol" (PBS) - a BBS will be punched.
- "Punch Begin of Tape Symbol" (PBT) - a BTS will be punched.
- "Enter Edge Punch Card Symbol" (EPC) - Data Tape: one sprocket hole will be punched. E. P. C. : continuous sprocket holes will be punched until the Positioning Hole is found.

| CODE | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | CODE | | |
|------|---|---|---|---|---|---|---|---|-----|------------|------------|-----|
| 120 | | | | | | | | | PES | will punch | END | 27 |
| 123 | | | | | | | | | PBS | will punch | BBS | 128 |
| 114 | | | | | | | | | PBT | will punch | BTS | 129 |
| 112 | | | | | | | | | EPC | will punch | Spr. Holes | 0 |

Figure 3-15

3.4.2.4 Data Tape and E.P.C. Format

At least three zeros as "Run-in" and nine zeros as "Run-out" have to be programmed. The zeros allow duplicating the whole E.P.C.

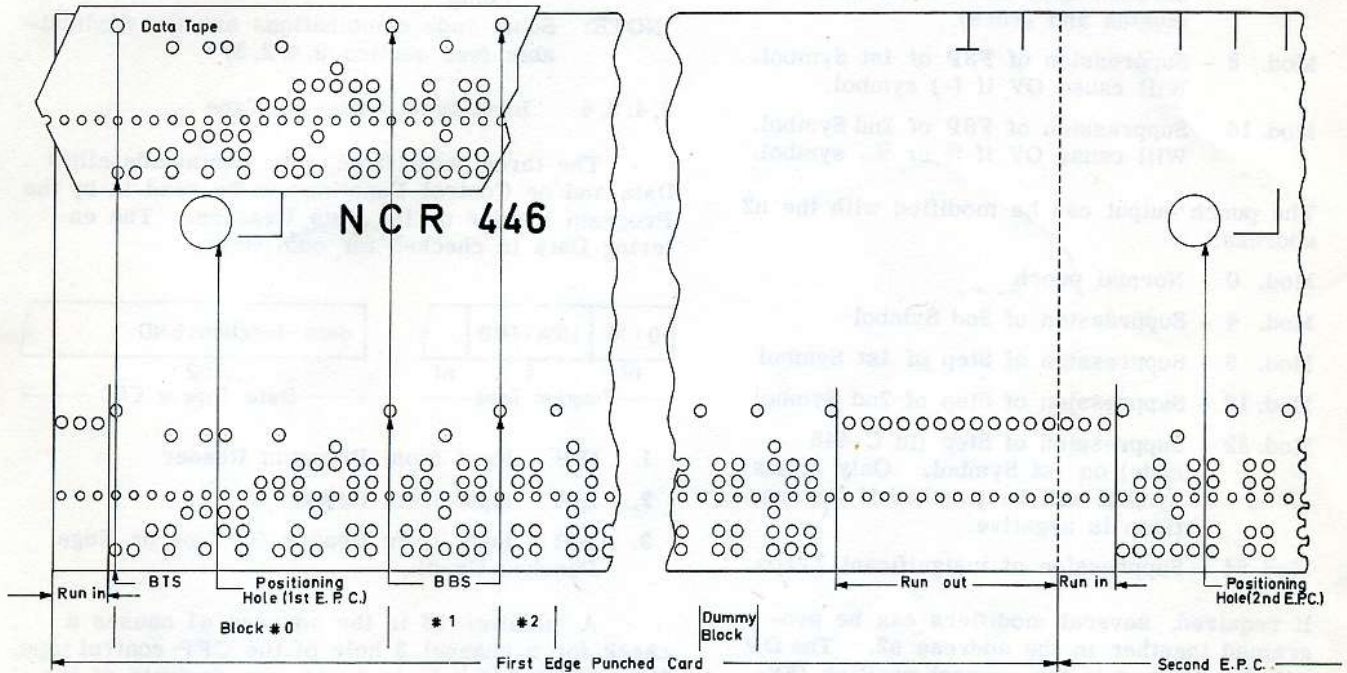


Figure 3-16. Data Tape and E.P.C. Format

3.4.2.5 CFF Control Loop

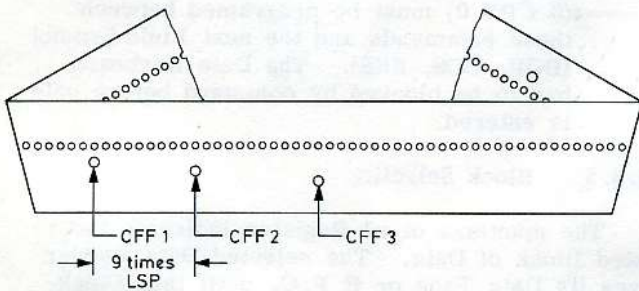


Figure 3-17. CFF Control Loop

3.4.3 COMMAND STRUCTURE

The internal command sequence is not changed. On some input/output commands the n_2 address is located in the Data Tape or E.P.C. If the address zero is programmed, a sprocket hole only (code zero) must be punched.

3.4.3.1 Copy

With $n_2 = H$ an OV will occur if:

- A. the mantissa of $n_1 > 1048575$;
- B. the exponent of $n_1 \neq 0$;
- C. the mantissa of n_1 is negative.

NOTE: Only bit 1 through bit 20 of Register H can be used for storage.

3.4.3.2 Output from Storage

The contents of the n_1 Register will be printed and/or punched. The address n_2 is used by modifiers and does not allow a programmed 2nd Symbol or function. The UC-Symbol is not usable anymore. Printed and/or punched as 2nd Symbol will be only the % or %o code and/or FSP/Step.

- A. The five OFS variations are:
 1. OFS - print.
 2. OFS1 - print and punch C-446 code.
 3. OFS2 - punch C-446 code.
 4. OFS3 - print C-446 code and punch selected code, do not punch DP.
 5. OFS4 - punch selected code, do not punch DP.

B. The print output can be modified with the n2 address:

1. Mod. 0 - Normal print
2. Mod. 1 - Protective print
3. Mod. 2 - Date print (6 numbers with FSP's between days and months and months and years)
4. Mod. 8 - Suppression of FSP of 1st Symbol. Will cause OV if (-) symbol.
5. Mod. 16 - Suppression of FSP of 2nd Symbol. Will cause OV if % or ‰ symbol.

C. The punch output can be modified with the n2 address:

1. Mod. 0 - Normal punch
2. Mod. 4 - Suppression of 2nd Symbol
3. Mod. 8 - Suppression of Step of 1st Symbol
4. Mod. 16 - Suppression of Step of 2nd Symbol
5. Mod. 32 - Suppression of Step (in C-446 code) on 1st Symbol. Only minus symbol will be punched if the mantissa is negative.
6. Mod. 64 - Suppression of insignificant Zeros.

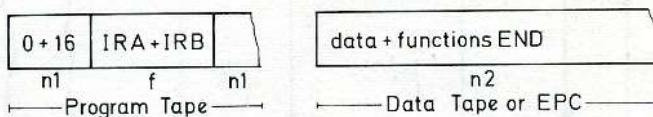
If required, several modifiers can be programmed together in the address n2. The DP will be punched in the correct position (Exception: OFS3 and OFS4). An OV-Condition will be set up if the programmed FW was exceeded.

6. ORA2 - Output from Data Tape A, punch only.
7. ORB - Output from Data Tape B, print only.
8. ORB1 - Output from Data Tape B, print and punch
9. ORB2 - Output from Data Tape B, punch only.

NOTE: Some code combinations are not duplicatable (see section 3.4.2.3)

3.4.3.4 Input from Program Tape

The three Input from Tape commands allow Data and/or Control Functions to be read in by the Program Reader or the Data Readers. The entering Data is checked for odd parity.



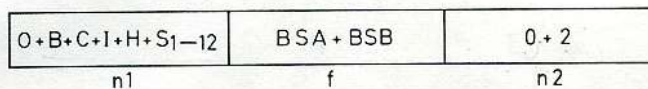
1. IFP - Input from Program Reader
2. IRA - Input from Reader A
3. IRB - Input from Reader B (Tape or Edge Punched Card)

A modifier 16 in the address n1 causes a check for a channel 3 hole of the CFF control tape. If no channel 3 hole is found, the contents of the address n2 will be ignored. Care must be taken that no vertical paper movement occurs while executing this command with modifier 16.

NOTE: If a Program Tape Selection is started (IRA or IRB) at least a dummy command (0 CPY 0) must be programmed between these commands and the next Field Symbol (BOP, BOS, SKS). The Data Keyboard has to be blocked by command before data is entered.

3.4.3.5 Block Selection

The mantissa of n1 Register indicates the wanted Block of Data. The selected Data Reader moves its Data Tape or E.P.C. until this Block is found.



- A. With n1 = 0 and n2 = 2 the Data Tape or Edge Punched Card moves (RTB) until a BTS is found.
- B. An OV occurs if:
 1. the mantissa > 255
 2. the n1 Register includes an exponent or a sign bit.

NOTE: For Blocks not used, a dummy Block must be programmed (see 3.4.2.1B4)

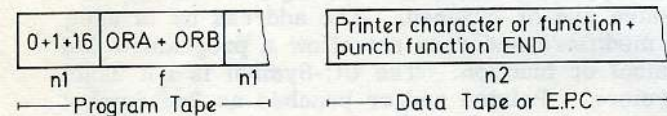
3.4.3.3 Output from Program Tape

All printer characters can be printed and/or punched with nine Output from Tape commands.

A modifier 1 in address n1 will translate the C-446 code into selected code.

A modifier 16 in the address n1 will cause a parity check of all punched characters.

A switch on the tape recorder selects even or odd parity.



1. OFP - Output from Program Tape, print only.
2. OFP1 - Output from Program Tape, print and punch.
3. OFP2 - Output from Program Tape, punch only.
4. ORA - Output from Data Tape A, print only.
5. ORA1 - Output from Data Tape A, print and punch.

The Core Memory Project

3.4.3.6 Punch from Printer Keyboard

With a released Printer and Data Keyboard all Printer Characters and functions can be punched.

| | | |
|------------------------------|-----|------------|
| 0+B+C+I+H+S ₁ -12 | PPK | 0+1+2+4+16 |
| n1 | f | n2 |

The mantissa of n1 Register indicates the Field length (max. length 255 characters).

The specified n1 length will be punched. All insignificant positions will be punched with the "Ignore" Code. For n2 modifiers see table 3-18.

For each UC character the PHS Key has to be depressed individual. Depression of the AB-Bar terminates the command.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|----------------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE |
| • | • | • | • | • | • | • | • | Ignore code 26 |

3.4.3.7 Additional Functions

The following OFP-ORA-ORB functions activate the peripheral Units.

- A. Data Tape and Edge Punched Card selection.
 1. SSA or SSB - performs a Data Reader TSS.
 2. RNA/RTA or RNB/RTB - highspeed movement until a BTS or BBS is found and updates the internal Block counter.
 3. Enter Punched Card (EPC) - feeds-in the E.P.C. until the Positioning Hole is found.

- B. Continuous Form Feed Selection
 - CFF1, CFF2 or CFF3 - feeds-in the Form until the Loop Hole # 1, # 2 or # 3 (channel 4-6) is found.

NOTE: Test for channel 4 hole in IFP when called for by a modifier.

- C. Line Finder Selection
 1. LFI - in-feeds the Ledger Card.
 2. LFE - ejects the Ledger Card.
 3. LFS - performs a L. F. Line Space.
 4. LFM - marks the Ledger Card.

PROGRAMING CODES

verifying non all codes reading below

OUTPUT FROM PROGRAM

21 Red Rec-R-5

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CARACTERS | CODE |
|----------------|---|---|---|---|---|---|---|-----------|------|
| 0 | . | . | . | . | . | . | . | / | 95 |
| 1 | . | . | . | . | . | . | . | € | 127 |
| 2 | . | . | . | . | . | . | . | " | 87 |
| 3 | . | . | . | . | . | . | . | + | 119 |
| 4 | . | . | . | . | . | . | . | \$ | 79 |
| 5 | . | . | . | . | . | . | . | & | 111 |
| 6 | . | . | . | . | . | . | . | | 71 |
| 7 | . | . | . | . | . | . | . | | 103 |
| 8 | . | . | . | . | . | . | . | - | 107 |
| 9 | . | . | . | . | . | . | . | ~ | 75 |
| A | . | . | . | . | . | . | . | ∅ | 124 |
| B | . | . | . | . | . | . | . | | 84 |
| C | . | . | . | . | . | . | . | c | 116 |
| D | . | . | . | . | . | . | . | | 76 |
| E | . | . | . | . | . | . | . | | 108 |
| F | . | . | . | . | . | . | . | | 68 |
| G | . | . | . | . | . | . | . | ~ | 100 |
| H | . | . | . | . | . | . | . | | 104 |
| I | . | . | . | . | . | . | . | ° | 72 |
| J | . | . | . | . | . | . | . | # | 126 |
| K | . | . | . | . | . | . | . | ⊙ | 86 |
| L | . | . | . | . | . | . | . | | 118 |
| M | . | . | . | . | . | . | . | m | 78 |
| N | . | . | . | . | . | . | . | | 110 |
| O | . | . | . | . | . | . | . | | 70 |
| P | . | . | . | . | . | . | . | ⊗ | 102 |
| Q | . | . | . | . | . | . | . | | 106 |
| R | . | . | . | . | . | . | . | ∅ | 74 |
| S | . | . | . | . | . | . | . | ∅ | 85 |
| T | . | . | . | . | . | . | . | ‰ | 117 |
| U | . | . | . | . | . | . | . | * | 77 |
| V | . | . | . | . | . | . | . | ∅ | 109 |
| W | . | . | . | . | . | . | . | ‰ | 69 |
| X | . | . | . | . | . | . | . | ⊠ | 101 |
| Y | . | . | . | . | . | . | . | ≠ | 105 |
| Z | . | . | . | . | . | . | . | ≠ | 73 |
| Å | . | . | . | . | . | . | . | ∅ | 66 |
| Ä | . | . | . | . | . | . | . | Å | 67 |
| Ö | . | . | . | . | . | . | . | Å | 93 |
| = | . | . | . | . | . | . | . | : | 125 |
| , | . | . | . | . | . | . | . | ? | 65 |
| - | . | . | . | . | . | . | . | | 94 |
| — | . | . | . | . | . | . | . | II | 92 |
| · | . | . | . | . | . | . | . | ! | 64 |
| DKR | . | . | . | . | . | . | . | | 16 |
| DKL | . | . | . | . | . | . | . | | 17 |
| FKR,DKR,OV.off | . | . | . | . | . | . | . | | 20 |
| PKR | . | . | . | . | . | . | . | | 21 |
| PKL | . | . | . | . | . | . | . | | 22 |
| LON | . | . | . | . | . | . | . | | 24 |
| PHR | . | . | . | . | . | . | . | | 40 |
| BSP | . | . | . | . | . | . | . | | 42 |
| LSP | . | . | . | . | . | . | . | | 44 |
| PHT | . | . | . | . | . | . | . | | 48 |
| FSP | . | . | . | . | . | . | . | | 32 |
| PHS | . | . | . | . | . | . | . | | 34 |
| STP | . | . | . | . | . | . | . | | 31 |
| LF1 | . | . | . | . | . | . | . | | 54 |
| LFE | . | . | . | . | . | . | . | | 30 |
| LFS | . | . | . | . | . | . | . | | 52 |
| LFM | . | . | . | . | . | . | . | | 53 |
| CFF 1 | . | . | . | . | . | . | . | | 49 |
| CFF 2 | . | . | . | . | . | . | . | | 51 |
| CFF 3 | . | . | . | . | . | . | . | | 50 |
| SSA | . | . | . | . | . | . | . | | 3 |
| SSB | . | . | . | . | . | . | . | | 5 |
| RNA | . | . | . | . | . | . | . | | 12 |
| RNB | . | . | . | . | . | . | . | | 13 |
| RTA | . | . | . | . | . | . | . | | 10 |
| RTB | . | . | . | . | . | . | . | | 11 |
| PES | . | . | . | . | . | . | . | | 120 |
| PBS | . | . | . | . | . | . | . | | 123 |
| PBT | . | . | . | . | . | . | . | | 114 |
| EPC | . | . | . | . | . | . | . | | 112 |
| END | . | . | . | . | . | . | . | | 27 |

FW CODES

| | BOP | BOS | SKS | BOC |
|-------|-----|-----|-----|-----|
| FW 0 | 224 | 192 | 160 | 128 |
| 1 | 225 | 193 | 161 | 129 |
| 2 | 226 | 194 | 162 | 130 |
| 3 | 227 | 195 | 163 | 131 |
| 4 | 228 | 196 | 164 | 132 |
| 5 | 229 | 197 | 165 | 133 |
| 6 | 230 | 198 | 166 | 134 |
| 7 | 231 | 199 | 167 | 135 |
| 8 | 232 | 200 | 168 | 136 |
| 9 | 233 | 201 | 169 | 137 |
| 10 | 234 | 202 | 170 | 138 |
| 11 | 235 | 203 | 171 | 139 |
| FW 12 | 236 | 204 | 172 | 140 |

FIELD SYMBOLS PROGRAM TAPE

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE | |
|---|---|---|---|---|---|---|---|------------|-----|
| . | . | . | . | . | . | . | . | BOP | 224 |
| . | . | . | . | . | . | . | . | BOS | 192 |
| . | . | . | . | . | . | . | . | SKS | 160 |
| . | . | . | . | . | . | . | . | BOC | 128 |
| . | . | . | . | . | . | . | . | UCS(Basic) | 16 |
| . | . | . | . | . | . | . | . | FW 1 | 1 |
| . | . | . | . | . | . | . | . | 2 | 2 |
| . | . | . | . | . | . | . | . | 3 | 3 |
| . | . | . | . | . | . | . | . | 4 | 4 |
| . | . | . | . | . | . | . | . | 5 | 5 |
| . | . | . | . | . | . | . | . | 6 | 6 |
| . | . | . | . | . | . | . | . | 7 | 7 |
| . | . | . | . | . | . | . | . | 8 | 8 |
| . | . | . | . | . | . | . | . | 9 | 9 |
| . | . | . | . | . | . | . | . | 10 | 10 |
| . | . | . | . | . | . | . | . | 11 | 11 |
| . | . | . | . | . | . | . | . | 12 | 12 |

ADDRESS

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE | |
|---|---|---|---|---|---|---|---|------------|----|
| . | . | . | . | . | . | . | . | Storage 1 | 1 |
| . | . | . | . | . | . | . | . | 2 | 2 |
| . | . | . | . | . | . | . | . | 3 | 3 |
| . | . | . | . | . | . | . | . | 4 | 4 |
| . | . | . | . | . | . | . | . | 5 | 5 |
| . | . | . | . | . | . | . | . | 6 | 6 |
| . | . | . | . | . | . | . | . | 7 | 7 |
| . | . | . | . | . | . | . | . | 8 | 8 |
| . | . | . | . | . | . | . | . | 9 | 9 |
| . | . | . | . | . | . | . | . | 10 | 10 |
| . | . | . | . | . | . | . | . | 11 | 11 |
| . | . | . | . | . | . | . | . | 12 | 12 |
| . | . | . | . | . | . | . | . | Register C | 13 |
| . | . | . | . | . | . | . | . | Register B | 14 |
| . | . | . | . | . | . | . | . | Register I | 15 |
| . | . | . | . | . | . | . | . | Register H | 16 |
| . | . | . | . | . | . | . | . | 0 | 0 |
| . | . | . | . | . | . | . | . | HLT | 32 |

PARITY CHECK: PERIPHERAL

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE | |
|---|---|---|---|---|---|---|---|---------------|----|
| . | . | . | . | . | . | . | . | Digit 0 | 64 |
| . | . | . | . | . | . | . | . | 1 | 1 |
| . | . | . | . | . | . | . | . | 2 | 2 |
| . | . | . | . | . | . | . | . | 3 | 67 |
| . | . | . | . | . | . | . | . | 4 | 4 |
| . | . | . | . | . | . | . | . | 5 | 69 |
| . | . | . | . | . | . | . | . | 6 | 70 |
| . | . | . | . | . | . | . | . | 7 | 7 |
| . | . | . | . | . | . | . | . | 8 | 8 |
| . | . | . | . | . | . | . | . | 9 | 73 |
| . | . | . | . | . | . | . | . | Decimal point | 82 |
| . | . | . | . | . | . | . | . | Minus | 19 |
| . | . | . | . | . | . | . | . | ‰ | 84 |
| . | . | . | . | . | . | . | . | ‰ | 21 |

INPUT FROM PROGRAM

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE | |
|---|---|---|---|---|---|---|---|---------------|----|
| . | . | . | . | . | . | . | . | Digit 0 | 64 |
| . | . | . | . | . | . | . | . | 1 | 1 |
| . | . | . | . | . | . | . | . | 2 | 2 |
| . | . | . | . | . | . | . | . | 3 | 3 |
| . | . | . | . | . | . | . | . | 4 | 4 |
| . | . | . | . | . | . | . | . | 5 | 5 |
| . | . | . | . | . | . | . | . | 6 | 6 |
| . | . | . | . | . | . | . | . | 7 | 7 |
| . | . | . | . | . | . | . | . | 8 | 8 |
| . | . | . | . | . | . | . | . | 9 | 9 |
| . | . | . | . | . | . | . | . | Decimal point | 18 |
| . | . | . | . | . | . | . | . | Minus | 19 |
| . | . | . | . | . | . | . | . | ‰ | 20 |
| . | . | . | . | . | . | . | . | ‰ | 21 |
| . | . | . | . | . | . | . | . | CLK | 24 |
| . | . | . | . | . | . | . | . | Skip 0 | 32 |
| . | . | . | . | . | . | . | . | 1 | 33 |
| . | . | . | . | . | . | . | . | 2 | 34 |
| . | . | . | . | . | . | . | . | 3 | 35 |
| . | . | . | . | . | . | . | . | 4 | 36 |
| . | . | . | . | . | . | . | . | 5 | 37 |
| . | . | . | . | . | . | . | . | 6 | 38 |
| . | . | . | . | . | . | . | . | 7 | 39 |
| . | . | . | . | . | . | . | . | 8 | 40 |
| . | . | . | . | . | . | . | . | 9 | 41 |
| . | . | . | . | . | . | . | . | TAB | 43 |
| . | . | . | . | . | . | . | . | RSP | 44 |
| . | . | . | . | . | . | . | . | PSL | 46 |
| . | . | . | . | . | . | . | . | PSR | 47 |
| . | . | . | . | . | . | . | . | SP 0 | 48 |
| . | . | . | . | . | . | . | . | 1 | 49 |
| . | . | . | . | . | . | . | . | 2 | 50 |
| . | . | . | . | . | . | . | . | 3 | 51 |
| . | . | . | . | . | . | . | . | 4 | 52 |
| . | . | . | . | . | . | . | . | 5 | 53 |
| . | . | . | . | . | . | . | . | 6 | 54 |
| . | . | . | . | . | . | . | . | 7 | 55 |
| . | . | . | . | . | . | . | . | 8 | 56 |
| . | . | . | . | . | . | . | . | 9 | 57 |
| . | . | . | . | . | . | . | . | 10 | 58 |
| . | . | . | . | . | . | . | . | 11 | 59 |
| . | . | . | . | . | . | . | . | 12 | 60 |
| . | . | . | . | . | . | . | . | 13 | 61 |
| . | . | . | . | . | . | . | . | 14 | 62 |
| . | . | . | . | . | . | . | . | 15 | 63 |
| . | . | . | . | . | . | . | . | STP | 31 |

COMMANDS

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | COMMANDS | CODE |
|---|---|---|---|---|---|---|---|----------|------|
| . | . | . | . | . | . | . | . | ADD | 24 |
| . | . | . | . | . | . | . | . | SUB | 25 |
| . | . | . | . | . | . | . | . | MLT | 28 |
| . | . | . | . | . | . | . | . | DIV | 26 |
| . | . | . | . | . | . | . | . | CFM | 29 |
| . | . | . | . | . | . | . | . | RSH | 71 |
| . | . | . | . | . | . | . | . | SHF | 70 |
| . | . | . | . | . | . | . | . | CPY | 9 |
| . | . | . | . | . | . | . | . | SDP | 14 |
| . | . | . | . | . | . | . | . | IFC | 6 |
| . | . | . | . | . | . | . | . | OFS | 12 |
| . | . | . | . | . | . | . | . | OFS 1 | 46 |
| . | . | . | . | . | . | . | . | OFS 2 | 47 |
| . | . | . | . | . | . | . | . | OFS 3 | 62 |
| . | . | . | . | . | . | . | . | OFS 4 | 63 |
| . | . | . | . | . | . | . | . | IFP | 50 |
| . | . | . | . | . | . | . | . | IRA | 18 |
| . | . | . | . | . | . | . | . | IRB | 34 |
| . | . | . | . | . | . | . | . | OFF | 116 |
| . | . | . | . | . | . | . | . | OFF 1 | 118 |
| . | . | . | . | . | . | . | . | OFF 2 | 119 |
| . | . | . | . | . | . | . | . | ORA | 84 |
| . | . | . | . | . | . | . | . | ORA 1 | 86 |
| . | . | . | . | . | . | . | . | ORA 2 | 87 |
| . | . | . | . | . | . | . | . | ORB | 100 |
| . | . | . | . | . | . | . | . | ORB 1 | 102 |
| . | . | . | . | . | . | . | . | ORB 2 | 103 |
| . | . | . | . | . | . | . | . | BSA | 92 |
| . | . | . | . | . | . | . | . | BSB | 108 |
| . | . | . | . | . | . | . | . | PPK | 19 |
| . | . | . | . | . | . | . | . | END | 27 |

DATA TAPE

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | CODE | |
|---|---|---|---|---|---|---|---|------|-----|
| . | . | . | . | . | . | . | . | BBS | 128 |
| . | . | . | . | . | . | . | . | BTS | 129 |
| . | . | . | . | . | . | . | . | END | 27 |
| . | . | . | . | . | . | . | . | EPC | 0 |

9.2 GLOSSARY

9.2.0 Introduction

The Glossary of Terms is intended to aid the technician in understanding the different terms and abbreviations.

9.2.1 Glossary of Terms

AB Activating Bar.
 ADD Addition.
 B Register B (2nd Operand).
 BOC Begin of Column (Symbol).
 BOF Begin of Field (Symbol).
 BOP Begin of Program (Symbol).
 Borrow To take from the next higher denomination in the minuend.
 BOS Begin of Sub-Program (Symbol).
 BSP Back Space (Printer).
 C Register C (Result).
 C1 Register Read Time.
 C2 Logic Write Time.
 C3 Logic Read Time.
 C4 Register Write Time.
 Carry To transfer to the next higher order.
 CC Output of the Constant Current Flip-Flop.
 CCI Constant Current Inhibit.
 CCR Constant Current Read.
 CCW Constant Current Write.
 CFM Compare for Magnitude (Command).
 CLK Clear Key (Control Keyboard).
 CPY Copy (Command).
 CS Encapsulated NOR Gates or Flip-Flop.
 DC Divide Constant (about 10^{10}).
 DIV Division (Command).
 DKL Data Keyboard Lock (Symbol).
 DKR Data Keyboard Release (Symbol).
 DP Decimal Point.
 EF Emitter Follower.
 END End of Information Symbol (IFP, OFP and CFM).
 f The function of a Command.
 F(1-8) Function Transfer Circuit Terms.
 FF Flip-Flop.
 FKL Function Keyboard Lock (Symbol).
 FKR Function Keyboard Release (Symbol).
 FSP Forward Space (Symbol).
 FW(1-12) Field Width (Symbol).
 G Register G (Control).
 H Register H (Storage).
 HLT Halt (Symbol). Stop Program.
 I Register I (Input).
 ICL Inhibit Core Logic.
 IFC Input from Control Keyboard (Command).
 IFP Input from Program Tape (Command).
 IL Logic Inverter.
 IP Power Inverter.
 i/p Input to a Gate or Flip-Flop.

K(1-8) K Transfer Circuit Terms (Determines the Program Count).
 LC Lower Case (Printer).
 LD Lamp Driver.
 Logic time Access time to the Logic Cores.
 LSP Line Space (Symbol).
 MAB Minus Activating Bar (Control Keyboard).
 MIN Minus Bar (Control Keyboard).
 Minuend The quantity from which another is to be subtracted.
 MSS Motor Stepping Switch.
 n1 Address of the first operand.
 n2 Address of the second operand; or a modification of the n1 Address; or an input/output.
 O Dummy Address.
 OB Output Buffer.
 OC Overflow Constant = 10^{12} .
 OFP Output from Program Tape (Command).
 OFS Output from Storage (Command).
 OL Overlapping.
 o/p Output of a Gate or Flip-Flop.
 OV Overflow.
 P(1-12) Power Gates.
 PC Program Count.
 PHR Printhead Return (Symbol).
 PHS Printhead Shift (Symbol).
 PHT Printhead Tabulate (Symbol).
 PKL Printer Keyboard Lock (Symbol).
 PKR Printer Keyboard Release (Symbol).
 PL(1-4) Logic Power Gate.
 PR(1-4) Register Power Gate.
 PSL Program Shift Left (Symbol).
 PSR Program Shift Right (Symbol).
 P time Bit time (Consists of four C times).
 PTL Pulse Time Limiter.
 R47 Next Program Count Storage.
 R48 Cores feeding the Output Buffer.
 Read time The time during which a Logic or Register Core will be read.
 Register time Access time to the Register Cores.
 RSH Round off and Shift (Command).
 RSP Return to Sub-program (Symbol and Key).
 RTL Resistor Transistor Logic.
 S(1-12) Registers S1 through S12 (Storage).
 SC Subtract Constant = 10^{11} .
 SD Solenoid Driver.
 SDP Shift Decimal Point (Command).
 SHF Shift (Command).
 SK(1-9) Skip 1 through 9 (Symbol).
 SKS Skip Symbol.
 SP(0-15) Sub-program 0 through 15 (Symbol).
 STP Step (Symbol).
 SUB Subtraction (Command).
 Subtrahend A quantity to be subtracted from another.
 T(1-9) Tape Transfer Circuits.
 TAB Tabulate Tape and Printhead (Symbol and Key).