

CORTEX FIG - FORTH USER MANUAL

REL 1.0

CONTENTS

INTRODUCTION
INSTALLATION
SYSTEM MEMORY MAP
GLOSSARY
EXTENSIONS
EXAMPLE FORTH DEFINITIONS
CASSETTE SUPPORT
DISC SUPPORT

APPENDIX

(A) ERROR MESSAGES
(B) SYSTEM VECTORS
(C) BIBLIOGRAPHY

INTRODUCTION

THE FORTH LANGUAGE WAS INVENTED IN 1970 BY CHARLES MOORE. IT IS UNIQUE AMONG HIGH LEVEL LANGUAGES IN THAT THE WHOLE KERNAL INCLUDING COMPILER, EDITOR AND ASSEMBLER ONLY OCCUPIES ABOUT 8K BYTES. THE LANGUAGE PACKAGE SUPPLIED HERE IS FIG-FORTH WITH EXTENSIONS FOR DISC, CASSETTE AND THE CRU FACILITY OF THE POWERTRAN CORTEX COMPUTER.

THIS MANUAL SHOULD NOT BE REGARDED AS A TUTORIAL ON THE FORTH LANGUAGE, IT IS A DESCRIPTION OF THE "WORDS" AND FACILITIES OFFERED IN THE CORTEX FIG-FORTH EPROM SET SUPPLIED BY LOMBARD SYSTEMS. FOR THE USER NEW TO THE FORTH LANGUAGE WE RECOMMEND THE PURCHASE OF THE EXCELLENT BOOK BY LEO BRODIE "STARTING FORTH." SEE APPENDIX C.

FORTH TYPICALLY RUNS FIVE TO TEN TIMES FASTER THAN THE EQUIVALENT BASIC. IT IS INHERENTLY STRUCTURED AND ENCOURAGES TOP DOWN PROGRAM DESIGN. THE USER BUILDS HIS PROGRAM FROM PREVIOUSLY DEFINED "WORDS" USING THESE WORDS TO CREATE OTHER "WORDS" UNTIL THE WHOLE APPLICATION IS PERFORMED BY THE EXECUTION OF ONE "WORD."

INSTALLATION

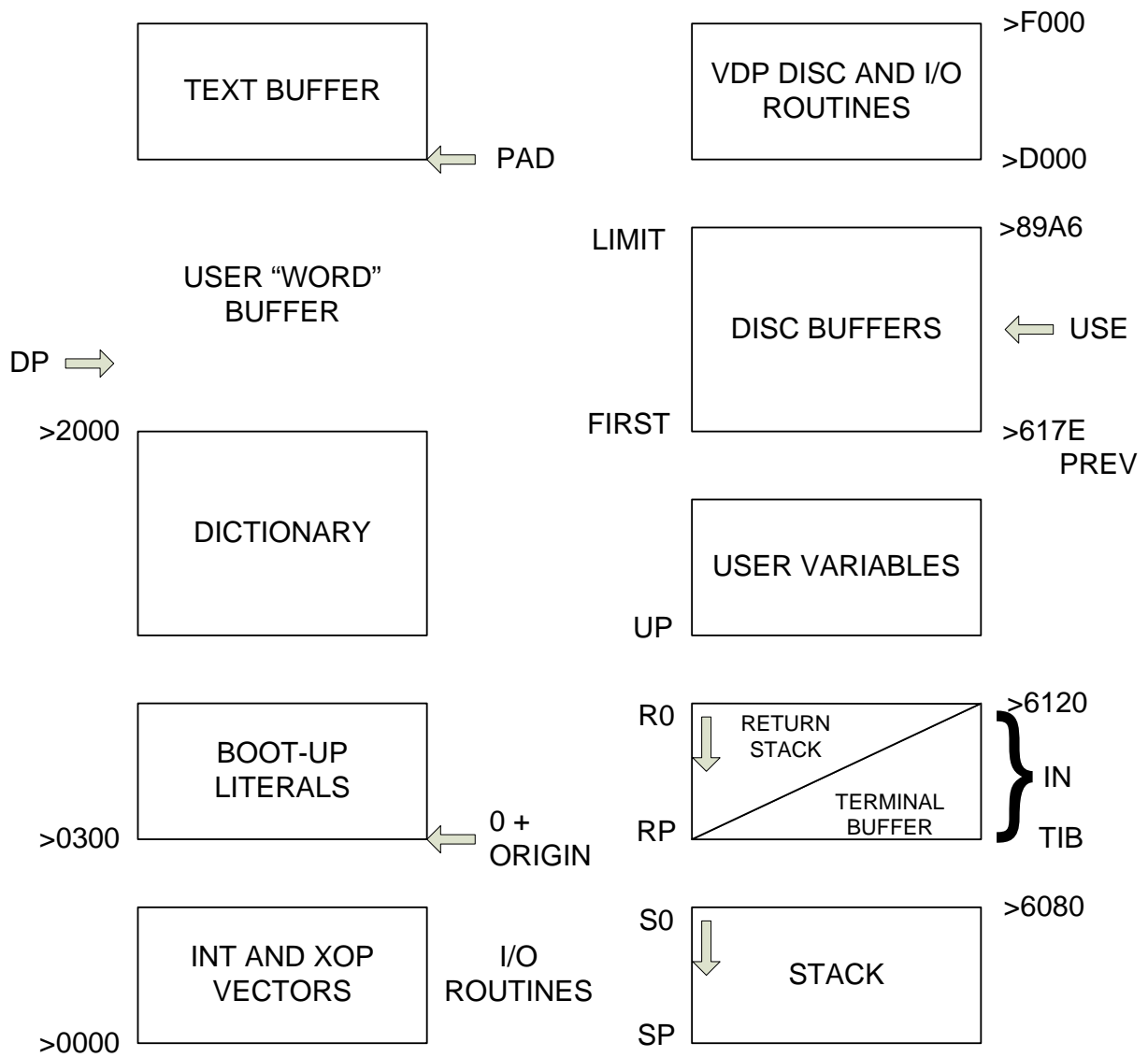
CORTEX FIG-FORTH IS SUPPLIED IN TWO 2564 EPROMS. THESE REPLACE THE FIRST TWO BASIC EPROMS.

CAREFULLY REMOVE THE FORTH EPROMS FROM THEIR PACKING. IT IS A GOOD IDEA TO TOUCH AN EARTHED OBJECT BEFORE HANDLING THE EPROMS AS THEY CAN BE DAMAGED BY STATIC CHARGE.

REMOVE THE COVER FROM YOUR CORTEX AND LOCATE THE BASIC EPROMS (SOCKETS IC45, IC46, AND IC47 IN THE ELECTRONICS TODAY INTERNATIONAL ARTICLE). REMOVE THE FIRST TWO EPROMS IC47 AND IC46. PLUG THE EPROM MARKED >0000 INTO SOCKET IC47, AND THE >2000 EPROM INTO SOCKET IC46. CORTEX-FORTH OCCUPIES MEMORY IN PROM FROM >0000 TO >3FFF (THE ">" SIGN MEANS HEX). THE THIRD BASIC EPROM MAY BE LEFT IN OR REMOVED AS YOU DESIRE, EITHER WAY WILL MAKE NO DIFFERENCE TO THE EXECUTION OF THE FORTH SYSTEM.

REPLACE THE COVERS AND POWER ON THE CORTEX. THE SYSTEM SHOULD REPLY WITH THE SIGN ON BANNER CORTEX-FIG FORTH REL 1.0XX, WHERE XX ARE THE RELEASE LETTERS. IF THIS FAILS TO OCCUR, GO BACK AND RE-CHECK YOUR WORK TO MAKE SURE NONE OF THE PINS OF THE EPROMS ARE BENT OVER OR OUTSIDE THE SOCKETS. HAPPY FORTH COMPUTING!

CORTEX FIG - FORTH MEMORY MAP



CORTEX FIG - FORTH GLOSSARY

THIS SECTION CONTAINS ALL OF THE STANDARD DEFINITIONS IN CORTEX FORTH. THE FIRST LINE OF EACH ENTRY SHOWS A SYMBOLIC REPRESENTATION OF THE PROCEDURE'S ACTION ON THE PARAMETER STACK. THE SYMBOLS ARE IN ORDER OF THEIR POSITION ON THE PARAMETER STACK, THE THREE DASHES REPRESENT THE PROCEDURE'S EXECUTION POINT; ANY PARAMETERS LEFT ON THE STACK AFTER EXECUTION ARE LISTED TO THE RIGHT OF THE THREE DASHES. THE TOP OF THE STACK IS RIGHT MOST IN ORDER. THIS HAS BECOME STANDARD NOTATION FOR FORTH "WORD" ACTIONS.

THE SYMBOLS ARE AS FOLLOWS:

ADDR	MEMORY ADDRESS
B	8 BIT BYTE (HI 8 BITS ZERO)
C	7 BIT ASCII CHARACTER (HI 9 BITS ZERO)
D	32 BIT SIGNED DOUBLE INTEGER, MOST SIGNIFICANT PORTION WITH SIGN ON TOP OF THE STACK.
F	BOOLEAN FLAG, 0 = FALSE, NON-ZERO = TRUE
FF	BOOLEAN FALSE FLAG = 0
N	16 BIT SIGNED INTEGER NUMBER
U	16 BIT UNSIGNED INTEGER
TF	BOOLEAN TRUE FLAG = NON-ZERO

THE CAPITAL LETTERS ON THE RIGHT SHOW DEFINITION CHARACTERISTICS;

C	MAY ONLY BE USED WITHIN A COLON DEFINITION. A DIGIT INDICATES THE NUMBER OF MEMORY ADDRESSES USED, IF MORE THAN ONE.
E	INTENDED FOR EXECUTION ONLY.
L0	LEVEL ZERO DEFINITION OF FORTH-78
L1	LEVEL ONE DEFINITION OF FORTH-78
P	HAS PRECEDENCE BIT SET, WILL EXECUTE EVEN WHEN COMPILING.
U	USER VARIABLE

UNLESS OTHERWISE NOTED, ALL REFERENCES TO NUMBERS ARE FOR 16 BIT SIGNED INTEGERS. FOR 32 BIT SIGNED DOUBLE NUMBERS, THE MOST SIGNIFICANT PART (WITH THE SIGN) IS ON TOP. ALL ARITHMETIC IS IMPLICITLY 16 BIT SIGNED INTEGER MATH, WITH ERROR AND UNDER-FLOW INDICATION UNSPECIFIED.

(ABORT)

EXECUTES AFTER AN ERROR WHEN WARNING IS -1. THIS WORD NORMALLY EXECUTES ABORT, BUT MAY BE ALTERED (WITH CARE) TO A USER'S ALTERNATIVE PROCEDURE.

(DO)

C

THE RUN TIME PROCEDURE COMPILED BY 'DO' WHICH MOVES THE LOOP CONTROL PARAMETERS TO THE RETURN STACK. SEE 'DO'.

(FIND)

ADDR1 ADDR2 --- PFA B TF (OK)
 ADDR1 ADDR2 --- FF (BAD)

SEARCHES THE DICTIONARY STARTING AT THE NAME FIELD ADDRESS ADDR2, MATCHING TO THE TEXT AT ADDR1. RETURNS PARAMETER FIELD ADDRESS. LENGTH BYTE OF NAME FIELD AND BOOLEAN TRUE FOR A GOOD MATCH. IF NO MATCH IS FOUND, ONLY A BOOLEAN FALSE FLAG IS LEFT.

(LINE)

N1 N2 --- ADDR COUNT

CONVERT THE LINE NUMBER N1 AND THE SCREEN N2 TO THE DISC BUFFER ADDRESS CONTAINING THE DATA. A COUNT OF 64 INDICATES THE FULL LINE TEXT LENGTH.

(LOOP)

C2

THE RUN TIME PROCEDURE COMPILED BY 'LOOP' WHICH INCREMENTS THE LOOP INDEX AND TESTS FOR LOOP COMPLETION. SEE 'LOOP'.

(NUMBER)

D1 ADDR1 --- D2 ADDR2

CONVERT THE ASCII TEXT BEGINNING AT ADDR1+1 WITH REGARD TO BASE. THE NEW VALUE IS ACCUMULATED INTO DOUBLE NUMBER D1, BEING LEFT AS D2. ADDR2 IS THE ADDRESS OF THE FIRST UNCONVERTABLE DIGIT. USED BY NUMBER.

*

N1 N2 --- PROD

L0

LEAVE THE SIGNED PRODUCT OF TWO SIGNED NUMBERS

*/

N1 N2 N3 --- N4

L0

LEAVE THE RATIO $N4 = N1 * N2 / N3$ WHERE ALL ARE SIGNED NUMBERS. RETENTION OF AN INTERMEDIATE 31 BIT PRODUCT PERMITS GREATER ACCURACY THAN WOULD BE AVAILABLE WITH THE SEQUENCE $N1 N2 * N3 /$

*/MOD

N1 N2 N3 --- N4 N5

L0

LEAVE THE QUOTIENT N5 AND REMAINDER N4 OF THE OPERATION $N1 * N2 / N3$ A 31 BIT INTERMEDIATE PRODUCT IS USED AS FOR */.

+

N1 N2 --- SUM

L0

LEAVE THE SUM OF $N1 + N2$.

+!

N1 ADDR ---

L0

ADD N TO THE VALUE AT THE ADDRESS. PRONOUNCED "PLUS-STORE".

+-

N1 N2 --- N3

APPLY THE SIGN OF N2 TO N1, WHICH IS LEFT AS N3.

- . N --- L0**
 PRINT A NUMBER FROM A SIGNED 16 BIT TWO'S COMPLEMENT VALUE, CONVERTED ACCORDING TO THE NUMERIC BASE. A TRAILING BLANK FOLLOWS . PRONOUNCED "DOT".
- ." L0**
 USED IN THE FORM: ." CCCC"
 COMPILES AN IN-LINE STRING CCCC (DELIMITED BY THE TRAILING "), WITH AN EXECUTION PROCEDURE TO TRANSMIT THE TEXT TO THE SELECTED OUTPUT DEVICE. IF EXECUTED OUTSIDE A DEFINITION, '."' WILL IMMEDIATELY PRINT THE TEXT UNTIL THE FINAL ' "'.
- .LINE LINE SCR ---**
 PRINT ON THE TERMINAL DEVICE A LINE OF TEXT FROM THE DISC BY ITS LINE AND SCREEN NUMBER. TRAILING BLANKS ARE SUPPRESSED.
- .R N1 N2 ---**
 PRINT THE NUMBER N1 RIGHT ALIGNED IN A FIELD WHOSE WIDTH IS N2. NO FOLLOWING BLANK IS PRINTED.
- / N1 N2 --- QUOT L0**
 LEAVE THE SIGNED QUOTIENT OF N1/N2.
- /MOD N1 N2 --- REM QUOT L0**
 LEAVE THE REMAINDER AND SIGNED QUOTIENT OF N1/N2. THE REMAINDER HAS THE SIGN OF THE DIVIDEND.
- 0 1 2 3 --- N**
 THESE SMALL NUMBERS ARE USED SO OFTEN THAT IT IS ATTRACTIVE TO DEFINE THEM BY NAME IN THE DICTIONARY AS CONSTANTS.
- 0< N --- F L0**
 LEAVE A TRUE FLAG IF THE NUMBER IS LESS THAN ZERO (NEGATIVE), OTHERWISE LEAVE A FALSE FLAG.
- 0= N --- F L0**
 LEAVE A TRUE FLAG IF THE NUMBER IS EQUAL TO ZERO, OTHERWISE LEAVE A FALSE FLAG.
- OBRANCH F --- C2**
 THE RUN-TIME PROCEDURE TO CONDITIONALLY BRANCH. IF F IS FALSE (ZERO), THE FOLLOWING IN-LINE PARAMETER IS ADDED TO THE INTERPRETIVE POINTER TO BRANCH AHEAD OR BACK. COMPILED BY IF, UNTIL, AND WHILE.
- 1+ N1 --- N2 L1**
 INCREMENT N1 BY 1
- 2+ N1 --- N2**
 LEAVE N1 INCREMENTED BY 2.

- : P, E, L0
 USED IN THE FORM CALLED A COLON DEFINITION:
 : CCCC ... ;
 CREATES A DICTIONARY ENTRY DEFINING CCCC AS EQUIVALENT TO THE FOLLOWING SEQUENCE OF FORTH WORD DEFINITIONS '...' UNTIL THE NEXT ';' OR ';' CODE'. THE COMPILING PROCESS IS DONE BY THE TEXT INTERPRETER AS LONG AS STATE IS NON-ZERO. OTHER DETAILS ARE THAT THE CONTEXT VOCABULARY IS SET TO THE CURRENT VOCABULARY AND THAT WORDS WITH THE PRECEDENCE BIT SET (P) ARE EXECUTED RATHER THAN BEING COMPILED.
- ; P, C, L0
 TERMINATE A COLON-DEFINITION AND STOP FURTHER COMPILATION. COMPILES THE RUN-TIME ';' S'.
- ; CODE P, C, L0
 USED IN THE FORM:
 : CCCC ... ;CODE
 ASSEMBLY MNEMONICS
 STOP COMPILATION AND TERMINATE A NEW REFINING WORD CCCC BY COMPILING (;CODE). SET THE CONTEXT VOCABULARY TO ASSEMBLER, ASSEMBLING TO MACHINE CODE THE FOLLOWING MNEMONICS. WHEN CCCC LATER EXECUTES IN THE FORM:
 CCCC NNNN
 THE WORD NNNN WILL BE CREATED WITH ITS EXECUTION PROCEDURE GIVEN BY THE MACHINE CODE FOLLOWING CCCC. THAT IS, WHEN NNNN IS EXECUTED, IT DOES SO BY JUMPING TO THE CODE AFTER NNNN, AN EXISTING DEFINING WORD MUST EXIST IN CCCC PRIOR TO ';' CODE'.
- ; S P, L0
 STOP INTERPRETATION OF A SCREEN. ';' S' IS ALSO THE RUN-TIME WORD COMPILED AT THE END OF A COLON-DEFINITION WHICH RETURNS EXECUTION TO THE CALLING PROCEDURE.
- < N1 N2 --- F L0
 LEAVE A TRUE FLAG IF N1 IS LESS THAN N2; OTHERWISE LEAVE A FALSE FLAG.
- <# L0
 SETUP FOR PICTURED NUMERIC OUTPUT FORMATING USING THE WORDS:
 <# # #S SIGN #>
 THE CONVERSION IS DONE ON A DOUBLE NUMBER PRODUCING TEXT AT PAD.
- <BUILDS C, L0
 USED WITHIN A COLON-DEFINITION:
 : CCCC <BUILDS ...
 DOES> ... ;
 EACH TIME CCCC IS EXECUTED, <BUILDS DEFINES A NEW WORD WITH A HIGH LEVEL EXECUTION PROCEDURE. EXECUTING CCCC IN THE FORM:
 CCCC NNNN
 USES '<BUILDS' TO CREATE A DICTIONARY ENTRY FOR NNNN WITH A CALL TO THE 'DOES>' PART FOR NNNN. WHEN NNNN IS LATER EXECUTED, IT HAS THE ADDRESS OF

@	ADDR --- N	L0
	LEAVE THE 16 BIT CONTENTS OF ADDRESS.	
ABORT		L0
	CLEAR THE STACKS AND ENTER THE EXECUTION STATE. RETURN CONTROL TO THE OPERATORS TERMINAL, PRINTING SIGN ON MESSAGE.	
ABS	N --- L1	L0
	LEAVE THE ABSOLUTE VALUE OF N AS U.	
AGAIN	ADDR N --- (COMPILING)	P, C2, L0
	USED IN A COLON-DEFINITION IN THE FORM: BEGIN ... AGAIN AT RUN-TIME, AGAIN FORCES EXECUTION TO RETURN TO CORRESPONDING BEGIN. THERE IS NO EFFECT ON THE STACK. EXECUTION CANNOT LEAVE THIS LOOP (UNLESS R> DROP IS EXECUTED ONE LEVEL BELOW). AT COMPILE TIME, AGAIN COMPILES BRANCH WITH AN OFFSET FROM HERE TO ADDR. N IS USED FOR COMPILE TIME ERROR CHECKING.	
ALLOT	N ---	L0
	ADD THE SIGNED NUMBER TO THE DICTIONARY POINTER DP. MAY BE USED TO RESERVE DICTIONARY SPACE OR RE-ORIGIN MEMORY. N IS A BYTE VALUE.	
AND	N1 N2 --- N3	L0
	LEAVE THE BITWISE LOGICAL AND OF N1 AND N2 AS N3.	
B/BUF	--- N	
	THIS CONSTANT LEAVES THE NUMBER OF BYTES PER DISC BUFFER, THE BYTE COUNT READ FROM DISC BY BLOCK.	
B/SCR	--- N	
	THIS CONSTANT LEAVES THE NUMBER OF BLOCKS PER EDITING SCREEN. BY CONVENTION AN EDITING SCREEN IS ORGANIZED AS 1024 BYTES ORGANIZED AS 16 LINES OF 64 CHARACTERS EACH.	
BACK	ADDR ---	
	CALCULATE THE BACKWARD BRANCH OFFSET FROM HERE TO ADDR AND COMPILE INTO THE NEXT AVAILABLE DICTIONARY MEMORY ADDRESS.	
BASE	--- ADDR	U, L0
	A USER VARIABLE CONTAINING THE CURRENT NUMBER BASE USED FOR INPUT AND OUTPUT CONVERSION.	
BEGIN	--- ADDR N (COMPILING)	P,L0
	OCCURS IN A COLON-DEFINITION IN THE FORM: BEGIN ... UNTIL BEGIN ... AGAIN BEGIN ... WHILE ... REPEAT	

FORTH

P, LO

THE NAME OF THE PRIMARY VOCABULARY. EXECUTION MAKES FORTH THE CONTEXT VOCABULARY. UNTIL ADDITIONAL USER VOCABULARIES ARE DEFINED, NEW USER DEFINITIONS BECOME A PART OF FORTH. FORTH IS IMMEDIATE, SO IT WILL EXECUTE DURING THE CREATION OF A COLON-DEFINITION, TO SELECT THIS VOCABULARY AT COMPILE TIME.

HERE

--- ADDR

LO

LEAVE THE ADDRESS OF THE NEXT AVAILABLE DICTIONARY LOCATION.

HEX

LO

SET THE NUMERIC CONVERSION BASE TO SIXTEEN (HEXADECIMAL).

HLD

--- ADDR

LO

A USER VARIABLE THAT HOLDS THE ADDRESS OF THE LATEST CHARACTER OF TEXT DURING NUMERIC OUTPUT CONVERSION.

HOLD

C ---

LO

USED BETWEEN '<#' AND '#>' TO INSERT AN ASCII CHARACTER INTO A PICTURED NUMERIC OUTPUT STRING, E. G., 2E HOLD WILL PLACE A DECIMAL POINT.

I

--- N

C, LO

USED WITHIN A DO-LOOP TO COPY THE LOOP INDEX TO THE STACK; SEE R.

ID.

ADDR ---

PRINT A DEFINITION'S NAME FROM ITS NAME FIELD ADDRESS.

IF

F --- (RUN-TIME)

--- ADDR N (COM

P, C2, LO

OCCURS IN A COLON-DEFINITION IN THE FORM:

IF (TP) ... ENDIF

IF (TP) ... ELSE (FP) ... ENDIF

AT RUN-TIME, IF SELECTS EXECUTION BASED ON A BOOLEAN FLAG. IF F IS TRUE (NON-ZERO), EXECUTION CONTINUES AHEAD THRU THE TRUE PART. IF F IS FALSE (ZERO), EXECUTION SKIPS TO JUST AFTER ELSE TO EXECUTE THE FALSE PART. AFTER EITHER PART, EXECUTION RESUMES AFTER ENDIF. ELSE AND ITS FALSE PART ARE OPTIONAL; IF MISSING FALSE EXECUTION SKIPS TO JUST AFTER ENDIF. AT COMPILE-TIME IF COMPILES OBRANCH AND RESERVES SPACE FOR AN OFFSET AT ADDR. ADDR AND N ARE USED LATER FOR RESOLUTION OF THE OFFSET AND ERROR TESTING.

IMMEDIATE

MARK THE MOST RECENTLY MADE DEFINITION SO THAT WHEN ENCOUNTERED AT COMPILE TIME, IT WILL BE EXECUTED RATHER THAN BEING COMPILED. I. E., THE PRECEDENCE BIT IN ITS HEADER IS SET. THIS METHOD ALLOWS DEFINITIONS TO HANDLE UNUSUAL COMPILING SITUATIONS, RATHER THAN BUILD THEM IN TO THE FUNDAMENTAL COMPILER. THE USER MAY FORCE COMPILATION OF AN IMMEDIATE DEFINITION BY PRECEEDING IT WITH [COMPILE].

IN --- ADDR LO

A USER VARIABLE CONTAINING THE BYTE OFFSET WITHIN THE CURRENT INPUT TEXT BUFFER (TERMINAL OR DISC) FROM WHICH THE NEXT TEXT WILL BE ACCEPTED. WORD USES AND MOVES THE VALUE OF IN.

INDEX FROM TO ---

PRINT THE FIRST LINE OF EACH SCREEN OVER THE RANGE FROM, TO. THIS IS USED TO VIEW THE COMMENT LINES OF AN AREA OF TEXT ON DISC SCREENS.

INTERPRET

THE OUTER TEXT INTERPRETER WHICH SEQUENTIALLY EXECUTES OR COMPILES TEXT FROM THE INPUT STREAM (TERMINAL OR DISC) DEPENDING ON STATE. IF THE WORD NAME CANNOT BE FOUND AFTER A SEARCH OF CONTEXT AND THEN CURRENT IT IS CONVERTED TO A NUMBER ACCORDING TO THE CURRENT BASE. THAT ALSO FAILING, AN ERROR MESSAGE ECHOING THE NAME WITH A "?" WILL BE GIVEN. TEXT INPUT WILL BE TAKEN ACCORDING TO THE CONVENTION FOR WORD. IF A DECIMAL POINT IS FOUND AS PART OF A NUMBER, A DOUBLE NUMBER VALUE WILL BE LEFT. THE DECIMAL POINT HAS NO OTHER PURPOSE THAN TO FORCE THIS ACTION. SEE NUMBER.

KEY --- C LO

LEAVE THE ASCII VALUE OF THE NEXT TERMINAL KEY STRUCK.

LATEST --- ADDR

LEAVE THE NAME FIELD ADDRESS OF THE TOP MOST WORD IN THE CURRENT VOCABULARY.

LEAVE C, LO

FORCE TERMINATION OF A DO-LOOP AT THE NEXT OPPORTUNITY BY SETTING THE LOOP LIMIT EQUAL TO THE CURRENT VALUE OF THE INDEX. THE INDEX ITSELF REMAINS UNCHANGED, AND EXECUTION PROCEEDS NORMALLY UNTIL LOOP OR +LOOP IS ENCOUNTERED.

LFA PFA --- LFA

CONVERT THE PARAMETER FIELD ADDRESS OF A DICTIONARY DEFINITION TO ITS LINK FIELD ADDRESS.

LIMIT --- N

A CONSTANT LEAVING THE ADDRESS JUST ABOVE THE HIGHEST MEMORY AVAILABLE FOR A DISC BUFFER. USUALLY THIS IS THE HIGHEST SYSTEM MEMORY.

- TASK**
 A NO-OPERATION WORD WHICH CAN MARK THE BOUNDARY BETWEEN APPLICATIONS. BY FORGETTING TASK AND RE-COMPILING, AN APPLICATION CAN BE DISCARDED IN ITS ENTIRETY.
- THEN** P, C0, L0
 AN ALIAS FOR ENDIF.
- TIB** --- ADDR U
 A USER VARIABLE CONTAINING THE ADDRESS OF THE TERMINAL INPUT BUFFER.
- TOGGLE** ADDR B ---
 COMPLEMENT THE CONTENTS OF ADDR BY THE BIT PATTERN B.
- TRAVERSE** ADDR1 N --- ADDR2
 MOVE ACCROSS THE NAME FIELD OF A CORTEX FIG-FORTH VARIABLE LENGTH NAME FIELD. ADDR1 IS THE ADDRESS OF EITHER THE LENGTH BYTE OR THE LAST LETTER. IF N=1, THE MOTION IS TOWARD HI MEMORY, IF N=-1, THE MOTION IS TOWARD LOW MEMORY. THE ADDR2 RESULTING IS THE ADDRESS OF THE OTHER END OF THE NAME.
- TRIAD** SCR ---
 DISPLAY ON THE SELECTED OUTPUT DEVICE THE THREE SCREENS WHICH INCLUDE THAT NUMBERED SCR, BEGINING WITH A SCREEN EVENLY DIVISIBLE BY THREE. OUTPUT IS SUITABLE FOR SOURCE TEXT RECORDS, AND INCLUDES A REFERENCE LINE AT THE BOTTOM TAKEN FROM LINE 15 OF SCREEN 4.
- TYPE** ADDR COUNT --- L0
 TRANSMIT COUNT CHARACTERS FROM ADDR TO THE SELECTED OUTPUT DEVICE.
- U*** U1 U2 --- UD
 LEAVE THE UNSIGNED DOUBLE NUMBER PRODUCT OF TWO UNSIGNED NUMBERS.
- U/** UD U1 --- U2 U3
 LEAVE THE UNSIGNED REMAINDER U2 AND UNSIGNED QUOTIENT U3 FROM THE UNSIGNED DOUBLE DIVIDEND UD AND UNSIGNED DIVISOR U1.
- UNTIL** F --- (RUN-TIME)
ADDR N --- (COMPILE) P,C2,L0
 OCCURS WITHIN A COLON-DEFINITION IN THE FORM:
BEGIN . . . UNTIL
 AT RUN-TIME, UNTIL CONTROLS THE CONDITIONAL BRANCH BACK TO THE CORRESPONDING BEGIN. IF F IS FALSE, EXECUTION RETURNS TO JUST AFTER BEGIN; IF TRUE, EXECUTION CONTINUES AHEAD. AT COMPILE-TIME, UNTIL COMPILES (OBRANCH) AND AN OFFSET FROM HERE TO ADDR. N IS USED FOR ERROR TESTS.

- UPDATE LO
MARKS THE MOST RECENTLY REFERENCED BLOCK (POINTED TO BY PREV) AS ALTERED. THE BLOCK WILL SUBSEQUENTLY BE TRANSFERRED AUTOMATICALLY TO DISC SHOULD ITS BUFFER BE REQUIRED FOR STORAGE OF A DIFFERENT BLOCK.
- USE --- ADDR
A VARIABLE CONTAINING THE ADDRESS OF THE BLOCK BUFFER TO USE NEXT, AS THE LEAST RECENTLY WRITTEN.
- USER N --- LO
A DEFINING WORD USED IN THE FORM:
N USER CCCC
WHICH CREATES A USER VARIABLE CCCC. THE PARAMETER FIELD OF CCCC CONTAINS N AS A FIXED OFFSET RELATIVE TO THE USER-POINTER REGISTER UP FOR THIS USER VARIABLE. WHEN CCCC IS LATER EXECUTED, IT PLACES THE SUM OF ITS OFFSET AND THE USER AREA BASE ADDRESS ON THE STACK AS THE STORAGE ADDRESS OF THAT PARTICULAR VARIABLE.
- VARIABLE E, LO
A DEFINING WORD USED IN THE FORM:
N VARIABLE CCCC
WHEN VARIABLE IS EXECUTED, IT CREATES THE DEFINITION CCCC WITH ITS PARAMETER FIELD INITIALIZED TO N. WHEN CCCC IS LATER EXECUTED, THE ADDRESS OF ITS PARAMETER FIELD (CONTAINING N) IS LEFT ON THE STACK, SO THAT A FETCH OR STORE MAY ACCESS THIS LOCATION.
- VOC-LINK --- ADDR U
A USER VARIABLE CONTAINING THE ADDRESS OF A FIELD IN THE DEFINITION OF THE MOST RECENTLY CREATED VOCABULARY. ALL VOCABULARY NAMES ARE LINKED BY THESE FIELDS TO ALLOW CONTROL FOR FORGET ING THRU MULTIPLE VOCABULARYS.
- VOCABULARY E, L
A DEFINING WORD USED IN THE FORM:
VOCABULARY CCCC
TO CREATE A VOCABULARY DEFINITION CCCC. SUBSEQUENT USE OF CCCC WILL MAKE IT THE CONTEXT VACABULARY WHICH IS SEARCHED FIRST BY INTERPRET. THE SEQUENCE "CCCC DEFINITIONS" WILL ALSO MAKE CCCC THE CURRENT VOCABULARY INTO WHICH NEW DEFINITIONS ARE PLACED. IN CORTEX FIG-FORTH, CCCC WILL BE SO CHAINED AS TO INCLUDE ALL DEFINITIONS OF THE VOCABULARY IN WHICH CCCC IS ITSELF DEFINED. ALL VOCABULARYS ULTIMATELY CHAIN TO FORTH. BY CONVENTION VOCABULARY NAMES ARE TO BE DECLARED IMMEDIATE. SEE VOC-LINK.
- VLIST
LIST THE NAMES OF THE DEFINITIONS IN THE CONTEXT VOCABULARY.

]

L1

RESUME COMPILATION, TO THE COMPLETION OF A COLON-DEFINITION. SEE '['.

BYTE IN --- B
 INPUT A 8 BIT BYTE USING THE CURRENT CRUBASE.
 (INPUT IS RIGHT JUSTIFIED ON THE TOS, HIGH 8 BITS ZERO; 8 BIT PORT READ)

THE FOLLOWING WORDS SUPPORT ADDITIONAL FEATURES OF CORTEX FIG-FORTH.
THESE WILL BE EXTENDED WITH FUTURE RELEASES OF CORTEX FORTH.

LU. U---
 PRINT THE UNSIGNED 16 BIT INTEGER ON THE CONSOLE DEVICE.

MEMP ADDR --- ADDR+8
 OUTPUT THE ADDRESS FOLLOWED BY THE CONTENTS OF THE ADDRESS AND THE NEXT 3
 MEMORY LOCATIONS. USED BY DUMP.

DUMP ADDR N ---
 OUTPUT THE CONTENTS OF THE NEXT N ADDRESSES TO THE CONSOLE DEVICE. OUTPUT
 IS FORMATTED 4 WORDS PER LINE.

SIZE ---
 OUTPUTS THE CURRENT DICTIONARY AND BLOCK BUFFER SIZE TO THE CONSOLE
 DEVICE.

CLS
 CLEAR THE CONSOLE SCREEN.

C/L
 CONSTANT >40 CHARACTERS PER LINE USED BY THE FORTH EDITOR.

COLOUR B ---
 SET THE VDP FOREGROUND/BACKGROUND COLOUR. THE LEFT HAND NIBBLE IS THE
 FOREGROUND COLOUR; THE RIGHT HAND, THE BACKGROUND COLOUR.

(8INIT)
 INITIALIZE DRIVE ZERO TO 8INCH, ONE DRIVE, STEP TIME 3MSEC, SETTLLING TIME
 15MSEC, HEAD LOAD TIME 35MSEC. RECAL TO TRACK ZERO.

\$INIT
 USED BY (8INIT)

SOME SIMPLE FORTH DEFINITIONS

SQUARE A NUMBER

THE FOLLOWING ROUTINE WILL SQUARE A NUMBER ON THE TOP OF THE STACK.

```
: SQR DUP * ; (N --- NSQUARED)
```

TYPE "." AND FORTH WILL PRINT THE TOP OF STACK VALUE.

STRING OUTPUT

OUTSIDE A DEFINITION: . " HELLO" WILL ECHO THE STRING IMMEDIATELY A CARRIAGE RETURN IS TYPED.

INSIDE A DEFINITION:

```
: GREETING . " HELLO I'M A CORTEX COMPUTER" ;
```

TYPE GREETING AND SEE THE DIFFERENCE.

THE DO LOOP

TYPE IN THE FOLLOWING DEFINITION:

```
: COUNTER 0 DO 1 . LOOP ;
```

NOW TYPE 8 COUNTER, FORTH WILL THEN TYPE 0 1 2 3 4 5 6 7 OK: ON THE SCREEN. TRY OTHER VALUES.

THIS IS AN EXAMPLE OF A DEFINITE LOOP, THE PROCEDURE INSIDE THE DO . . . LOOP WILL EXECUTE N-1 TIMES, WHERE N IS THE STACK VALUE.

MEMP USES A DO LOOP TO OUTPUT THE CONTENTS OF THE FOUR ADDRESS VALUES.

FLASH THE TIME LED

THIS ROUTINE WILL FLASH THE TIME LED SIX TIMES AT APPROXIMATELY 3 SEC INTERVALS, THEN TYPE A SIGN OFF MESSAGE.

```
: WAIT-3SEC 3 0 DO $WA LOOP ;
```

```
: LAMP-ON 0 RB ;
```

```
: LAMP-OFF 0 SB ;
```

```
: FLASH 6 0 DO WAIT-3SEC LAMP-ON WAIT-3SEC LAMP-OFF LOOP
```

```
CR ." LIGHTS OUT FOLKS" ;
```

SUBSTITUTE 6 RB AND 6 SB IN THE LAMP-ON AND LAMP-OFF DEFINITIONS.

ARRAY GENERATION

THE FOLLOWING WORD IS A DEFINING WORD TO CREATE TWO DIMENSION BYTE ARRAYS.

SUBSCRIPTS START AT ONE.

```
: ARRAY (2D BYTE ARRAY DEFINING WORD )
```

```
<BUILDS DUP C, * ALLOT HERE =CELLS DP!
```

```
DOES> ROT 1 - OVER C@ * + + ;
```

TO CREATE A 2D 4 BY 4 ARRAY CALLED ANIMALS, TYPE 4 4 ARRAY ANIMALS

TO OBTAIN THE VALUE OF AN ELEMENT, TYPE N N ANIMALS C@. THIS WILL PLACE THE BYTE VALUE ON THE STACK.

CASSETTE SUPPORT

THE FOLLOWING WORDS ALLOW THE USER TO SAVE THE USER "WORD" BUFFER TO CASSETTE TAPE.

DSAVE

SAVE THE CURRENT USER DICTIONARY TO CASSETTE TAPE. DSAVE WILL PROMPT FOR A NAME. THIS CAN BE FROM ONE TO TWELVE CHARACTERS IN LENGTH. THE OUTPUT TO TAPE WILL COMMENCE AS SOON AS THE TWELFTH CHARACTER IS INPUT, OR IF THE NAME IS LESS THAN TWELVE CHARACTERS A CARRIAGE RETURN TERMINATES NAME INPUT. IT IS IMPORTANT TO START THE TAPE BEFORE THE TERMINATOR IS ENTERED. DSAVE SAVES FROM \$DS UNTIL HERE.

DREAD

READ A PREVIOUSLY SAVED USER DICTIONARY FROM CASSETTE TAPE DREAD SEARCHES FOR A NAMED FILE ON TAPE. IT WILL READ THE FIRST NAMED FILE FOUND. THE FILE NAME WILL BE OUTPUT AS FOLLOWS:

FOUND 'CCC...' WHERE 'CCC...' IS THE FILE NAME.

INPUT TO FORTH IS ACCEPTED FROM ANY INPUT PORT, CONSEQUENTLY DO NOT HIT ANY KEYS ON THE KEYBOARD OR INPUT FROM THE >80 PORT DURING DREAD. THE FILE LOADED IS CHECKED FOR ERRORS, EITHER OF THE TWO FOLLOWING ERROR CONDITIONS CAN OCCUR DURING TAPE INPUT.

H-ERROR

AN ERROR WAS FOUND IN THE HEADER RECORD, THE DREAD IS ABORTED.

D-ERROR

THE HEADER RECORD WAS RECEIVED CORRECTLY BUT AN ERROR OCCURED IN THE DATA PORTION. IN THIS CASE BAD DATA WILL HAVE BEEN LOADED BUT ONLY AT THE CORRECT ADDRESSES. AGAIN THE DREAD IS ABORTED.

THE FOLLOWING WORDS SUPPORT THE TWO MAIN CASSETTE SAVE AND READ WORDS. THEY MAY BE CALLED DIRECTLY BY THE USER (WITH CAUTION!)

\$CS

SET THE >180 PORT FOR CASSETTE I/O. (2 STOP, NO PARITY, 8 BITS, 300 BAUD)

\$CR

RESET THE >180 PORT FOR NORMAL I/O. (7 BITS- EVEN PARITY, 2 STOP BITS, 300 BAUD)

\$CO

-B1 ---

OUTPUT THE RIGHT HAND BYTE AT TOS TO THE CASSETTE PORT.

\$CI --- B1
INPUT A BYTE FROM THE CASSETTE PORT TO THE RH BYTE OF THE TOS.

\$SS B2 B1--- B1 B2
BYTE REVERSE THE ORDER OF THE TOS (STACK SWAP)

U< U1 U2 --- F
RETURN A TRUE, F=1 FLAG IF U1 < OR EQUAL TO U2. OTHERWISE RETURN FALSE F=0.

\$DS
A VARIABLE CONTAINING THE VALUE EQUAL TO THE START OF THE USER DICTIONARY.

2DROP D ---
REMOVE THE TOP DOUBLE PRECISION VALUE FROM THE STACK.

2DUP D1 --- D1 D1
DUPLICATE THE TOP DOUBLE PRECISION VALUE ON THE STACK.

2SWAP D2 D1 --- D1 D2
EXCHANGE THE TOP TWO DOUBLE PRECISION VALUES ON THE STACK.

\$NA
PROMPT FOR AND GENERATE A NAME HEADER FOR CASSETTE SAVE.

\$PT
A VARIABLE CONTAINING A POINTER TO NAME CHARACTER INPUT.

\$RN
GET A FILE NAME FROM TAPE AND DISPLAY IT ON THE TERMINAL DEVICE.

\$WA
WAIT APPROXIMATLY ONE SECOND.

\$CK
A VARIABLE CONTAINING THE TAPE CHECKSUMS.

WWD B2 B1 ---
OUTPUT THE TWO BYTES AT THE TOS TO THE CASSETTE PORT LOW BYTE FIRST.

\$HD U2 U1 ---
GENERATE A HEADER WITH CHECKSUM FOR THE TWO VALUES ON THE STACK.

\$CD U2 U1 ---
OUTPUT BYTES FROM ADDRESS U1 UPTO AND INCLUDING ADDRESS U2, THEN OUTPUT
A CHECKSUM CELL.

CORTEX FIG - FORTH USER MANUAL

RDW --- B2B1
 INPUT TWO BYTES FROM THE CASSETTE PORT, THE FIRST BYTE IS PLACED LOW ORDER
 ON THE STACK.

\$RH --- U2 U1 F
 READ THE TWO HEADER CELLS FROM THE CASSETTE PORT. (WHERE U2= END ADDRESS,
 U1= START ADDRESS, F=1 IF CHECKSUM IS OK).

\$RD U2 U1 --- F
 READ DATA FROM THE CASSETTE PORT, WHERE U1 EQUALS THE START ADDRESS, U2
 THE END ADDRESS AND F=1 IF THE CHECKSUM IS OK.

\$SP
 SAVE PARAMETERS FOR THIS DICTIONARY TO TAPE.

\$RP
 READ THE PREVIOUS DICTIONARY PARAMETERS FROM TAPE.

DISC SUPPORT

THE CURRENT CORTEX FIG-FORTH REL 1. 0 SUPPORTS A SINGLE 8 INCH DISC DRIVE INSTALLED AS DRIVE ZERO. PLEASE ENQUIRE FOR MULTIPLE DRIVE VARIANTS.

THE DISC FORMAT IS IBM 3740, 77 TRACKS PER SIDE/SINGLE SIDED, 26 SECTORS PER TRACK AND 128 BYTES PER SECTOR. BEFORE USING THE DISC, THE CONTROLLER MUST BE INITIALIZED AND THE DRIVE RE-CALIBRATED. THE (8INIT) WORD PERFORMS THIS FUNCTION. WITH AN 8 INCH DRIVE CONNECTED AND CONFIGURED AS DRIVE ZERO, TYPE (8INIT) FOLLOWED BY RETURN, THE DRIVE SHOULD RE-CALIBRATE AND FORTH WILL REPLY "INITIALIZATION COMPLETE".

DISC OPERATION IS NORMALLY CONTROLLED BY THE FORTH OPERATING SYSTEM VIA THE BLOCK AND BUFFER WORDS. FORTH MANAGES ALL DISC ACCESSES AS A VIRTUAL MEMORY SYSTEM ALLOCATING BLOCK BUFFERS AND WRITING UPDATED BUFFERS BACK TO DISC. SEE THE BUFFER, UPDATE AND BLOCK WORDS IN THE GLOSSARY.

A MORE PRIMITIVE FORM OF DISC ACCESS IS AVAILABLE VIA THE R/W WORD. SEE ALSO THE SYSTEM VECTORS SECTION, APPENDIX B.

AN 8 INCH DISC CONTAINING THE FORTH EDITOR, AN ASSEMBLER AND A NUMBER OF UTILITY PROGRAMS IS AVAILABLE FROM LOMBARD SYSTEMS. PLEASE ENQUIRE FOR DETAILS.

APPENDIX AFORTH ERROR MESSAGES

ERR MSG #	MESSAGE	DESCRIPTION
0	? (PRONOUNCED "HUH?")	THE WORD IN QUESTION CANNOT BE FOUND IN THE DICTIONARY.
1	EMPTY STACK	MORE VALUES HAVE BEEN REMOVED FROM THE PARAMETER STACK THAN WERE ADDED (NOT ENOUGH VALUES ARE ON STACK TO SUPPORT DEFINITION EXECUTION)
2	DICTIONARY FULL	THE DICTIONARY HAS GROWN INTO THE TERMINAL INPUT BUFFER
3	INCORRECT ADX MODE	USED BY THE FORTH ASSEMBLER
4	ISN'T UNIQUE	THE NAME OF THIS DEFINITION ALREADY EXISTS ELSEWHERE IN THE DICTIONARY
6	DISK RANGE?	A DISC ACCESS TO A PHYSICALLY NON-EXISTANT BLOCK NUMBER WAS REQUESTED (CURRENTLY 249 FOR THIS RELEASE. MAKE SURE YOU ARE NOT IN BASE HEX WHEN YOU THINK YOU ARE IN BASE DECIMAL)
7	FULL STACK	TOO MANY VALUES HAVE BEEN ADDED TO THE PARAMETER STACK
8	DISK ERROR	AN I/O ERROR OCCURED WHILE ATTEMPTING TO READ OR WRITE TO VIRTUAL I/O
17	COMPILATION ONLY	THIS WORD MUST ONLY BE USED WITHIN A COLON DEFINITION
18	EXECUTION ONLY	THE WORD MUST NOT BE USED WHILE THE SYSTEM IS IN COMPILE MODE

19	CONDITIONALS NOT PAIRED	THERE IS SOME SORT OF PROBLEM WITH THE PAIRING OF CONDITIONALS WITHIN THE DEFINITION BEING COMPILED (DO WITHOUT LOOP OR BEGIN WITHOUT UNTIL, ETC.)
20	DEFINITION NOT FINISHED	THE POSITION OF THE PARAMETER STACK POINTER DIFFERS FROM WHAT IT WAS WHEN THIS DEFINITION BEGAN COMPILING. SOMETHING IS WRONG WITH THE DEFINITION
21	PROTECTED DICTIONARY	THE ADDRESS OF THE DEFINITION BEING "FORGOTTEN" IS LESS THAN THE VALUE STORED IN FENCE. CHANGE THE VALUE IN FENCE.
22	USE ONLY WHEN LOADING	THIS DEFINITION SHOULD ONLY BE USED WHEN LOADING
23	OFF CURRENT EDIT SCREEN	OCCURS WHEN USING CORTEX FORTH EDITOR
24	DECLARE VOCABULARY	CONTEXT AND CURRENT ARE NOT AIMING AT THE SAME VOCABULARY WHEN ATTEMPTING TO FORGET

APPENDIX BSYSTEM VECTORS

THESE ARE ASSEMBLY LANGUAGE LEVEL CALL VECTORS, IMPROPER USE MAY CRASH THE FORTH SYSTEM.

INPUT-OUTPUT

XOP 0 IS USED FOR LOW LEVEL O/P, CALLED BY EMIT

XOP 1 IS USED FOR LOW LEVEL I/P

ALL OTHER XOP'S ARE AVAILABLE FOR USER USE AT PRESENT, HOWEVER IT IS RECOMMENDED THAT ANY USER XOP'S ARE ALLOCATED FROM 9 UPWARDS. FURTHER RELEASES OF CORTEX FIG-FORTH MAY COMMIT XOP'S LOW ORDER UPWARDS.

NOTE: ALL INPUT IS INTERRUPT DRIVEN (INT 4)

(">" INDICATES A HEXADECIMAL VALUE)

SYSTEM ADDRESSES

CHAR\$	EQU >0080	CONTAINS VALUE OF LAST CHARACTER INPUT.
EVECT	EQU >E284	FORTH DISC READ-WRITE ENTRY VECTOR, EXPECTS STACK SET UP AS PER R/W WORD.
TOG 1	EQU >E28A	SIDE TOGGLE VALUE 0 FOR STANDARD CORTEX. CONTROLS THE SENSE OF THE SIDE SIGNAL SENT TO THE DISC DRIVE. LOAD WITH >0400 TO CHANGE SIDE SIGNAL POLARITY.
MAXB	EQU >E28C	MAXIMUM FORTH BLOCK NUMBER ALLOWED BY SYSTEM.
TMX	EQU >E28E	MAXIMUM TRACK AND MAXIMUM SECTOR VALUES.
PROF	EQU >E290	FORTH WRITE PROTECT FLAG, IF SET TO A 1 WILL WRITE PROTECT TRACK IN MD WORD.
MD	EQU >E292	MAXIMUM TRACK AND PROTECTED TRACK NUMBER. THE LOW ORDER BYTE CONTAINS 2 TIMES THE VALUE OF THE DESIRED PROTECTED TRACK.

SYSTEM CALLS

INIT EQU >E294 CALL BY BLWP @INIT
 INITIALIZE THE DISC CONTROLLER. THIS ROUTINE DOES NOT
 PERFORM A FULL INITIALIZATION OF THE DISC. CONSEQUENTLY
 USER CALLS TO THIS ROUTINE ARE NOT RECOMMENDED.

TXFER EQU >E2F2 CALL BY BLWP @TXFER
 CALLERS WORKSPACE AS FOLLOWS:
 R0= RETURNED STATUS
 R1= NNNN NNNS DDNN NNNN
 TRACK # DRIVE STARTING
 SIDE SECTOR #
 R2= WKNN NNNN NNNN NNNN
 WRITE # OF SECTORS RESERVED
 FLAG TO TRANSFER RH BYTE
 SEEK ONLY
 R3= DATA TRANSFER ADDRESS

IF THE WRITE BIT IS SET, OPERATION IS A SINGLE/MULTI
 SECTOR WRITE. STARTING ADDRESS IN R3. OTHERWISE R3
 CONTAINS START ADDRESS OF WHERE READ INFORMATION
 IS TO BE PLACED. IF THE SEEK BIT IS SET OPERATION ABORTS
 AFTER SEEK.

EXAMPLE:

TO WRITE 8 SECTORS FROM >6000 IN MEMORY TO TRACK >12
 STARTING SECTOR >5 SIDE ZERO, DRIVE 0

CLR R0	CLEAR STATUS
LI R1,>2405	TRACK >12, SIDE 0, DRIVE 0, STARTING SECTOR 5.
LI R2,>8800	WRITE, 8 SECTORS.
LI R3,>6000	TRANSFER ADDRESS >6000.
BLWP @TXFER	EXECUTE COMMAND.
CI R0,>FF00	CHECK STATUS.
JNE ERROR	CALL ERROR IF BAD STATUS.

RCALIB EQU >E3BE CALL BY BLWP @RCALIB
 CALLERS WORKSPACE AS FOLLOWS:
 R0= RETURNED STATUS
 R1= ABAB ABAB NNNN DDDD
 RATES DRIVE
 D3 D2 D1 D0 RESERVED
 THE RATE PAIRS ARE PER DRIVE AS INDICATED, (SEE THE
 TMS 9909 DATA MANUAL) THE LH NIBBLE OF THE RH BYTE
 IS RESERVED. THE DRIVE VALUE IS ONE BIT PER DRIVE
 EXCLUSIVE. E. G., DRIVE 2 = 0100. DRIVE 0 = 0000.

APPENDIX C

BIBLIOGRAPHY

FIG-FORTH INSTALLATION MANUAL

FORTH INTEREST GROUP, PO BOX 1105, SAN CARLOS, CA 94070

THE FIG GLOSSARY SUPPLIED IN THIS MANUAL WAS TAKEN FROM THE ABOVE PUBLIC DOMAIN PUBLICATION.

BAKER & DERICK	FORTH ENCYCLOPEDIA (MVP, 1982)
BRODIE, LEO	STARTING FORTH (PRENTICE-HALL/1981)
HAYDON, G	ALL ABOUT FORTH (MVP, 1983)
HUANG, T	AND SO FORTH (MVP)
KATZAN, H JR.	INVITATION TO FORTH (VAN NOSTRAND)
KNECHT, K	INTRODUCTION TO FORTH (HOWARD SAMS)
LOELIGER, R	THREADED INTERPRETIVE LANGUAGES (MCGRAW HILL, 1981 BYTE BOOKS)
SCANLON, L	FORTH PROGRAMMING (HOWARD SAMS, 1982)
TING, C	SYSTEMS GUIDE TO FIG-FORTH (MVP)
WINFIELD, A	THE COMPLETE FORTH (SIGMA TECHNICAL PRESS, 1983)
MVP	MOUNTAIN VIEW PRESS PO BOX 4656 MOUNTAIN VIEW, CALIFORNIA 94040 U.S.A.