

# **FORTH NOTEBOOK**

**C. H. TING**, PHD

**VOLUME 2**

(c) All Rights Reserved

OFFETE ENTERPRISES, INC.

**1987**

(c) Copyright, 1987 by C. E. Ting

First Edition: November 1987

\*\*\*\*\*  
All rights reserved. This book, or any part  
thereof, may not be reproduced in any form  
without written permission from the Author.  
\*\*\*\*\*

Printed in the United States of America  
by

OFFETE ENTERPRISES, INC.  
1306 South B Street  
San Mateo, CA 94402  
Tel. (415) 574-8250

## PREFACE

It has been four years since the 'Forth Notebook' was first published. The purpose of it was educational as it was a collection of examples and materials I used to teach Forth. All the code was developed on an LSI-11/04 microcomputer with a polyForth kernel resident in PROM's. The entire book was written in blocks with the polyForth line editor. Stripping out the screen numbers and line numbers, the remaining text is fairly readable. Most people were not aware of the block boundary if I had not mention it.

Since then, I had encountered many more computers and peripheral equipment. In dealing with different computers and related hardware, Forth had been a great help. In most circumstances where a group of hardware people and software people were working together, the hardware people generally had an upper hand, and software people were in disadvantage. If any problem occurred and fingers were pointing, hardware engineers could demonstrate that hardware worked and concluded that the fault had to be in software. It was very difficult for the software people to prove that the software worked and threw the ball back to the other court. With Forth, the computer was under my control and I could show that the computer and the software were doing what they were supposed to do. The hardware people had to work much harder in order to turn the table around.

Much had happened in the last four years. FigForth and Forth-79 gave way to Forth-83, and now an ANS Forth standard is in the works. 16 bit microprocessors had replaced 8 bit microprocessors in the personal computers, and now 32 bit microprocessors are replacing the 16 bit ones. Chuck Moore's baby, the Novix NC4000 Forth engine saw the light of day, and Harris is putting this engine in their ASIC library. Parallel processors also marched out of the laboratories and into the market place. Neural network is becoming the brightest star rising in the horizon of artificial intelligence. I had the good fortune to touch many of these new developments.

This new volume of 'Forth Notebook' documents my programming activity in the last four years, reflecting my reactions to the rapidly changing scene in the computer industry. It is a collection of research papers on a wide range of problems I encountered in this period. A goal in publishing this volume is to show that Forth is very adaptive in solving these problems and that the solutions can be documented for other programmers to use. Hardware and technology may change, but this remarkable software methodology remains as sharp as ever.

The materials are arranged in four parts. Part One contains papers dealing with Forth systems hosted on various CPU's. F83 was made ROMmable by modifying the metacompiler and the kernel, which were used to build many dedicated maicrocomputers. The 8086 and 68000 disassemblers show how to recover machine mnemonics from assembled code. Part Two collects the works related to image processing with recently available boards which can be used in a microcomputer for capturing and enhancing video images. Part Three is devoted primarily to parallel processing using NCR GAPP chips, which have very unique property allowing millions of processor units to be link together to solve problems with extremely high throughput. An early paper on a floating point array processor is included here due to the similar orientation towards high speed math operations. Part Four is the fun part, with articles on humor, music, poems, and other miscellaneous subjects, demonstrating my inability to focus my attention on more serious subjects.

An important area of my interests not represented in this volume is that of NC4000, the Forth microprocessor. Most of my work on this processor were published in the 'More on NC4000' series of newsletters; therefore, it is not necessary to collect them into this volume now. May be in a couple of years, I will be able to put them into a separate volume as more experiments and experience will be accumulated.

A special effort is made to publish this volume and have it available for the Nine Annual Forth Convention to be held in San Jose, California, November 13-14, 1987. The theme of this Convention will be the tenth anniversary of the Forth Interest Group. I wish that this volume will be my salute to the Forth Interest Group and all the people involved in its founding and evolution. Happy birthday, FIG!

C. H. Ting

November 1987  
San Mateo, California

## CONTENTS

Part One.	Forth Systems	1
1.	Metacompiling F83 for ROM based systems	3
2.	Forth for the Harris 80 computer	31
3.	VAX as a 32 bit Forth machine	47
4.	8086 Disassembler	61
5.	68000 Disassembler	75
Part Two.	Low Cost Image Processing	87
6.	Datacube QVG-123 image processor	91
7.	Matrox PIP-1024 image processor	107
8.	CAT-1600 imaging system	117
Part Three.	Parallel Processing	127
9.	High density parallel processing: I. The processing array and macro controller	131
10.	High density parallel processing: II. Software and programming	135
11.	Simulator of a GAPP processor array	141
12.	GAPP assembler	155
13.	An interactive operating system for AP120B array processor	175
Part Four.	Miscellaneous Work	195
14.	A polyphonic electronic organ	197
15.	A devil's dictionary	217
16.	Neural networks and expert systems	233
17.	Chinese limericks	245
18.	The simplest line drawing routine	253
19.	Dirty words in Forth	257
20.	Statistics of word usage in F83	261
21.	Zapping the F83 dictionary	267
Index		271

## LISTINGS

1.	Metacompiler for ROM based systems	8
2.	Kernel of ROM target system	12
3.	Utility	28
4.	Editor and assembler for Harris 80	40
5.	Forth Kernel in VAX machine code	51
6.	8086 disassembler	66
7.	68000 disassembler	79
8.	QVG-123 source code	99
9.	Matrox PIP-1024 utility	112
10.	CAT-1600 source code	122
11.	Game of life with GAPP	139
12.	GAPP simulator using IP5500 image processor	147
13.	GAPP assembler on PC	159
14.	Controlling AP120B array processor	192
15.	Demonstration of electronic organ	206
16.	Devil's dictionary	228
17.	Neural networks	240
18.	Chinese limericks	251
19.	Line drawing	255
20.	Statistical analysis of words	266
21.	Zapping unused words	270

## FIGURES

1.	Commands in Harris 80 Forth	39
2.	Ranking of CISC machines	48
3.	F83 kernel words and their VAX counter parts	49
4.	8086 instruction set matrix	65
5.	Disassembled 8086 machine code	66
6.	Index of 68000 disassembler code	76
7.	Examples of disassembled 68000 machine code	77
8.	Control and data registers in QVG-123	95
9.	Control and data registers in AF-123	96
10.	QVG-123 programming procedures	97
11.	Self portrait	98
12.	Datacube QVG-123 brochure	105
13.	PIP-1024 register addresses and functions	108
14.	Intensity transformation curves	111
15.	Matrox PIP-512 brochure	115
16.	GAPP processor array	133
17.	Functional units within the DMC	134
18.	Instruction formats for the distributed macro controller	136
19.	Schematic diagram of a GAPP processor unit	145
20.	Truth table of GAPP ALU	146
21.	Logic operations of GAPP ALU	146
22.	NCR GAP chip brochure	166
23.	Programming model of AP120B array processor	188
24.	A demo session of AP120B Forth	189
25.	Entry points of the AP load module	190
26.	Constants in the MD memory	191
27.	PC interface of the electronic organ	198
28.	8253 circuit for tone generation	199
29.	Active channels and control register	201
30.	Words in the Devil's dictionary	224
31.	Sample dialog	226
32.	A model of neural network	239
33.	A trivial expert system	237
34.	Examples of Chinese poems	248
35.	Chinese limericks generated by POEM	249
36.	Screen display of limericks	250
37.	Pure and dirty words in Forth-83 Standard	260
38.	Forth word layout in F83	261
39.	Statistics of F83 FORTH word usage	264
40.	Most often occurred FORTH words in F83	265
41.	Word distribution in F83/8086	267
42.	Zapped FORTH vocabulary	269