PERSONAL COMPUTER WORLD

SPECIAL SHOW NUMBER

50p U.S.$2.00
OCTOBER 1978

Europe's first magazine for personal computers for home and business use

Pilot flies across the Atlantic

Something bit me

Small business case study



MINOTAUR-New computer game

TRS-80—THE BIGGEST NAME IN LITTLE COMPUTERS

Complete and Ready to Go NOW!



- Amazing all-in system at a low price
- Big 12" video monitor
- Standard typewriter Keyboard

The TRS-80 computer system is 100% wired, and tested for 240 VAC so you can put it to work immediately! It's ideal for finances, education, accounting, laboratory—even games at home. Below are four complete TRS-80 systems incorporating different combinations of RAM (4K and 16K) and ROM (Level-1 and Level-11 BASIC). Choose the one that's right for you. Expansion is easy due to TRS-80's modular design. All TRS-80 systems below include a 12" video monitor, Realistic CTR-41 battery/AC cassette recorder, power supply, 232-page user's manual, and a 2-game cassette.

Level-1 BASIC system

4K ROM, 4K RAM
Cat.No. 26-1001, 26-1201, 14-841

£499

Level-11 BASIC system

12K ROM, 4K RAM
Cat.No. 26-1004, 26-1201, 14-841

£578

Level-1 with 16K RAM

4K ROM, 16K RAM
Cat.No. 26-1003, 26-1201, 14-841

£728

Level-11 with 16K RAM

12K ROM, 16K RAM
Cat.No. 26-1006, 26-1201, 14-841

£807

You can see the TRS-80 at these Tandy Stores and Dealerships

BASINGSTOKE	22 London Street, Basingstoke. Tel: 52795
BIRMINGHAM AREA	Bilston Road, Wednesbury. Tel: (021) 556 6429
	528 The Bridge, Bull Ring Shopping Centre, Birmingham. Tel: (021) 643 3876
	57-58 Dale End, Birmingham. Tel: (021) 236 4744
BOLTON	5 Nelson Square, Bolton. Tel: 386538
BOURNEMOUTH	134 Commercial Road, Bournemouth. Tel: 293606
BRIGHTON	70 London Road, Brighton. Tel: 693446
BRISTOL	5 Badminton Road, Downend, Bristol. Tel: 561917
BROMSGROVE	6-8 St. John Street, Bromsgrove. Tel: 71800
CAMBRIDGE	12 High Street, Cambridge. Tel: 68155
CHESTER	Kwik Save Centre, Sealand Road, Chester. Tel: 375794
COVENTRY	4 Hales Street, Coventry. Tel: 22894
DARLINGTON	15-16 Priestgate, Darlington. Tel: 58676
DERBY	33 Victoria Street, Derby. Tel: 371066
DONCASTER	32-34 Kingsgate, Waterdale Centre, Doncaster. Tel: 21992
GLASGOW	Audio Visual, 340 Argyle Street, Glasgow. Tel: (041) 221 8958
GLOUCESTER	47 Kings Square, Clarence Street, Gloucester. Tel: 31323
LEEDS	72 Merrion Centre, Leeds. Tel: 42520
LIVERPOOL	168 St. John's Centre, Market Way, L'pool. Tel: (051) 708 0161
LONDON AREA	The Colonnades, Porchester Road, Queensway, Bayswater, W.2. Tel: (01) 221 5317
	7 Embassy Court, Welling. Tel: (01) 303 5483
	124-126 The Broadway, Wimbledon, S.W.19. Tel: (01) 542 6389
	6 New Broadway, S.W.5. Tel: (01) 579 1320
	21 Sentinel Square, Hendon, N.W.4. Tel: (01) 202 7331
LUTON	157 Dunstable Road, Luton. Tel: 36159
MANCHESTER	4-8 The Mall Shopping Centre, Hyde. Tel: (061) 368 0268
	13-15 The Mall Shopping Centre, Sale. Tel: (061) 973 0371
	The Arndale Centre, Stretford. Tel: (061) 865 8214
NEWCASTLE-UPON-TYNE	23 Newgate Centre, Newgate, Newcastle. Tel: 21478
NORTHAMPTON	Weston Favel Shopping Centre, Northampton. Tel: 326354
NOTTINGHAM	126-128 Front Street, Arnold, Nottingham. Tel: 202626
SUTTON	206 High Street, Sutton, Surrey. Tel: (01) 643 8687
SWANSEA	Radio Supplies, 80 Gower Road, Sketty, Swansea. Tel: 24140
WITNEY	Witney Audio, 29 Corn Street, Witney. Tel: 2414
WOLVERHAMPTON	1 Market Street, Wolverhampton. Tel: 21148

Or contact Computer Sales Department, Tandy Corporation, Bilston Road, Wednesbury, West Midlands. WS10 7JN. Tel: (021) 556 6101

TANDY

WORLDWIDE ELECTRONICS

PERSONAL COMPUTER WORLD

Europe's first magazine for personal computers for home and business use

UK 50p
US $2.00

Vol 1, No 6
October 1978

CONTENTS

PUBLISHER'S LETTER 8

EDITORIAL NOTE 8

PCW BOOK REVIEW *Michael James* 8

LETTERS 10

TIDBITS
Products ... services ... news 12

THE MICRO IS A MANY SPLENDORED THING *Leslie Solomon*
Aladdin's lamp brought up to date 15

SOMETHING BIT ME *Chris Howland*
A reader's personal experience 17

PILOT FLIES ACROSS THE ATLANTIC *John Coll*
Pilot, finely tuned to handle text 19

SMALL BEGINNINGS *Derick Daines*
Computing in the primary school 22

SUBMARINE CHASE *A.J. Harding*
"Run silent, run deep" 24

MINOTAUR *John D. Lee and Timothy D. Lee*
The fiery breath of the Bull of Minos 28

TOUCHDOWN *N. Rushton*
Landing on a pocket-sized moon 32

EXHIBITION CATALOGUE 33

ASSEMBLY CODE PROGRAMMING FOR THE BEGINNER *Stephen Collins* 41

BINARY FINERY *Neil Harrison*
Return visit to The Elegant Minmon of PCW Issue one 44

PCW SMALL BUSINESS CASE STUDY *Boris Sedacca*
If you have a small business, read this. If you haven't read it all the same. 48

PCW OPEN PAGE *Mike Lord*
The Amateur Computer Club ... MUSE news 52

PET PREENING
Tasty morsels to keep your Pet purring 54

GETTING IT TOGETHER *Mike Banahan*
Continuing "your own assembler" 55

BUZZWORDS *Peter Reynolds*
The buzzing of the B'zz 60

TIME TABLING FOR SCHOOLS *Charles Sweeten*
What to do to avoid having two teachers in the same classroom at the same time teaching two different subjects, and with no pupils around. 62

Editorial and Advertising Office:
62A Westbourne Grove, London W2
Phone: 01-229 5599 (Publisher)
01-727 8758 (Editorial and Advertising)

Publisher:
A. Zgorelec
Editor:
Meyer N. Solomon
Editorial Assistant:
Roger C. Wilkins
Policy Advisor:
Peter Crofton-Sleigh, FRAS

Consultants:
John Coll, Mike Dennis, Neil Harrison, Charles Sweeten, Patrick Sutton, Michael James, R.W. Davy, David Hebditch, Sheridan Williams.
Art: **Sauveur Laurent Sant, Kathryn Hamme**
Secretarial:
Vanessa Blackburn Kiddle
Layout Consultants:
D. Norris, T. Gabos
PCW Photography: **Yoshi Imamura, Peter McGee**
Typesetting & Artwork: **Gilfillan**
Cover painting: **Sant**

CONTRIBUTORS:

We welcome interesting articles written simply and clearly. You need not be a specialist to write for us. MS should not be more than 3000 words long, lines double spaced, with wide margins. Line drawings and photographs wherever possible. Enclose a stamped self-addressed envelope if you would like your article returned.

Manufacturers, suppliers and dealers are welcome to contribute technical articles, and send product information, but we are pledged to an independent viewpoint and will publish evaluations and reasoned criticism or praise, space permitting. Naturally there will be right of reply. Views expressed in articles are not necessarily those of Personal Computer World.

We may make arrangements to offer our readers products at special prices, for a limited period, in line with the policy outlined above.

Published monthly by Intra Press, 62A Westbourne Grove, London W2. Phone: 01-229 5599. Contents fully protected by copyright. All rights reserved. Subscription rates: Britain £8 for 12 issues. Prices include postage. USA – $10 for six issues, $20 for 12 issues. Continent and elsewhere: £9.80 for twelve issues. Prices include postage.
Printed by Carlisle Web Offset, 55 Conduit Street, Newtown Trading Estate, Carlisle CA2 7NR. Sole UK Distributors: Seymour Press Ltd., 334 Brixton Road, London S.W.9., England. Distribution to specialist shops by Intra Press.



I BYTE

BUT COMPUTERS DON'T

Come and see for yourself at

the BYTE SHOP Ltd

Stockists of the largest range of micro computers in the U.K.

Take the opportunity to experiment with and get to know any of the vast range of micro computers always in stock at The Byte Shop.

Whether you want a micro computer for your home, your business, for industry, for education - or if you'd just like to find out which model you get on with best - you'll find a visit to The Byte Shop a new and invaluable experience.

Call in at The Byte Shop any time from Monday to Saturday. It's right by Gants Hill tube station.

The Byte Shop 426/428 Cranbrook Rd., Gants Hill, Ilford, Essex. Telex 897311
Telephone 01-554 2177

DIODES/ZENERS

1N914	100v	10mA	.05
1N4005	600v	1A	.08
1N4007	1000v	1A	.15
1N4148	75v	10mA	.05
1N4733	5.1v	1 W Zener	.25
1N753A	6.2v	500 mW Zener	.25
1N758A	10v	"	.25
1N759A	12v	"	.25
1N5243	13v	"	.25
1N5244B	14v	"	.25
1N5245B	15v	"	.25

SOCKETS/BRIDGES

8-pin	pcb	.20	ww	.35
14-pin	pcb	.20	ww	.40
16-pin	pcb	.20	ww	.40
18-pin	pcb	.25	ww	.75
22-pin	pcb	.35	ww	.95
24-pin	pcb	.35	ww	.95
28-pin	pcb	.45	ww	1.25
40-pin	pcb	.50	ww	1.25
Molex pins .01		To-3 Sockets		.25
2 Amp Bridge		100-prv		.95
25 Amp Bridge		200-prv		1.95

TRANSISTORS, LEDS, etc.

2N2222	NPN (2N2222 Plastic .10)	.15
2N2907	PNP	.15
2N3906	PNP (Plastic - Unmarked)	.10
2N3904	NPN (Plastic - Unmarked)	.10
2N3054	NPN	.35
2N3055	NPN 15A 60v	.50
T1P125	PNP Darlington	.35
LED Green, Red, Clear, Yellow		.15
D.L.747	7 seg 5/8" High com-anode	1.95
MAN72	7 seg com-anode (Red)	1.25
MAN3610	7 seg com-anode (Orange)	1.25
MAN82A	7 seg com-anode (Yellow)	1.25
MAN74A	7 seg com-cathode (Red)	1.50
FND359	7 seg com-cathode (Red)	1.25

C MOS

4000	.15
4001	.15
4002	.20
4004	3.95
4006	.95
4007	.20
4008	.75
4009	.35
4010	.35
4011	.20
4012	.20
4013	.40
4014	.75
4015	.75
4016	.35
4017	.75
4018	.75
4019	.35
4020	.85
4021	.75
4022	.75
4023	.20
4024	.75
4025	.20
4026	1.95
4027	.35
4028	.75
4030	.35
4033	1.50
4034	2.45
4035	.75
4040	.75
4041	.69
4042	.65
4043	.50
4044	.65
4046	1.25
4049	.45
4050	.45
4066	.55
4069/74C04	.25
4071	.25
4081	.30
4082	.30
MC 14409	14.50
MC 14419	4.85
4511	.95
74C151	1.90

— T T L —

7400	.10	7473	.25	74176	.85	74H72	.35
7401	.15	7474	.30	74180	.55	74H101	.75
7402	.15	7475	.35	74181	2.25	74H103	.55
7403	.15	7476	.40	74182	.75	74H106	.95
7404	.10	7480	.55	74190	1.25		
7405	.25	7481	.75	74191	.95	74L00	.25
7406	.25	7483	.75	74192	.75	74L02	.20
7407	.55	7485	.55	74193	.85	74L03	.25
7408	.15	7486	.25	74194	.95	74L04	.30
7409	.15	7489	1.05	74195	.95	74L10	.20
7410	.15	7490	.45	74196	.95	74L20	.35
7411	.25	7491	.70	74197	.95	74L30	.45
7412	.25	7492	.45	74198	1.45	74L47	1.95
7413	.25	7493	.35	74221	1.00	74L51	.45
7414	.75	7494	.75	74367	.75	74L55	.65
7416	.25	7495	.60			74L72	.45
7417	.40	7496	.80	75108A	.35	74L73	.40
7420	.15	74100	1.15	75491	.50	74L74	.45
7426	.25	74107	.25	75492	.50	74L75	.55
7427	.25	74121	.35			74L93	.55
7430	.15	74122	.55			74L123	.85
7432	.20	74123	.35	74H00	.15		
7437	.20	74125	.45	74H01	.20	74S00	.35
7438	.20	74126	.35	74H04	.20	74S02	.35
7440	.20	74132	.75	74H05	.20	74S03	.25
7441	1.15	74141	.90	74H08	.35	74S04	.25
7442	.45	74150	.85	74H10	.35	74S05	.35
7443	.45	74151	.65	74H11	.25	74S08	.35
7444	.45	74153	.75	74H15	.45	74S10	.35
7445	.65	74154	.95	74H20	.25	74S11	.35
7446	.70	74156	.70	74H21	.25	74S20	.25
7447	.70	74157	.65	74H22	.40	74S40	.20
7448	.50	74161	.55	74H30	.20	74S50	.20
7450	.25	74163	.85	74H40	.25	74S51	.25
7451	.25	74164	.60	74H50	.25	74S64	.15
7453	.20	74165	1.10	74H51	.25	74S74	.35
7454	.25	74166	1.25	74H52	.15	74S112	.60
7460	.40	74175	.80	74H53J	.25	74S114	.65
7470	.45			74H55	.20		
7472	.40						

74S133	.40
74S140	.55
74S151	.30
74S153	.35
74S157	.75
74S158	.30
74S194	1.05
74S257 (8123)	1.05
74LS00	.20
74LS01	.20
74LS02	.20
74LS04	.20
74LS05	.25
74LS08	.25
74LS09	.25
74LS10	.25
74LS11	.25
74LS20	.20
74LS21	.25
74LS22	.25
74LS32	.25
74LS37	.25
74LS38	.35
74LS40	.30
74LS42	.65
74LS51	.35
74LS74	.35
74LS86	.35
74LS90	.55
74LS93	.55
74LS107	.40
74LS123	1.00
74LS151	.75
74LS153	.75
74LS157	.75
74LS164	1.00
74LS193	.95
74LS367	.75
74LS368	.65

9000 SERIES

9301	.85	95H03	1.10
9309	.35	9601	.20
9322	.65	9602	.45

LINEARS, REGULATORS, etc.

MCT2	.95	LM320T5	1.65	LM340K15	1.25	LM723	.40
8038	3.95	LM320T12	1.65	LM340K18	1.25	LM725N	2.50
LM201	.75	LM320T15	1.65	LM340K24	1.25	LM739	1.50
LM301	.45	LM324N	1.25	78L05	.75	LM741 (8-14)	.25
LM308 (Mini)	.95	LM339	.75	78L12	.75	LM747	1.10
LM309H	.65	7805 (340T5)	.95	78L15	.75	LM1307	1.25
LM309K (340K-5)	.85	LM340T12	.95	78M05	.75	LM1458	.65
LM310	.85	LM340T15	.95	LM373	2.95	LM3900	.50
LM311D (Mini)	.75	LM340T18	.95	LM380 (8-14 PIN)	.95	LM75451	.65
LM318 (Mini)	1.75	LM340T24	.95	LM709 (8, 14 PIN)	.25	NE555	.35
LM320K5 (7905)	1.65	LM340K12	1.25	LM711	.45	NE556	.85
LM320K12	1.65					NE565	.95
						NE566	1.25
						NE567	.95

MICRO'S, RAMS, CPU'S, E-PROMS

74S188	3.00	8214	8.95
1702A	4.50	8224	3.25
MM5314	3.00	8228	6.00
MM5316	3.50	8251	8.50
2102-1	1.45	8255	10.50
2102L-1	1.75	8T13	1.50
2114	9.50	8T23	1.50
TR1602B	3.95	8T24	2.00
TMS 4044-	9.95	8T97	1.00
		2107B-4	4.95
8080	8.95	2708	9.50
8212	2.95	Z80 PIO	8.50

INTEGRATED CIRCUITS UNLIMITED

7889 Clairemont Mesa Blvd., San Diego, CA 92111 U.S.A.

No Minimum

All prices in U.S. dollars. Please add postage to cover method of shipping. Orders over $100 (U.S.) will be shipped air no charge. Payment should be submitted with order in U.S. dollars. All IC's Prime/Guaranteed. All orders shipped same day received.

SPECIAL DISCOUNTS

Total Order	Deduct
$35 - $99	10%
$100 - $300	15%
$301 - $1000	20%

Phone (714) 278-4394 **BarclayCard / Access / American Express / BankAmericard / Visa / MasterCharge**



RESEARCH MACHINES 380Z

COMPUTER SYSTEMS

A professional microcomputer system with excellent support for scientific and educational users.

380Z systems come complete with:

- Z80A Processor
- RAM Memory
- 3K ROM Operating System
- VDU Interface
- Cassette Interface
- Parallel Input/Output
- Keyboard
- Room for Expansion
- Documentation

Only an inexpensive cassette recorder and a television are needed to make the system fully operational. No teletype or VDU terminal are required.

The standard 380Z system is expandable to include:

- Additional Memory
- Printers
- Serial Interfaces
- Parallel Interface with Real Time Clock
- Floppy Disc Systems
- Analogue Interface

The RESEARCH MACHINES 380Z is a professional general-purpose microcomputer. It has found broad application in Scientific Research and Education with a large number of systems in use in the field. The system is supported by an excellent range of software including Text Editor, Basics and Assemblers.

RESEARCH MACHINES 380Z System with 16K Bytes of RAM

£965.00

RESEARCH MACHINES 380Z System with 32K Bytes of RAM

£1158.00

RESEARCH MACHINES Computer Systems are distributed through SINTEL, P.O. Box 75, 209 Cowley Road, Oxford. Telephone: OXFORD (0865) 49791. Please contact SINTEL for the 380Z Information Leaflet. Prices do not include VAT @ 8% or Carriage.

Room P.W.29
313 Kingston Road, Ilford,
Essex, IGI 1PJ, England
01-553 1001

L.P. ENTERPRISES

From the representatives in Europe . . . for America's leading Micro-computer magazines and books, for the hobbyist, educationist and professional alike, we bring you a little light browsing!

Reading maketh a full man . . . . Francis Bacon (1561–1626)

Tick or indicate quantity ordered

From Adam Osborne Associates	
INTRODUCTION TO MICROCOMPUTERS	
Volume 0: The Beginners Book	£5.95
Volume 1: Basic Concepts	£5.95
Volume 2: Some Real Products (Revised Late 1977)	£11.95
6800 Programming for Logic Design	£5.95
8080 Programming for Logic Design	£5.95
Z80 Programming for Logic Design	£5.95
8080A/8085 Assembly Language Programming	£6.95
6800 Assembler Language Programming	£6.95
Some Common BASIC Programs	£5.95
BUSINESS PROGRAMS IN BASIC	
Payroll with Costing Accounting	£9.95
Accounts Payable & Accounts Receivable	£9.95
General Ledger (Available from late summer 1978)	£9.95
From Scelbi Computer Consulting Inc.	
6800 Software Gourmet Guide & Cookbook	£7.95
8080 Software Gourmet Guide & Cookbook	£7.95
8080 Programmers Pocket Guide	£2.25
8080 Hex Code Card	£2.25
8080 Octal Code Card	£2.25
8080 Guide and One 8080 Code Card	£4.20
8080 Guide and Both Code Cards	£6.00
Understanding Microcomputers & Small Computer Systems	£7.95
SCELBI 'BYTE' Primer	£9.95
8080 Standard Assembler (In Book Format)	£15.95
8080 Standard Editor (In Book Format)	£9.95
8080 Standard Monitor (In Book Format)	£7.95
8080 Galaxy Game (In Book Format)	£9.95
From Peoples Computer Company	
Reference Books of Personal & Home Computing	£4.95
What to Do After You Hit Return	£7.00
Dr. Dobbs Journal Volume 1	£10.00
From Kilobaud/73 Magazine Inc.	
Hobby Computers Are Here	£3.95
New Hobby Computers	£3.95
From Dymax Inc.	
Instant BASIC by Jerald R.Brown	£4.95
Your Home Computer by James White	£4.95
My Computer Like Me . . . When I Speak BASIC by Bob Albrecht	£1.65
Games with a Pocket Calculator by Thiagarajan & Stilovitch	£1.75
Games, Tricks and Puzzles for a Hand Calculator by W.Judd	£2.49
From BYTE Publications Inc.	
Paperbytes:	
Tiny Assembler for 6800 Systems	£5.75
Bar Code Loader for 6800, 8080, Z80 & 6502 Micros	£1.75
Best of BYTE Volume 1	£8.95

Tick or indicate quantity ordered.

	Price UK	Price Overseas If Different
From Creative Computing Press		
Best of Creative Computing Volume 1	£6.95	
Best of Creative Computing Volume 2	£6.95	
101 BASIC Games (Revised & Reprinted Feb.78)	£5.50	
The Colossal Computer Cartoon Book	£3.95	
Computer-Rage (A new Board Game)	£6.95	
Artist and Computer	£3.95	
From Everyone Else		
Magazine Storage boxes (holds 12 minimum)	£1.25	
Sybex: Introduction to Personal and Business Computing	£4.95	
Sybex: Microprocessors from Chips to Systems by R.Zacs	£7.95	
Sybex: Microprocessors Interfacing Techniques by R.Zacs	£7.95	
Dilithium: Home Computers Volume 1: Hardware	£6.50	
Dilithium: Home Computers Volume 2: Software	£5.95	
Getting involved with your Own Computer	£4.75	
TV Typewriter Cookbook by Don Lancaster	£7.50	
TTL Cookbook	£7.50	
CMOS Cookbook	£7.95	
IC Timer Cookbook	£7.50	
IC OP–AMP Cookbook	£9.50	
RTL Cookbook	£4.25	
Z-80 Microprocessor Handbook	£7.50	
Computer Programs that work (In BASIC)	£2.55	
From Basic Software Library (from Scientific Research Instruments)		
Vol 1: Business and Personal Booking Programs	£17.50	
Vol 2: Maths and Engineering Programs	£17.50	
Vol 3: Advanced Business Programs	£26.95	
Vol 4: General Purpose Programs	£7.95	
Vol 5: Experimenters Programs (General Purpose)	£7.95	
Vol 6: General Ledger Program	£32.50	
Vol 7: Professional Programs	£26.95	
Magazines: Back Issues		
Personal Computing	£1.75	
Interface Age	£2.25	
Dr. Dobbs Journal	£1.75	
Computer Music Journal	£2.50	
Peoples Computers	£1.75	
*BYTE	£2.25	
Creative Computing	£1.75	
Calculators & Computers	£1.75	
ROM	£1.75	
Kilobaud	£2.25	
73	£2.25	
6502 Journal	£1.75	
MAGAZINES: Subscriptions		
Personal Computing (Twelve Issues Yearly)	£16.00	£17.00
Interface Age (Twelve Issues Yearly)	£20.00	£20.50
Dr. Dobbs Journal (Ten Issues Yearly)	£13.00	£13.50
Computer Music Journal (Four Issues Yearly)	£8.50	£9.00
Peoples Computers (Six Issues Yearly)	£8.00	£8.50
Kilobaud (Twelve Issues Yearly)	£20.00	£21.00
*BYTE (Twelve Issues Yearly) via USA	£15.00	
BYTE (12 issues yearly) via U.K.	£21.00	
Creative Computing (Six Issues Yearly)	£8.50	£9.00
Creative Computing (Twelve issues yearly)	£16.00	£17.00
Calculators & Computers (Seven Issues Yearly)	£10.00	£10.50
73 (Twelve Issues Yearly)	£20.00	£21.00
6502 Journal (Six issues Yearly)	£9.00	£9.50

THIS LIST CANCELS ALL PREVIOUS PRICE LISTS: EFFECTIVE AUGUST 1978

PCW/878/VC

HOW TO ORDER

Please note our prices include postage and packing, but not insurance, if wanted add 12p for every £10. of **books** ordered. Make cheques, PO's etc. payable to:–

L.P. Enterprises.

CREDIT CARDS accepted
BARCLAYCARD VISA / ACCESS
DINERS CLUB / AMERICAN EXPRESS

Phone: 01-553 1001 for Credit Card orders (24–hr service)

Send to address above — All Orders must be Prepaid

Indicate Payment Method:

Total Enclosed £ . . . . . .

. . . . . My cheque, P.O., I.M.O. is enclosed in **Sterling** on **U.K. Bank**

. . . . . Charge to Barclaycard/Visa/Access/Diners/American Express

Credit Card No . . . . . Expiry Date . . . . . .

Name . . . . . .

Address . . . . . .

. . . . . POSTCODE . . . . . .

Signature . . . . . .

All publications are published in U.S.A. and shipped air-freight by **L.P. Enterprises.** In unusual cases, processing may exceed 30 days.
*BYTE subscriptions are processed in U.S.A. and are air-freighted & posted from Amsterdam and will take 3 months to start.
U.K. Subscriptions start within 3 weeks.

TRADE ENQUIRIES WELCOME

Publisher's Letter

Dear Reader,

I went to the Personal Computing and Small Business Computer Show held in Philadelphia this August, and received a warm reception from the Show's director, John H Dilks III. Our magazine went down very well with the Americans.

This, the sixth and Show issue of **PCW**, is a milestone for personal computing in Europe. We are able now to say with confidence that **PCW** is read by people who command respect in their professions: Businessmen, academics, researchers, engineers, doctors, lawyers, accountants, computer specialists, journalists . . . and the corollary is that there is a strong base in Europe for personal computing.

Editorial

Thou shalt not lose face

That seems to be the first commandment in publishing. Which explains, I suppose, why a "good" editor is worth his weight in the words needed to explain away modify or suppress criticism.

I have in front of me a long letter from a reader. In it he comments on the first few issues of **PCW**. He doesn't make a polite genuflection before expressing his forthright opinions on articles and authors. He's taken us at our word and challenges us to publish the letter.

Now should we take the risk and publish? Should we say to **Mr. A. J. Aylward,** "Good letter, but it's over long?" This in spite of its racy style and flashes of humour? Should we quietly forget our commitment to be sensitive to readers' opinions, especially when these opinions are critical?

The title I have in mind for the letter: "Donner und blitzen!"

Micro Elephant?

The British Government is financing *Inmos* to do for Britain's micro industry what Silicon Valley is doing for the U.S. Though this baby elephant hasn't been delivered yet, swarms of painters are dipping their brushes in white paint.

My paint brush is poised, but I've halted in mid-stroke. After all, even elephants grow to different sizes. In two to three years – roughly the time *Inmos* will be ready to trumpet its hardware – worldwide demand will be so tremendous, and applications will be so varied, that any halfway decent product will find a niche in the market. The really encouraging thing is that the track record of the people involved with *Inmos* ensures that its products will be far more than halfway decent. But I still haven't put down my paint brush.

LOOK !

Extra RAM for your PET

★ New LOW prices:– 16Kbytes £328
★ Simple internal plug-in board 24Kbytes £388
★ Full manual with BASIC test 32Kbytes £438
★ 6 month guarantee
★ Available NOW (Prices exclude VAT)

IJJ Design Ltd.
37 London Road, Marlborough, Wilts
SN8 2AA

PCW Book Review



SOME COMMON BASIC PROGRAMS

L. Poole, M. Borchers, 1977 ; 193 pages,
(Adam Osborne & Associates 8½ x 11, £5.95)

BASIC is an easy language to use. Why then should anyone buy a book full of listings of "common" applications programs written in BASIC? The answer is simple. Even the best programmer cannot expect to understand all the theory necessary to solve problems in mathematics/economics/statistics etc. "Some Common BASIC Programs" supplies easy answers in each of these fields and many more.

The large range of programs, 77 in all, makes it impossible to list them all here. Roughly speaking the programs are grouped into sections:– Commercial (e.g. Term of a Loan, Earned Interest Table), Mathematics (e.g. Parts of a Triangle, Coordinate Plot, Matrix Inversion), Statistics (e.g. Mann-Whitney U, Chi-Squared, Multiple Linear Regression), General (e.g. Recipe Cost, Day of the Week, Anglo to Metric).

Each program is well commented and an example of how to use it is included. Modification for particular applications not covered should be easy. One reservation is the number of possible errors. In the few programs that I have used to date there have been one or two errors (e.g. in Anglo to Metric and Multiple Regression). These have proved easy to fix once spotted, but indicate that the book should be used with care. However at 7.72p per program this collection should be ideal for anyone using BASIC.

Michael James

NOTE TO POTENTIAL CONTRIBUTORS

If you are sending listings for reproduction they must be absolutely clear. This can be achieved by the use of a carbon ribbon in the Printer.

ATTENTION PET OWNERS!

Phone or write **NOW** for the most exciting catalogue of **PET** games around. (FREE)

THE LONE KNIGHT (£5.00) – The most intriguing and difficult chess-orientated puzzle ever.

ACEY-DUCEY (£6.00) – A must for anyone with a gambling streak! "Real" cards display.

ONE ARM PET (£6.00) – The most sophisticated Bandit ever. Hold! Shoot for Gold or Silver!! Terrific graphics.

HANGMAN (£4.00) – A pleasure (ouch!) to be executed by **PET.** Over 300 "in memory" Words and definitions. Very smart display.

CHINESE FAN TAN (£6.00) – "Crap" game for 2 to 8 players. Ideal for parties. Totally addictive!

AND MANY MORE . . .

Send PO, IMO, Cheques made payable to:
MINI MICRO 47 Queens Road, LONDON, N11 2QP.

ACCESS MASTERCHARGE EUROCARD welcome.
Just send or phone card number (01-889 7615 9 a.m. to 7 p.m.). Orders and enquiries accepted Worldwide.

EXAR IC

XR205	5.75
XR215	3.60
XR320	1.10
XRL555	1.10
XR1310p	.99
XR2206	3.98
XR2207	2.98
XR2208	4.40
XR2209	1.10
XR2212	3.70
XR2216	5.40
XR2240	2.90
XR2242	.99
XR2272	.99
XR2567	2.24
XR4136	.99
XR4151	2.24
XR4194	3.22
XR4195	1.18
XR4741	1.38

Mostek IC's

MK5009	6.98
MK50240	8.12
MK50395	9.98
MK50396	9.98
MK50397	9.98
MK50398	7.48
MK50399	7.98

DISPLAYS

FND500cc	.95
FND501+%	.95
FND507ca	.95
DL701+%	.95
DL704cc	.95
DL707ca	.95
DL750cc	.98
DL747ca	.98

Signetics NE/IC

NE510	1.68
NE515	2.15
NE521	1.74
NE522	1.74
NE526	3.62
NE527	1.49
NE529	1.72
NE531	1.10
NE532	.62
NE535	.89
NE536	1.89
NE540	1.58
NE541	1.65
NE544	1.65
NE546	1.10
NE550	.89
NE555	.29
NE556	.82
NE558	1.18
NE559	1.25
NE560	2.45
NE565	.98
NE566	.98
NE567	.98
NE570	3.20
NE571	2.80
NE590	1.28
NE592	.98

RCA-CA Type

CA2111ae	1.70
CA3021	2.35
CA3023	2.35
CA3028	.89
CA3046	.79
CA3051	2.24
CA3052	2.10
CA3059	1.78
CA3060e	2.28
CA3062	3.25
CA3075	1.98
CA3079	.98
CA3080e	.82
CA3081	1.05
CA3082	1.05
CA3086	.58
CA3089e	2.68
CA3090aq	3.45
CA3100e	1.35
CA3118t	1.38
CA313ot	.98
CA313or	.98
CA314ot	.98
CA3140n	.98
CA3160e	.98
CA3161	2.55
CA3162	6.75
CA3189e	3.15
CA3240e	.98
CA3290ae	2.40
40673	.88
40841	.77

IC Sockets Texas

8 pin	.15
14 pin	.17
16 pin	.19
18 pin	.22
20 pin	.29
22 pin	.36
24 pin	.39
28 pin	.46
40 pin	.49

Wire Wrap

14 pin	.42
16 pin	.49

Voltage Regulators.

LM7805–12–15–18–24V=	.79ea.
LM7905–12–15–18–24V=	.99ea.
LM317T–1.90ea./LM337n	2.89ea.
LM350K–6.78ea./LM723n	.48ea.
Special Offer=7806 only=	.78ea.

Stop-Shop at Micro-Pop

1702A-EPROM, 256 x 8, 450ns	5.95
2101p–M/Ram, 256 x 4, 450ns	1.35
2102p–M/Ram, 1024 x 1, 450ns	1.35
2111p–M/Ram, 256 x 4, 450ns	1.35
2112b–M/Ram, 256 x 4, 500ns	1.45
2114p–S/Ram, 1024 x 4, 450ns	9.75
2704a–EPROM, 512 x 8, 450ns	8.75
2708a–EPROM, 1024 x 8, 450ns	9.85
2716p–EPROM, 2048 x 8, 450ns	18.40
4104n–S/Ram, 4096 x 1, 350ns	9.85
5204Q–EPROM, 512 x 8, 500ns	9.85
TMS6011c–UART, 5v, – 12v	4.35
INS8154n–128 x 8 Ram I/O	7.95
Z80 CPU Mostek	13.65
Z81 PIO Mostek	8.60
Z82 CTC Mostek	8.60
8080a-8 bit CPU	6.45
8085a-8 bit CPU X 5V	13.95
TMS9980-16 bit CPU	19.95
ISP8A/600n-SCMp II	9.45
8212p-8 bit I/O port	2.90
8216p-Non. Inv.B/Driver	2.40
8224p-Clk. Generator	3.75
8226p-Inv/Bi-Drt/B-Drv.	2.20
8229p-System Contrler	4.65
8251-USART Communicat	7.75
8253p-Interval Timer	14.40
8255p-Prog. Interface	4.65
8279p-Keyboard Interface	9.65
SFF. 96364-TV-CRT Cont.	14.65
25LS2521-8 bit Comprt.	3.90
RO3-2513-5x7, Chr. Gen.	5.95
MC1480p-Quad L/Driver	.90
MC1480p-Quad L/Receiver	.90
AY5-9200-10x22 Rep. Dial	9.60

MFG's 100% Guarantee as Texas, Intel-NEC-AMD-GI-National.

LED's 3mm + 5mm

Red 3mm+5mm	.14
Gr. 3mm+5mm	.17
Yell. 3mm+5mm	.17

IC's Incredible

TBA120a	.78
TBA120s	.98
TBA231n	.85
TCA730	2.65
TCA740	2.85
TDA1022	5.89
TDA1034	3.95
TDA1037	.95
TDA2002	2.40
S041p	.89
S042p	.89
UAA170+180=	1.35
95H90 =	7.98
MM5316n=	3.85
CT7004=	7.80
ZN1040e=	7.65

C-MOS HEF/BP + SSS/BP

4000	.12
4001	.12
4002	.12
4006	.49
4007	.14
4008	.49
4009/4049	
4010/4050	
4011	.12
4012	.15
4013	.19
4014	.49
4015	.77
4016	.27
4017	.47
4018	.47
4019	.37
4020	.65
4021	.59
4022	.54
4023	.14
4024	.44
4025	.15
4026	1.20
4027	.29
4028	.49
4029	.64
4030	.15
4031	1.20
4033	1.20
4035	.62
4036	1.25
4040	.69
4041	.49
4042	.49
4043	.49
4044	.49
4046	.59
4047	.49
4049	.22
4050	.22
4051	.62
4052	.62
4053	1.45
4059	6.45

Low Power Schottky + TTL – Signetics & Texas Only

74 LS00	.14
LS01	.15
LS02	.15
LS03	.15
LS04	.16
LS05	.17
7406	.19
7407	.19
LS08	.17
LS09	.17
LS10	.17
LS11	.17
LS12	.19
LS13	.22
LS14	.39
LS15	.19
7416	.23
7417	.23
LS20	.19
LS21	.19
LS22	.23
LS26	.26
LS27	.32
LS28	.28
LS30	.19
LS32	.19
LS33	.19
LS37	.28
LS38	.28
LS40	.19
LS42	.42
7443	.49
7445	.59
LS47	.76
LS48	.86
LS49	.86
7450	.19
LS51	.19
7453	.19
LS54	.17
LS55	.17
7460	.19
LS63	.87

7470	.28
7472	.28
LS73	.28
LS74	.29
LS75	.39
LS76	.29
LS78	.35
7480	.38
LS83	.69
LS85	.69
LS86	.29
7489	1.98
LS90	.39
LS91	.89
LS92	.39
LS93	.39
7494	.59
LS95	.58
LS96	.58
74100	.94
LS107	.32
LS109	.32
LS112	.32
LS113	.32
LS114	.32
74116	.98
74119	1.65
74121	.39
LS122	.39
LS123	.59
LS124	1.10
LS125	.39
LS126	.39
74128	.47
LS132	.59
LS133	.49
LS136	3.50
LS138	.59
LS139	.59
74141	.47

74 LS145	.59
LS147	1.70
LS148	1.20
74150	.98
LS151	.69
LS153	.69
LS154	.99
LS155	.69
LS156	.69
LS157	.69
LS158	.69
LS160	1.20
LS161	1.20
LS162	1.20
LS163	.88
LS164	.88
LS165	1.20
LS166	1.35
LS170	1.35
LS173	.99
LS174	.99
LS175	.99
74177	1.25
74180	.99
LS181	1.35
74182	.99
LS183	2.35
LS190	.69
LS191	.69
LS192	1.10
LS193	1.10
LS194	1.10
LS195	1.25
LS196	1.25
LS197	1.20
74198	1.47
74199	1.47
LS221	.99
LS240	.145
LS241	1.65

LS242	1.55
LS243	1.55
LS244	1.25
LS245	1.95
LS247	1.25
LS248	1.25
LS249	1.25
LS251	.99
LS253	.99
LS257	.99
LS258	.99
LS259	.88
LS260	.27
LS261	1.10
LS266	.37
LS273	.97
LS275	6.68
LS279	1.10
LS283	.98
LS290	.98
LS293	.98
LS295	.98
LS298	.98
LS299	1.58
LS324	1.10
LS325	.99
LS326	.99
LS327	.99
LS352	1.10
LS365	.42
LS366	.46
LS367	.46
LS368	.47
LS373	1.98
LS375	.49
LS377	.82
LS378	.98
LS386	.29
LS390	1.98
LS395	.98
LS399	1.14
LS670	1.04

Delay per Gate: 7400-10ns/10mW 74LS-10ns/2mW

10% for orders over 35, – Postage .70p – VAT Included

4060	1.35	4517	1.98	4161	.96
4063	.99	4518	.76	4162	.98
4066	.39	4519	.49	4163	.98
4067	1.95	4520	.78	4192	.98
4068	.15	4521	1.38		
4069	.15	4522	1.28		
4070	.15	4526	1.28		
4071	.15	4528	.78		
4072	.15	4531	.98		
4073	.15	4532	1.38		
4075	.15	4539	.68		
4076	.59	4543	.98		
4077	.15	4555	.78		
4078	.15	4556	.78		
4081	.16	4557	1.58		
4082	.16	4585	.98		
4085	.42	40161	.99		
4086	.42	40162	.99		
4093	.42	40163	.99		
4502	.86	40174	.62		
4510	.99	40175	.62		
4511	.90	40192	.69		
4512	.90	40193	.78		
4514	1.65	40194	.78		
4515	1.65	40195	.78		
4516	.99	4160	.96		

4193	.98
HEF4739	14.70
GZFI200	

10% for Orders over 50,

E.W.S. Electronics

Bulowsvej 40 . DK-1870 Kobenhavn V
Tlf. 01-*371412 · 373247
Telex: 22408 eltron dk

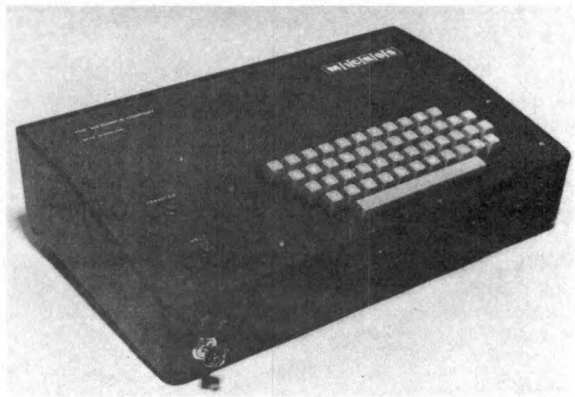
ANNOUNCING THE MICROS

JOIN THE MICRO REVOLUTION

Access — We welcome Access

master charge — THE INTERBANK CARD

EUROCARD



£399 for a Z80 based microcomputer, built and tested

Designed for educational establishments, personal computing and small business users

* Includes 1K monitor Eprom, 47 key solid state keyboard, video, TV, cassette and teletypewriter interfaces,serial i/o, 2 parallel i/o ports, 2K bytes RAM, power supplies and instrument housing.
* Connect to domestic TV or video monitor to complete the system
* 48 x 16 character video matrix
* 47 key contactless ASC11 keyboard
* Hard copy on teletypewriter
* 2 TTL compatible parallel i/o ports
* RS232 serial i/o port
* Load and dump programmes on unmodified cassette recorder
* Up to 16K byte mixed RAM and Eprom in table top housing
* Expandable up to 64K bytes
* Security locked power switch
* British designed and built
* Available in kit form for £360
* Credit terms available

THE MICRONICS COMPANY

1, STATION ROAD TWICKENHAM MIDDLESEX

8972 7044

PART OF THE MICRO REVOLUTION

Prices exclusive of VAT and carriage

Letters

PETulant ?

I read John Coll's review of the Pet 2001 in issue 2 with interest but with increasing dismay the further I progressed. He starts by saying "Let me give you my overall impressions of the Pet 2001" etc., and proceeds to do so but based on (his admission) only one day's use. This would be fine if followed up with an in depth evaluation such as accorded to Tandy's TRS-80. However John Coll's treatment of the Pet far from being objective appeared to be more a series of digs at CBM itself, apparent in his opening paragraph and especially so in the article conclusion.

To my mind, far from being objective Mr. Coll raised serious doubts, not with the machine itself, but about the ethics of the company producing it.

I have no axe to grind but I am very interested in purchasing a microcomputer and I am especially interested in obtaining *unbiased* information on the Pet 2001. Mr. Coll has unfortunately left me with the feeling that all was not well with CBM's Pet and possibly his relationship with CBM itself.

That the Pet has disadvantages I agree. It also has a lot of very advantageous features as well, and despite its "individual" features a large software market has already opened up in various parts of the country with what seems to be an endless supply of programme material available at a reasonable cost.

In closing I would like to point out that CBM have available at present a printer at £459.00 inc. VAT and a second cassette at £59.40. This information appears in their Official Price List (April 1978). Also of interest is that 6500 KIM Programming & Hardware Manuals are also available at £5 each.

Please let us have an owner's report or an in depth review of the Pet which I'm sure will be popular, and keep up the high standards of your magazine (to which I subscribe) remembering that what is written in PCW is going to influence a vast number of absolute novices to computing.

G. D. Compton,
Somerset.

John Coll replies: I agree with the writer. The simple fact is that CBM would not provide the facilities to enable an in-depth review. I regret this as much as he does.

FLOORED !

I have just read the article 'Computing for Everybody' by Magnus Magnusson in the August issue of PCW. If any of your readers are intending to follow one of Mr. Magnusson's ideas by hooking their computer into ours I should ask them if they could try and keep the wiring as tidy as possible and use the false flooring provided.

Pete Harris,
Computer Services,
The Financial Times.

THE DOCTOR AND COMPUTING

Like your contributor Robert Johnson, I believe there is enormous scope for the use of microprocessors in General Practice.

I was surprised by his estimate of the small amount of tape required to log a year's work, but with reasonably wide categories for symptoms, signs, past history etc., it should be possible. It would be difficult, however, to record *all* symptoms systematically, considering the number of symptoms presented with, for example, a common cold.

There is one omission in the article which I believe is the greatest bar to the full use of the data for early recognition of side-effects -- the problem of confidentiality. It is axiomatic that any information a doctor receives in the course of a consultation may not be communicated to any other person without the consent of the patient. It is equally axiomatic that the central computer used to detect side effects must be able to link up the prescription of a drug at one time with the appearance of a new symptom at another, and therefore that each patient should be uniquely identifiable. The prospect of meticulously recorded medical information on the whole population being available to a government department is not a happy one. Would *you* sign away your right to confidentiality?

On the other hand, I am in no doubt about the value of data-handling devices in general practice, particularly now that micros make it possible for GP's to do their own data-handling, without relying on time-sharing and its attendant security risks.

Finally, I should be most interested to correspond with anyone else who is interested in this field – as a newcomer I need all the help I can get!

Dr. Michael Hendry,
Kenilworth,
East Road,
Cupar, Fife.

IN DEFENCE OF THE EUROPA-BUS.

Some time ago it was proposed that a new bus standard be introduced, in order to keep pace with recent developments in the Microprocessor field since the early days of the 8080 and its associated S100 bus system. The proposed E78 Europabus has advantages over many other systems in that it uses standard indirect connectors which, although expensive, are readily available, and in that the system is flexible and is readily adapted to several microprocessor types and configurations. Clearly it cannot cater for all eventualities, multiprocessor systems have been mentioned as difficult to adapt, but neither can any of the other established systems, S100, MUBUS et al.

As a member of the Southampton University ACC, SIG.99. design team I have been mostly concerned with the development of the bus for the 9900. Other groups elsewhere are working with other processors. We have decided that we will use the basic 64/64 configuration, with minor modifications, for the majority of signals. Other signals, the CRU bus and the 15 Interrupt lines, plus a few miscellaneous signals, will use a second connector. Regarding the number of pins. I do not know of any 96 way connectors. The 64/96 connector differs in having a greater spacing between the rows compared with the 64/64. (See Vero Catalogue for details).

If you do not wish to utilize the full double Eurocard system for certain peripheral boards, I would suggest using single cards. This is easily accomplished using, for example, the Vero KM4C series of cases. Of course there is a problem with the number of pins but certain peripheral boards could be non-standard. One other solution is to include extra wires or connectors in addition to the standard connector. Why not include a 24 way ribbon cable to interface, say to a floppy? Why not connect the board to a 'module' assembly, with standard connection at the rear, and V24 or RS232 type connector at the front?

Most of the problems can be, or have been, solved. Some microprocessor types have almost been finalised for use with the bus, in particular the 9900. The standard is almost here. Why not use it?

P. D. Maddison,
Hon. Pres. SUACC
Pres. SACC.
Southampton University

"NEIGHBOURHOOD CONSULTANCY"

Obviously David Francis (P.C.W., August, 1978) gets lots of fun out of "Neighbourhood Consultancy", and he certainly gives some good advice on methods. However, it really isn't as simple as that; a few caveats need to be entered. He suggests you should "get into some basic Operational Research and Systems Analysis". I'll start with Operational Research, since that is my profession.

First, Operational Research is not just techniques (linear programming, stock control etc.). At the University of Aston Management Centre we have an Operational Research and Systems Analysis Subject Group which devotes a uniquely high proportion of its Courses in Operational Research and Systems to methods as opposed to techniques.

Second, Operational Research is not computing either. Some industrial Operational Research groups have their own computing expert in the group, and he is often the "odd man out". "Don't ask me about Operational Research" he'll say, "I'm a computer man!"

Thirdly, the textbooks on Operational Research and Systems are full of techniques, but aren't terribly helpful about methods. R. C. Tomlinson, referring to methodology in his Inaugural Address as President of the Operational Research Society in 1974 said "I personally do not find the existing textbooks adequate". My current research at Aston is intended to help fill some of these gaps.

Fourth, it is certainly possible to learn the methods associated with Operational Research, Systems Analysis and other consulting type work. It is next to impossible to learn by reading alone. The best ways of learning this sort of thing are still a topic of lively debate in, for example, the Operational Research Society.

My fifth and final caveat is about people – whether the skills of the home computer man will match those required of a consultant.

David Broad (Managing Director of Comart) has made a charming analysis of his customers into five types:–.

1. Home Brew Computerist. 2. Home Computerist. 3. Micro-Entrepreneur or Small Businessman. 4. Educationalist. 5. Government and Research Scientist.

I think this covers the people David Francis is addressing.

If you are the consulting sort (i.e. in business, the professions, Operational Research, or Systems Analysis, etc.), and need a micro as a tool, you will probably have got one already (type

"3" above). If you are in it for the fun (types "1" and "2" above), it is less likely that you will have the consulting skills or interests. Your computer, whatever its size, is a tool. Your micro won't make you a consultant, any more than a saw makes you a carpenter. The danger is of disappointing your client as well as yourself, and possible giving a bad name to the micro fraternity.

In the light of my own consulting work with micros (in conjunction with Bernard Tate of Beta Systems), and my research (at Aston), I would suggest that one needs to keep ones eyes (and mind) far more open than most people usually do.

In this age of technology, science, logic and reason, one is driven to see and think about things in a rather rigid way.

Indeed, this sort of thinking (technique rather than method oriented) is necessary to the home brew computerist.

Consultancy demands the opposite — seeing things through other people's eyes (as well as your own) and thinking of them in several different ways.

Yes, there are opportunites to grab, but we must grab them with eyes wide open.

R. N. Woolley, B.A., M.Sc.
11 Kerry Close,
Brierley Hill,
West Midlands DY5 3YW

A READER'S PLEA

I realise that, with the increasing circulation of your magazine, you will soon have to start rationing the space devoted to readers' letters, but I hope that you may consider my comments to be of interest, especially since I am perhaps rather untypical in background, being a recent 'convert' to computing who took a job at a leading systems house as an export consultant about a year ago, and then moved gradually into increasingly technical areas (a reversal of the usual career progression, I suppose!) I have been involved with hard-ware, soft-ware, and commercial aspects in roughly equal proporations, and do not, fortunately, encounter anything but the latest technology, having never even seen a punched card or an ICL 1900!

So much for background — my real reason for writing to you is that my experiences while trying to acquire a useable personal computer system might be of wider interest, might save some readers the same frustrating 'learning curve' and, best of all, might provoke some entrepreneur into coming up with solutions. (I am sure my employers could, but I cannot afford their fees!)

I have in fact come to the conclusion that there is no useable personal system on the U.K. market for less than £2500, and yet that there is no reason why this should be so, since my definition of 'useable' is pretty modest (no line printers or 100 Mb discs !) being based on the following requirements:—

Hard copy output — unobtainable for less than about £700, and even then you can only choose between an upper-case only dot-matrix contraption suitable only for invoices and listings or a Golf-ball conversion which no-one will want to service.

Video output — at least £500 for a green-and-black display that cannot even take an A4 page (for that you would need £1500) is hardly sensible. I have heard, and more or less understand, the arguments against a domestic TV on the grounds of bandwidth etc., yet I have found that, in practice, a display on a modern colour TV (with one of those black grids to improve subjective definition) is more readable than the blobby green squiggles which the 'commercial' alternatives provide.

Colour is not just a frivolous extra, it provides another information-carrying channel which can compensate for any shortcomings in characters-per-line, and also seems to encourage people to present data more imaginatively and graphically.

Yet there are hardly any colour TV interfaces on the U.K. market, despite the fact that any-one contemplating a serious personal system will certainly be able to afford a colour TV (at £200 they seem like peanuts compared to some of the prices mentioned earlier.)

Viewdata and Teletext. Even more extraordinary, indeed almost scandalous, is the fact that no one else seems to pay any attention to the need to make a reputable system at least compatible with these — whatever their shortcomings (on graphics, for example) they provide an unlimited supply of data for manipulation by the personal system, and there is no reason why the video display should not have other options (high-definition graphics etc.) available as well.

Personally, I do not intend to spend anything on an actual micro-computer until the above situation has improved (perhaps when the big companies start churning out viewdata units with external DMA) but, when I do, I would start worrying about the following:

Storage. The falling price of RAM chips is the only good news. But anyone who has outgrown cassette storage (i.e. anyone who has owned their system for more than a week!) and who thinks that a £500 mini-floppy will solve their problems has a nasty shock coming to them — floppies are so unreliable that you will need *two* units anyway, and spend much of your time copying.

I do not know when this under-publicised problem will be solved, or how. Perhaps by bubble memories, or even the video cassette or disc (up to 100 Mb serial or 100 Mb random access for that £500) but in the short term I feel tempted to go for an EPROM programmer (and eraser) for the really vital soft-ware.

Interpreters. People seem to chase every last MHz of processor speed, only to slow their programs down by a factor of between 20 and 50 by using a 'Basic-in-ROM' instead of a compiler — but there are signs that this may improve.

Random number generator. This may not seem a key item, but you will have guessed from my emphasis on colour graphics and speed that I am interested in games, simulations, and computer art rather than 'DP'. It would seem to be so easy to implement a hard-ware random number source (for less than a tenner, I would think) that I cannot understand why people are still forced to use slow and often very non-random soft-ware routines (some of which are even 'sold' as debugging aids because they repeat the same sequence every time they are initialised!)

Real-time clock. Another little bit of hard-ware that would be so easy to offer as standard, and so much more useful than say, being able to connect 256 teletypes to your micro!

Finally, there are a couple of commercial aspects which should be considered by anyone who actually wants to use their system as opposed to endlessly assembling, testing, modifying and repairing it.

Kits. I may be prejudiced, but I feel that anyone who supplies a product in kit form *only* is merely evading the responsibilities he would have if he sold it as a complete product (guarantees, Sale of Goods Act, Trade Descriptions Act, etc.) In any case, kits never seen to be particularly cost-effective even if they work perfectly ever after — if they are any good they will soon start being mass-produced at prices below the original kit price (viz. TV games, calculators, etc.)

User base. The sad thing is that I will probably end up buying a second-rate system simply because a lot of other people already have it, just so that I can obtain soft-ware at the right price.

You will notice that there are a number of things I have omitted, the little matter of the processor being one! But, until someone starts coming up either with solutions to the above problems, or with price reductions so massive that they become tolerable, arguments about processor architecture, bus structure, etc., must remain of academic interest for anyone who is seriously contemplating investing their own money.

By the way, many thanks to Guy Kewney for his much needed debunking of some of the policies of U.K. suppliers (massive mark-ups over U.S. prices, but without the extra support or expertise that would justify them.)

C. A. G. Cary, M.A.
9 Eltisley Avenue,
Cambridge,
Cambs.

PCW Readers — please — space is our headache. We can't publish too many long letters, however interesting — **PCW.**

ROBOT — MAN

As a real newcomer to the world of computers, I have been reading PCW since its first issue. At the present moment I still feel as if I have picked up a magazine written in Sanskrit or some other obscure language but gradually I am learning to translate it into English.

I came to psychology via biology, and it is my interest in the biological sciences that really prompt me to write. In the August 78 PCW, W. V. Ringer gave us a stimulating article called "It's the thought that counts". However I think he went a little further than most biologists would be prepared to go at this stage in his exposition of the significance of DNA. He gave the impression that he thought instincts were somehow programmed into the DNA.

He was right in his description of DNA as a double helix, one from each parent but even that is an oversimplification of the facts. It is the order of the bases (nucleic acids) which determines the message it carries. Each group of three bases specify a particular amino acid after they have been transcribed into another kind of nucleic acid helix called (for short) RNA. They also specify 'stop' and 'start' signals. These amino acids are zipped together by the elimination of water by a tiny component of the living cell called a ribosome to make proteins. The proteins fold up into a shape determined by the forces which exist between the side atoms of the amino acids; for example Hydrogen atoms link up with Oxygen or Nitrogen, Sulphur atoms with Sulphur atoms. The shape of the protein determines what job it does; support, information transmission, etc. Besides proteins, cells contain and produce and use other complex materials like lipids and polysaccharides.

The proteins which are produced as a result of the translated and transcribed DNA, themselves can carry messages to other cells, but at what point 'Instinct' enters this chain of message carrying is difficult to see.

However, I am not quibbling over his basic thesis which is that man is the product of three factors: Genetic inheritance, environmental influence and soul. Though I doubt, come to think of it, if the theologians would go along with his definition of soul, but that is another letter.

Cyril D. Blount,
Teasdale House,
Easingwold,
York YO6 3PN

Tid Bits PRODUCTS . . . . COMPANY NEWS . . .

Data about Data

The relevance of data dictionaries and details of available systems are covered in a new book just published by The National Computing Centre. It is *Data Dictionary Systems* by J. D. Lomax.

The first section of this new book is devoted to explaining the purpose of data dictionaries, together with the implications and the practical issues involved, including implementation and running. It sets the scene for those investigating the subject, so that they can assess their requirement for DDS software. The book considers the question of whether to write one's own system or to buy a packaged product and proposes and examines a standard questionnaire for evaluating packages.

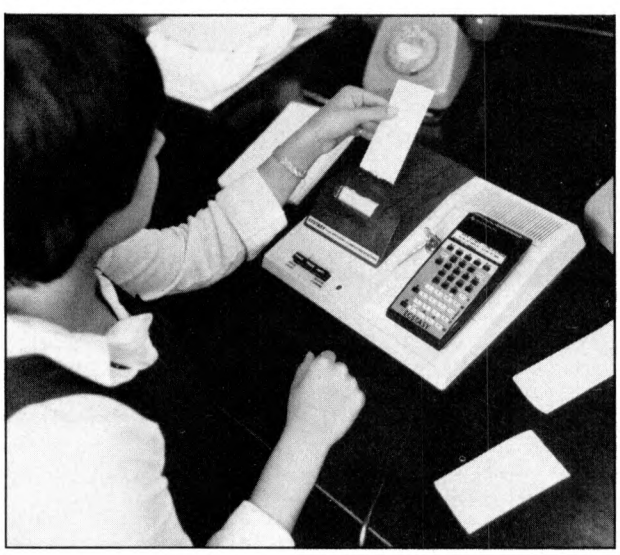
The second section, consists of eight available systems described in detail. The same procedure is used for each package and this is based on the standard questionnaire in the previous section.

Data Dictionary Systems by J. D. Lomax

ISBN 0 85012 191 4 price £12.00 128 pages

Available from technical bookshops or J. M. Dent & Sons (Distribution) Ltd., Dunhams Lane, Letchworth, Herts. (Cash + postage of £1.00 with order).

For further information
G. E. Hall or N. Candeland,
NCC Publications,
Telephone: 061 - 228 6333



The ECS Ecstasy machine, incorporating a Texas Instruments TI-58 programmable calculator with a 'Custom CROM' program module.

Custom-designed calculator module is key to new bookmakers' machine

A Texas Instruments TI-58 programmable calculator fitted with a special custom-designed program module forms the heart of a new machine designed to provide instant answers to the most complex betting problems experienced by bookmakers. The machine, known as ECSTASY, is being marketed by Efficient Computing Systems (ECS) of Douglas, Isle of Man, and is expected to become standard equipment in many betting shops.

The key to the betting-shop application is a device known as a 'Custom CROM' (Constant Read-Only Memory) – a small module containing a package of specially developed betting-shop programs which plugs into the back of the calculator. Dr. Les Waller, who is a consultant to ECS, and who has been involved for about ten years on the development of programs to ease bookmaking calculations, specified the CROM in such a way that the most complex variations of bets – doubles, trebles, block bets, each-way bets and 'anything-to-come' bets for a large number of selections – can be handled in a single operation, with a printout produced straight away on a built-in printer.

The ECS machine is the only one of its type currently available, and its success is due to the fact that Dr. Waller, a computer scientist by training, has been able to devise special algorithms to minimise the number of steps required for complicated betting calculations.

ECS is currently engaged in discussions with leading bookmakers on the use of the ECSTASY machine, and large orders are expected in the near future.

Further information:
John Gibbons, Texas Instruments Limited,
Telephone: Bedford (0234) 67466; or Dr. Les Waller, Middlesbrough (0642) 85399; or Peter Bush, Bush Steadman & Partners Limited, 4 Gold Street, Saffron Walden, Essex CB10 1EJ. Saffron Walden (0799) 23101/27240.

MEMEC OFFER:

16K Dynamic Ram. Z6116 Zilog's 16K Dynamic Ram.

The Z6116, 16K dynamic RAM organised as 16,384 word x 1 bit and packaged to industry standard in 16 pin D.I.L. is currently available with 250 and 200 nsecond access times.

The device is manufactured using ZILOG's double-poly ion implant silicon gate technology and features on-chip latches for address and data in and a common I/O capability using "early write".

All inputs are protected against static charge.

For full details and prices contact:
MEMEC (Memory & Electronic Components) LTD.
Thame Park Industrial Estate,
Thame, Oxon.
Telephone: Thame (084 421) 3416
Telex: 837508

Rapid Recall have 4K RAMs in Stock

In spite of recent comments about the shortage of 4K RAMs, Rapid Recall announce that they have substantial stock levels of a number of Intel 4K RAMs.

Harry Case, Rapid Recall's Managing Director, said "We have always claimed to carry the largest stocks of Intel products in the UK, but it is not until devices are in short supply that the extent of our stock levels becomes evident. Obviously we could not continue to supply these items in the face of an extended shortage, but our stock levels are sufficient to buffer the inevitable ups and downs that occur in the supply of semiconductor devices."

Further information:
John Weatherhead
Rapid Recall Ltd.
9 Betterton Street,
London WC2H 9BS
Telephone: 01 - 379 6741

MINI MICRO is a new firm and it will start its retailing activities in mid-September, initially by mail order, dealing exclusively with games.

Games programs will be, for the most part, generated in-house, but private contributions will of course be welcome.

Its first catalogue will be exclusive to CBM PET, but it will be presenting catalogues for a wide range of machines.

The greatest care will be taken in the debugging of programs, of course, as well as in the visual field, with displays designed especially to make games as attractive as possible.

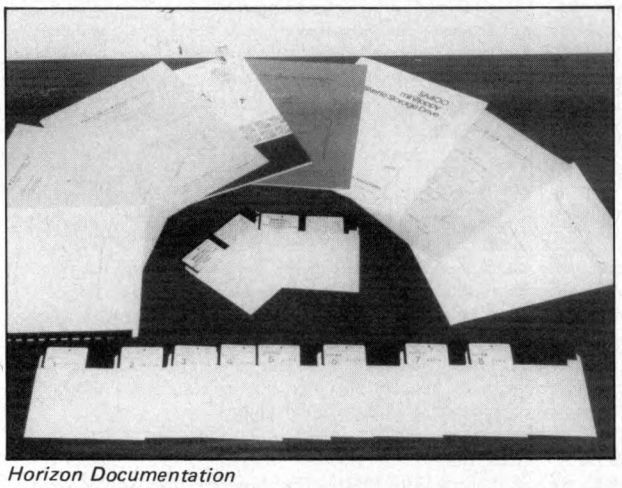
Further information:
Mini Micro,
47 Queens Road,
London N11 2QP
Telephone: 01 - 889 7615

COMART, specialists in S100 Microcomputer systems, now have available their Autumn 1978 Catalogue. Products in the catalogue for the first time include the Cromemco System Three, Single Card Computer and the SOL 20/16 Terminal Computer System. Prices of all Dynabyte memory modules have been reduced. The 16k byte dynamic memory card is now priced at £275 compared with the pre 1st August price of £310.

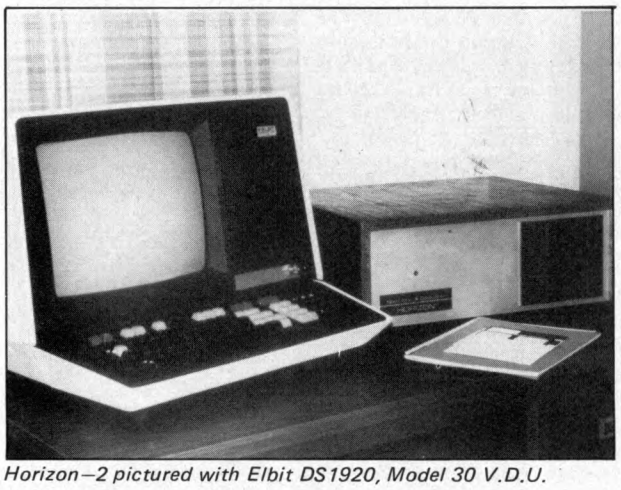
Catalogues are now available from Comart Limited, P.O. Box 2, St. Neots, Cambridgeshire PE19 4NY

The **HORIZON – 1** computer is made by North Star Computers of Berkeley, California. The Company also takes credit for the famous North Star Basic. The complete HORIZON –1 has a 16K RAM board, one minifloppy disk drive, 4MHz Z80A processor serial input/output port, and Extended Basic. The list is by no means exhaustive, and the upgraded HORIZON –2 offers two minifloppy drives. There is great software support, notably the North Star Software Exchange Newsletter.

Full details from the U.K. distributor:
Interam,
59 Moreton Street,
Victoria, London.
Telephone: 01-834 0261/2733



Horizon Documentation



Horizon–2 pictured with Elbit DS1920, Model 30 V.D.U.

GROWING EXPORT ORDERS FOR BRITISH MICRO-COMPUTER

Nascom Microcomputers launched its NASCOM – 1 microcomputer at the end of last year in face of intense competition from Japan and the USA. Since then the NASCOM –1 has become the country's fastest selling microcomputer, with orders approaching £2m. Three-quarters of the business has come from the rest of Europe and Scandinavia, particularly Germany, Holland, Sweden and Belgium.

NASCOM Seminar. Venue: Dragonara Hotel, Bristol.
Date: Saturday October 14 Time: 9.50 to 5.30.
Admission: £4.50 including VAT
Featuring: Five lectures, demos, open forum.
For further information:
Kerr Borland at Nascom Microcomputers
Telephone: (02405) 75151

Ocean Electronics seeing the need for a Low Cost floppy and controller have come up with a single card solution for PET users, 6500 family users 6800 and 8080 users. The difference is in the onboard PIA and Software supplied. The controller controls a single Floppy from Shugart, other floppies can be utilised, the Shugart SA 400 mini is used, also supplied is the software to control the PIA and Drive for your MPU family. Also full data on the 6500 (PET'S "Brain") and all the ICs in the 6500 family. Controller is expected to sell at £120 and SA400 at £225.

Write for details to:
Paul Wynter, Ocean Electronics,
Georgian House, 5 Bartholomews, The Lanes,
Brighton, Sussex BN1 1HG
Telephone: 0273-21952
Telex: 877159 RR Hove G.

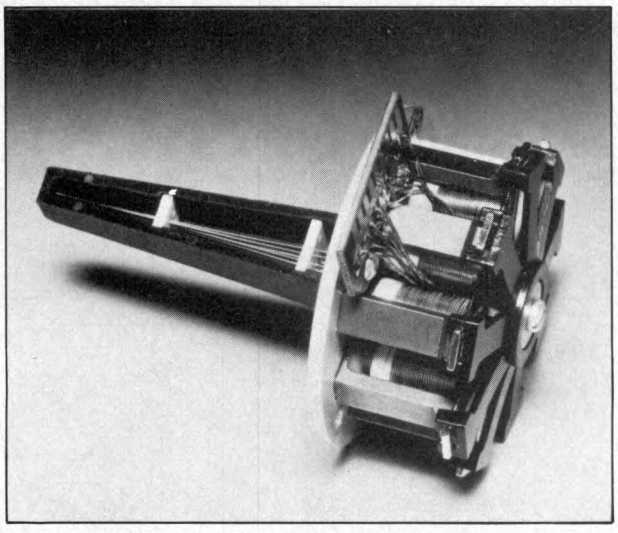
New Product: New Ballistic *9 wire Dot Matrix Print Head.

Peripheral Hardware announce a print head that employs an entirely new operating principle and which is manufactured to a very high engineering standard by Lear Siegler in America. Instead of solenoids with moving cores attached to the matrix wires, the Lear Siegler head from Peripheral Hardware uses simple electro-magnets to activate small hammers that ballistically propel the print wires. Since electromagnets use only a small fraction of the power that solenoids require, far less heat is generated. The head is therefore ideally suited for long, heavy-duty work cycles.

Since the wires are independent of the coils, the driving ends are arranged in a small, centrally located circle. This minimizes wire flexing and greatly simplifies guidance. As a result, the life expectancy of the print wires is substantially increased. This simplified guidance also eliminates tube clogging with inks, dust and paper fibres.

The Ballistic head is available in a 9 high wire configuration. The matrix can be utilized for printing both upper and lower case English character printing, plus underlining, symbols and special graphics. Foreign alphabets such as German, Arabic and Katakana are also possible.

*** Trade mark of Lear Siegler**
Contact:
Keith Searle
Peripheral Hardware Limited
Link House
Pool Close
West Molesey, Surrey KT8 0HW
Telephone: 01-941 4806



The new Lear Siegler print head.

SOFTWARE ARCHITECTS ANNOUNCE SALEM/1 A GENERAL PURPOSE EMULATOR FOR SERIES/1

Software Architects Limited are currently developing an ICL 7020 emulator as part of a major IBM Series/1 application for London Transport Executive.

The initial implementation of the emulator SALEM/1, due for handover in September 1978, will emulate a 7020 paper tape or card reader for the purpose of bulk data transfer to an ICL 1900 mainframe from a disk or diskette file on the Series/1. It will handle George 3 line disciplines by the emulation of teletype control commands.

The second phase of implementation will support bulk data from the 1900 to the Series/1 emulating a paper tape punch or line printer. It will also enable a 1900 mainframe operating under George 2 or Manual Executive to communicate with Series/1.

The range of peripherals will be extended to include the total range of 7020 and Series/1 devices. Any configuration of Series/1 and 7020 may then be specified. The handover date for this version will closely follow that of Phase 1.

SAL have already been contacted by numerous potential users of SALEM/1, both in the UK and Europe, as well as creating some interest amongst US software houses.

For more information, contact:
Jeff Goldsmith,
Software Architects Limited,
34-35 Dean Street,
London W1V 5AP
Telephone: 01 - 734 9402

Another new computer dealer:
Microdigital Ltd., of 25 Brunswick Street, Liverpool. Managing Director is Bruce Everiss, who is determined to be not only the first but also the most outstanding computer entrepreneur in Liverpool.

The range is wide: computers from the basic MK-14 to the Apple. Software will be "extensive" and will come from in-house as well as the elegant offerings of American software house GRT. Games programs will be on offer, as well as the musical micro doorbell CHROMACHIME. Literature on sale includes books and magazines.

Contact:
Bruce Everiss,
Telephone: 051 236 0707.

RESEARCH MACHINES 380Z – UPDATE INFORMATION

There have been several changes in the Research Machines 380Z since the PCW review of it in the June (No. 2) issue.

The operating system has been extended so that the following are standard on all 380Z and 280Z systems. Normal cassette I/O is now at 1200 baud, but the option of 300 baud (standard CUTS) is retained. The screen output, unless selected otherwise, is scrolled one page at a time, the next page being called by typing any key. Continuous scrolling and scrolling one line at a time are also possible and all the different modes can be selected under program control.

Printer and Serial Interface driving routines are now also included in the operating system, and direct connection may be made to parallel interface printers such as the PR40 or Centronics 700 range. Research Machines are themselves distributing the Centronics 779 and 701 and the Trend 800. Two low cost Serial Interfaces are offered for interfacing the 380Z to RS232/V24 or 20mA current loop teletypes or printers. These interfaces are bidirectional and a software routine is available allowing input from a paper tape reader. Another serial interface was designed for use in applications which require simultaneous input and output, for example when operating the 380Z as a terminal linked to an acoustic coupler.

Several specific criticisms or suggestions in the PCW review have been acted on: the fast cassette I/O rate is assumed as the normal rate, when the page is full in page mode the cursor blinks on and off, and loading can now be interrupted by a single key stroke at all times. The housing is completely new; a strong and attractive instrument case is used. There is the same room for system expansion and the case will fit into a standard 19" rack.

Further features have been added to the BASIC Interpreters. A version is available which includes the ability to read and write data files on cassette. It utilises the RML cassette file system and can be used with one or two cassette recorders. The RML Interactive Text Editor and the RML Absolute Assembler have been released. The Text Editor is character oriented and may be used in immediate mode; its uses include source program or data preparation. The Z80 assembler uses Zilog mnemonics and produces object code in either the industry standard 'Intel' format or the RML binary format. It contains its own text editor which uses a subset of the commands available in RML's full Text Editor.

Research Machines Ltd.,
PO Box 75,
Oxford

THE BYTE SHOP of 426/428 Cranbrook Road, Gants Hill, Ilford, Essex (01 - 554 2177) is a vigorous newcomer to the retail computer market. It offers an enviable range of hardware and software, details of which are given in an excellent catalogue, available on request from the address above. The Director, Bill Cannings, has very obviously studied the needs of potential clients carefully and this is reflected in the BYTE SHOP's offer of unbiased advice, hands-on experience for customers, no sales pressure, software, repairs, and post-buying support. The Shop is run by the able and energetic Vince Coen.

NEW PRODUCT

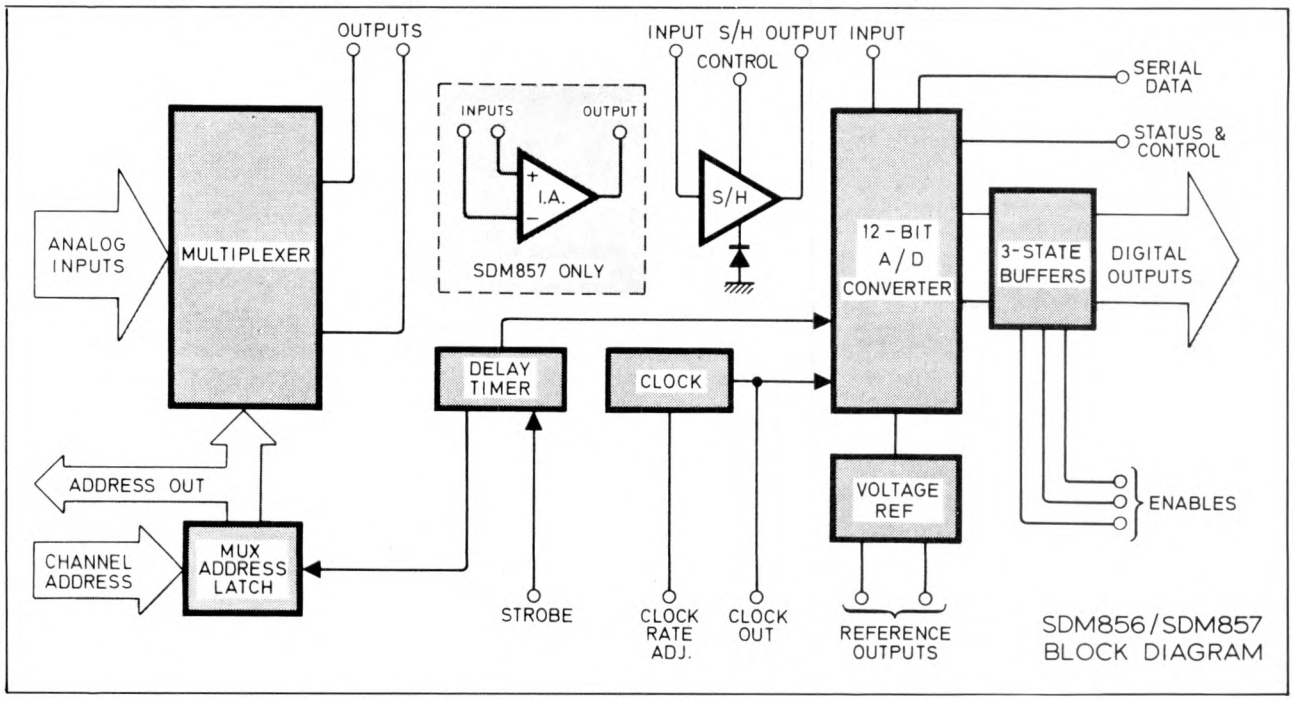
A Miniature Hybrid Data Acquisition System from Burr-Brown

Burr-Brown have just introduced a complete data acquisition system with eight differential inputs (16 single ended) and 12-bit resolution within the confines of a tiny package measuring only 55 x 43 x 5.6 mm (2.2 x 1.7 x 0.22 ins).

The unit comprises a 16-way input multiplexer with channel selection latches; sample and hold circuitry; a 12-bit analogue-to-digital converter complete with voltage reference, clock and timer; and three-state output buffer stage.

The unit is also available with an integral high-grade instrumentation amplifier. The inputs and outputs of the various sections of the unit are brought out to package pins, providing the designer with a great deal of flexibility in the way that the unit is configured. *(Diagram below)*

Contact:
Roger Isaacson,
Burr-Brown International Ltd.,
17 Exchange Road,
Watford, Herts WD1 7EB
Telephone: (0923) 33837



THE MICRO IS A MANY SPLENDOURED THING

LESLIE SOLOMON, Technical Director, Popular Electronics

It has now been 3½ years since the introduction of the first personal computer kit. Since that time, we have seen something like 300-400 computer stores, half a dozen computer magazines, and several hundred computer clubs spring up in the U.S.A. At this moment in time, it does appear that there is almost no end in sight. Many other countries are also getting involved in the computer craze, at a steadily increasing rate.

Many computer types are being accosted by local radio stations, newspapers, and TV stations, all asking pretty much the same question... "what can you do with a computer?". This longish letter is a partial reply to that question.

Of course, the first reply would be an intricate discussion of "number crunching" or text manipulation – the sort of things that most non-computer people consider the domain of the high-speed digital computer. Years of exposure to movie and TV programs have taught the general public that digital computers are massive collections of lights (usually flashing in mysterious patterns), large metal cabinets filled with complex outer-space type of things, huge noisy tape machines spinning madly, and large and noisy printers that clatter out all kinds of strange messages. Most messages are confined to either ruling the world, or solving almost impossible problems. Even today, people still see these large machines attended by mysterious figures dressed in white smocks, who seem to spend their time churning out mountains of data whose main aim is to confound the public with wrong bills!

However, with the home or personal computer, another world has opened up. The introduction of microprocessors and solid-state RAM and ROM have resulted in innocuous computers that do not look the least deadly. In fact, most look like gentle beasts, despite the fact that a personal computer of today is more powerful than the large beasties of only a decade ago. Consider the fact that a typical hobby computer is probably just as powerful as the IBM 360-Mod S, the business and scientific workhorse of yesterday.

Besides an electronic evolution, personal computers are also evolving new industry (we call them "cottage industries") where electronic and software people can now get involved for both love and profit. Many advanced hobbyists (entrepreneurial types) are moving into small-business systems where a single computer, a disc system, and a couple of terminals are all that are needed to service a number of shops with payroll, inventory, and business tax forms. These hobbyists either worked for a hardware or software company and have now elected to "do their own thing" on their own computers. These operations are springing up all over the US.

Probably the biggest use for personal computers (which really means private computers) is in various forms of "games". The general public has long been aware of the arcade games that range from ping-pong to "shoot 'em up" games involving planes, ships, subs, flying saucers, etc. Much money has been poured into slots since their introduction. This, of course, has led to the introduction of home video games that do almost the same thing.

Since most of these games are dedicated (can do only one thing), many computer enthusiasts have programmed their own computers to play more complex, interactive games on their systems. Some of these games, especially the well thought out ones, can be quite a challenge to the human operator. We already have games in one human group (a club or university) play a complex game against another human group – sort of one computer playing against another. These battles of software and strategy have been known to last several months without a clear winner.

Recently, chess programs have become the rage. Many of these programs can play almost at the grandmaster level, thus confounding most chess players who hate being beaten by a small piece of silicon and some programmer someplace. At the moment, there are a number of chess programs going on, where one machine and its program are "playing" another machine and its program. Chess clubs are being formed, and already, some chess programs are beginning to become famous in computer circles.

Having a computer play chess is nice, but many of us have not forgotten the children, and many programs exist that play interactively with a child. The results of this can be seen at almost every computer fair. The largest group sitting at consoles is invariably young – and playing computer games. That is where our future lies; with our children.

Since color (as in color TV) can be digitally derived, and computers can be programmed to create graphics, another dimension has been added to the personal computer. Not only games, but many programs for children have been developed to take advantage of color. Most color systems today use the conventional raster-scan technique, although some stroke systems are coming along. There are presently, quite a number of high-resolution graphic add ons available at computer stores, some with color capability. Some of these can accept an input from a conventional B/W TV camera and digitally produce some excellent color. Light pens are coming into use, so there we have the paintbrush. All we need now are the artists.

Probably the most interesting uses for the home computer are in an area that I call "non-computer" uses of the computer. In these uses, we must consider the computer as a "magic box" that accepts an input, does things to that input, then produces the desired output (usually on a CRT monitor). This concept is like using a TV set, you do not have to know how the TV set works in order to use it. Simply turn it on, select a channel, and the magic box follows your orders. The same applies to a car – just turn the ignition key to get it started, find a gear you like, and go. Everything is magic. And, that's the way it happens with a computer, You enter the program, hit the start switch, and things happen (hopefully correct).

The first non-computer use is one of speech input. *Popular Electronics,* back in May of 1977, introduced the Speechlab, an analog-to-digital converter that can accept an audible input, convert it to a six-bit digital word, then store that word in memory. The next time that word was spoken to the computer, the audio input was once again digitized, and the memory searched for a similar "template". Once found, the program caused the computer to do what it was supposed to do when it heard that word. In some uses, this approach is used to store words in an inventory control, where the user has a microphone around the neck, and simply talks the inventory into the computer.

Once we had a way to talk to the computer, we had to find some way to use the data. In December of 1977, *Popular Electronics* introduced the AC Controller – in which a digital computer could communicate over *existing* ac power lines within a house or factory. This allows a computer to control many functions without having to re-wire the complex. Now, if you have the right program, a computer can control dozens of electrical appliances via the ac lines. Since the computer could "talk" to the power sockets, the sockets could also "talk" to the computer. This means that a single computer in the basement could be accessed by terminals on other floors without running any wires. The possibilities of this approach are open ended.

Obviously, the next thing was to combine these two elements. Now we could speak to our computer and have it perform certain actions – some quite remote from the computer. My own machine turns on the lab lights when I enter the room and talk to it, turns on various peripherals, and even controls the TV and radio. It does get a little science-fictiony at times – but it is fun.

We have also been successful in interfacing (and programming) a D/A converter (equipped with audio filters), so that the computer could "speak" after a fashion. A little robotlike, but still understandable speech. Since the D/A conversion is phoneme oriented, the computer could say anything that we could digitally encode, and not be tied down to a few basic words.

Now, what can we do with this exotic setup? Well, besides having fun, we are quite serious about using this system to help physically and vocally handicapped people communicate and interact with the world.

Since Speechlab accepts an *audio* input that does not have to be in any recognized language, why not teach it to accept the various repeatable sounds that many handicapped people can generate? And, in fact, that is what we have done. These sounds (must be repeatable as this is the 'language' of the handicapped person) are translated by the computer either into English text that can be displayed on a CRT monitor or, using the controller, turn things on and off (open doors, windows, turn on lights, etc.). We have been reasonably successful in doing this. At present, there are a couple of institutions experimenting with this unique approach that could only have come with the introduction of the personal computer.

One simple byproduct of this experiment is the use of a computer terminal and modem that allows deaf people to communicate over conventional telephone lines.

At the moment, there are several computer hobby groups that are quite deeply involved in the use of computers to assist the physically handicapped.

Then there is the area of music. Not the usual synthesizer type of music that is becoming quite common, but a new form in which a person can sit down at a computer terminal and without any knowledge of the computer (there is that 'magic box' again), and even with a limited knowledge of music, can create some fascinating sounds. A special high-level language, called appropriately enough "Music" is self explanatory when it turns up on the CRT monitor. You can score existing melodies, or create your own sounds. All musical requirements are met.

On the video display, it is easy to specify the note (frequency), duration, key, and all the other musical parameters. A simple coding scheme is used. The computer generates its own tones from its internal square waves and all you need (besides the computer) is an audio amplifier and speaker. You can use conventional electronic organ filtering to create various sounds. Now, for the first time, music students, teachers, etc., can have an electronic string quartet at their bidding, day or night, to play their tunes. The advantage of the digital approach over conventional analog tape recording, is that in the computer program, you can make any changes you desire – such as pitch, note duration, speed, etc., as easy as touching a few keys. Several schools are using this approach to give their music students practical experience in writing and scoring music.

The latest non-computer use is in controlling various types of robots. These range from strange little boxes on wheels (like R2D2 in *Star Wars*) through complex things that look quite dangerous to approach. Most of these "robots" main claim to fame is that they move around and avoid obstacles. Not very exciting. With the introduction of CMOS logic (the 1802 processor for example) it now becomes feasible to build quite a sophisticated robot controller; so on to higher peaks.....

A couple of friends and myself are busy at work with our own concept of a robot. This machine is designed to perform useful tasks – such as pushing a vacuum cleaner around the house, or a lawnmower around the yard without chopping down trees and bushes. We took a look at the various wheeled and tracked things running around and decided that the designs left a lot to be desired. Most of these creeper-crawlers stumbled over stones, shorted out when traversing wet areas, and tended to fall down when they met steep little ridges or large cracks. Our approach leans toward the "spider" in which six long legs (three on a side) are made to move the electronic package (which hangs at the legs-junction like a spider body). So far, the radio-controlled model works fine. It looks like a strange thing from another planet, but it is our baby!

Instead of controlling the robot through its built-in computer, we are going to use the internal computer to perform 'housekeeping' – that is, monitor the battery, make sure that the machine is always level, and control the memory. We intend to use a *host* computer having both large memory, and a disc system to actually do the "thinking" for the robot, then transmit the data to the robot via an ultra-violet optical link. In this way, we are going to treat the robot as a highspeed (about 9600 baud) terminal. The robot will have a pair of "antenna" like a grasshopper, except at the end of one antenna is an IR receiver, and the other containing the companion IR transmitter.

We have tested a small battery-powered BW TV camera with a small rf device, so that we can "see" where the robot is at all times. Now, if we put a vocal interface onthe robot, and a Speechlab.....

SOMETHING BIT ME

Chris Howland's *Personal Experience*

"But what can it do?" Colin asked.

"What do you mean 'what can it do?' " I replied gruffly as I eased the keyboard out of its tightly packed box. "It's a computer – it can do anything!"

"If you ask me," Colin said with a wry smile, "you've gone and bought yourself a white elephant. What do you want a computer for if you don't know what it can do?"

I had an awful feeling that he was right but wasn't going to let him know it.

"Would you mind leaving me alone until I've got this thing connected," I said angrily. "Go and pour yourself a drink or something – or just go," I added, "I'm not particularly fussy at the moment!"

He went and joined my wife on the veranda and I heard laughter coming through my study window – laughter at my expense, obviously.

You see, I have a problem which has been bugging me for years and it has finally come to a head.

It must have started at school. As a pupil, all my systems were reasonably "go" except for maths and that's where I drew a complete blank. I had absolutely no head for figures; and even to this day, I count on my fingers.

I discovered the abacus in Bangkok and we became immediate friends. It wasn't long before I started looking around for things to add up or subtract and I soon realised that an imperfect mind can be greatly complemented by a machine however primitive. This, I think, was the beginning of my mania.

I bought an adding machine – but one which could multiply and divide as well. It weighed a ton and cost me a fortune in canvas bags because the handles kept on ripping off! We had great fun together, this machine and I, and we went everywhere together. When there was nothing to calculate, I took to adding up telephone numbers and as for division, this was a wow! My favourite was to ask it to divide 0.01 by 7.123! Once I had entered the numbers and pressed the "go" button, all hell would be let loose. The whirring and clanking of metal levers would go on for anything up to a minute as the poor thing wrestled with its useless problem. Then would come a decisive triple "crunch" and a little slip of paper would shoot out bearing what I assumed to be the right answer. I really punished that poor thing – punished it in the same way as I was punished at school.

It was about this same time that I first clapped eyes on a transistorised adding machine with a huge LED display. It was out of this world – but so was the price, so that little love affair came to an end before it started.

Then people began sending rockets to the moon and everything started getting smaller. I snapped up my first pocket calculator (it needed quite a big pocket!) in Spain of all places and when the digital wrist-watch appeared, I immediately made another investment. I still wear this space-age antique to this day even though it gains 20 seconds a month and cannot cope with Leap Years.

The whole thing began to gather momentum a couple of years ago when I acquired a fascinating little time computer with four programmable alarms. Then came a calculator with a memory and afterwards another one with a print-out. These were later augmented by a strange little device which is very flat and squeaks every time you press a key. At first I thought that there was something wrong with it but I was assured that it's meant to be that way so I now occasionally calculate to music. This one has the added attraction of sending the dog up the wall.

The crunch came at the end of last year. A friend of mine living in Brussels sent me a pamphlet about Radio Shack's TRS-80. The rest you already know.

Using a certain amount of leverage (I'm quite well known on TV over here and this can be useful at times) I managed to get myself what is probably one of the first TRS-80's in Germany and was just unpacking it when Colin came in.

There are four main parts: the keyboard which also contains all the wizardry, a mains unit, a video and a cassette recorder. There is also a fantastic – and I repeat – fantastic book of instructions which takes you step by step through this apparently logical jungle. If I can understand it, anyone can and I'm not kidding!

Colin came in again. "How are things going,

Einstein?" he asked. I ignored him and concentrated upon my equipment which was definitely not functioning properly.

"What's the matter?" Colin asked.

"I've got everything turned on but the video won't light up" I replied.

Colin took a closer look. "Try plugging it in," he said and lit up a cigarette.

That was lesson number one. If you're going to muck around with computers you must learn to have eyes in the back of your head. As far as the TRS-80 is concerned, each interface has it's own power supply so I found the cable, hooked up the video and the screen began to glow. I then switched on the computer and my heart jumped as the magic word "READY" appeared on the screen.

"Ready for what?" enquired Colin.

"Let me look through the book first," I pleaded and began turning the pages. "Ah!" I said triumphantly, "it can print my name!"

"Then let's see it," said Colin.

Very slowly I started to type my name and marvelled as the letters appeared on the screen. As I said earlier, I've worked on TV all my life but never like this. But I must have done something wrong because Colin suddenly said: "Since when is your name What?"

I looked and he was right; the word "WHAT?" was now in the top left-hand corner with "READY" underneath it.

I can't quite remember what happened after that. I know that Colin left rather suddenly and my wife went to bed without saying goodnight.

Something changed that evening. I discovered a completely new world with a brand new language with exotic words like CLOAD, GOSUB, CSAVE, IF-THEN and RND(N)! And it wasn't long before I began waking up at night murmuring "FOR X=1T05000:N.X:CLS" and other little 20th Century endearments. But it was the 12 hour clock which nearly undid me. I spent hours working out the programme on a train but when I ran it in the computer the TRS-80 arrogantly asked "HOW?". I nearly threw the whole damned shoot out of the window – power pack and all!

But I'm bitten – in fact, I've got bytes all over me. Now I know what a computer can do and my only problem is to learn how to instruct it. Talk about the blind leading the blind. But I won't stop – I can't in fact. Never in my life have I been forced to be so exact; never before did I realise that just one silly little glass of wine can blow the whole works. If you want to give up drinking, start computing! Nevertheless, I sometimes feel lonely because nobody wants to hear about my binary escapades.

Looking back, however, I realise that I'm not the only nut in the world. I had a friend who, like me, loved gadgets. One day he turned up and proudly showed me his latest acquisition. It was a solid gold watch which must have set him back at least £3,000! It had everything on it – the phases of the moon, the date and another little device which enabled you to determine the height of a building. It was quite simple. All you had to do was to drop a stone and immediately stop the watch. When the stone hit the ground, you stopped the watch again and read off the height of the building on the dial.

My friend took me to Cologne's newest bridge over the Rhine (this was back in 1954) and stood there with his stone and his fancy watch.

"Now we'll check how high this bridge really is," he said seriously.

He then dropped the watch and stopped the stone!

I wonder what Colin would have said about that.

MATROX FROM SHELTON

			PRICE (1 off)
1.	ALT-256**2E	256 x 256 S100 graphics card	£284.00
2.	ALT-2480E	24 lines of 80 characters S100	£213.00
3.	MTX-816	Big characters 8 rows 16 characters per line	£128.00
4.	MTX-1632	Very clear characters 32 characters 16 lines (SL version can be synchronised to TV picture)	£162.00
5.	MTX-A1/MTX-B1	Keyboard scanners and LED driver Single chips direct connection to any CPU bus	£28.00

SHELTON INSTRUMENTS LTD.,
22/24 Copenhagen Street, London N1 0JD Tel: 01-278 6273

John Coll

PILOT FLIES ACROSS THE ATLANTIC

I never cease to be amazed at the frequency of new and exciting developments in the micro-computer field. Last week a friend brought over a new language, sat me down in front of my computer and told me to play with it. Well, this has happened often enough before and it usually takes about 2 or 3 minutes to 'crash' the 'uncrashable' system so I approached it with the usual determination!

"What is your name?", the computer asked me. Well I wasn't feeling too responsive so I replied, on the keyboard, "GO TO HELL". I must admit that I was just a little taken aback when my previously well trained computer replied "Ah yes, the abode of your ancestors, perhaps some other time." As you will see by some of the output printed later on the computer was well able to keep up with my comments.

So this is PILOT which stands for Programmed Inquiry, Learnings Or Teachings. It is a languge developed in the States for Computer Aided Instruction or CAI as some people like to call it. BASIC is a language geared to handling numbers in an interactive way. PILOT is geared to handling text in a similar easy interactive way. In PILOT you can write a program which will talk to the user and accept replies and then branch off in various directions depending on the reply. In the same way that you *can* handle text in BASIC, so you *can* handle arithmetic in PILOT, but if the bulk of your interaction with the user is in words then PILOT is the language to use.

For example it is very easy (in PILOT) for the computer to recognise any of the following as the correct reply to the question "What is 15% of 20" Answer – 3 or three or Three or THREE or I think it's probably 3 or 3.0000. The real beauty is that the machine can very easily be made to give a sensible response to "I don't know" or even to minor mis-spellings like "THRE".

So lets have a look at some of the commands available in PILOT. All commands have to start as the first letter on a line and all commands end with a colon. The basic commands are:

```
R:    remark
T:    type text
:     type text continuation
A:    accept answer
M:    match answer
J:    jump
U:    use subroutine
E:    end (subroutine or program)
C:    compute a value or edit a string
XI:   execute indirect
FI:   file input
FO:   file output
D:    dimension
```

Certain "modifiers" and "conditioners" can be added to the basic commands as the following section of program will illustrate.

```
T:    What is your name
A:
M:    john
TY:   John is a very common name
TN:   I am glad it's not John
```

In the above example the "conditioners" Y and N are added to the T (type) command so that whether the line is typed or not will depend on the result of the last M (match) command. Notice that the accept command need have no variables associated with it. If there is a variable then that variable will be assigned the value that the user gives. For example:

```
A:    $A$
```

would cause the answer from the user to be put into the string A$. One particularly nice feature is the way the language deals with the following:

```
T:    What is 3 times 4
A:    #A
```

The user's answer is scanned for a number and if one is found than A is assigned the number. So a reply like "I think it's 12" will be quite acceptable. If no number is supplied by the user then a, testable, error condition is set – but the program continues.

Now that we have met some of the commands the best way of showing you how PILOT works will be to examine a short program:

```
*START
PR:                    LSG
D:                     A$(20)
T:                     What is your name
*INPUT A:              $A$
T(LEN(A$)=0):          Surely you have SOME name.
                       What is it?
JC:                    @A
M:                     i&call!call&me!name2is
TY:                    Sorry, but could you just type
                       your name
JY:                    @A
M:                     mr.!mrs!ms!sir!dr
TY:                    We have no need for such
                       formality. What do your friends
:                      call you?
JY:                    @A
M:                     1!2!3!4!5!6!7!8!9!0!!$(!)!+!*!;
                       !:!<!>!?!/!
MN:                    nut!ball!sh*t!hell!dam!s*x!
                       m*ck!scr*m!
MN:                    jesu!god
JN(LEN(A$)<15):        GOTNAM
T:                     I can't call you that! Come on,
                       what is your name?
J:                     @A
R:                     So now we have got a name
*GOTNAM
```

```
C:          A$C
T:          Well $A$ what subject do you
            want to do today?
A:
MJ:         math!arith!alg!geom!calc!trig!

J:          MATHS
MJ:         geog
J:          GEOG
MJ:         french!span!ital!germa!
J:          MODLAN
```

The "*" in the first column indicated that START is a label. "PR." stands for Problem Instruction and is used to mark the start of a section of a program. One can associate a number of options with the PR command as follows:

U	converts all input to upper case
L	converts all input to lower case
S	removes all spaces, multiple spaces are always reduced to single spaces
G	permits the user to use GOTOs during a program run
W	clears the present label table permitting reuse of old labels
E	permits the user to use the escape command within the program

The "L" and "S" options make it very easy to match input since all items in the match list can be in lower case and spaces will not confuse the matchings.

The "T(LEN(A$)=0)" command will cause the line to be printed only if the user has failed to enter anything before pressing RETURN, and the next line "JC: @A" will jump to the previous Accept statement only if the last testable condition (LEN(A$)=0) is true. Next an attempt is made to match the user input with mr. mrs. etc. and then with numbers and various other phrases. You may have to get some help in understanding these of course!

Just below the label "*GOTNAM" is the command C: A$ C which means turn the first letter of A$ into a capital letter – if possible. It is so easy in PILOT.

This is not meant to be a full PILOT manual so I haven't dealt with the special match characters, the jump statements, the modifiers (which for example suppress a carriage return/line feed). However I had better mention the mathematical functions that *are* supported in case you are left with the impression that it can't do any maths. The functions available include operations on full floating point variables and arrays and on strings. In addition to the simple mathematical functions such as add and subtract ABS FIX INT SGN RND SIN COS ATN SQR EXP LOG LN STR FLO ASC CHR LEN and INS are supported.

It is interesting that in the version that I have been using all the arithmetic is done by a National Semiconductor calculator chip – which is why it is a bit slow. It is fine for simple calculations but no good for a hundred square roots between Type statements. However when the fast AM 9511 Arithmetic Processing Unit is used the speed will be very acceptable even for quite complex arithmetic.

So what does one need to run PILOT? Well in single user mode on a SWTPC M6800 system one needs just the software and the calculator board. To work multi-user PILOT (and it does work!) one also needs the Multi-user board. So that is the M6800 system.

PILOT is not available in the U.K. for any other system at the moment but could be very easily. Software has been written for both 8080 and Z80 machines as well as a number of mini-computers such as DEC machines. My colleague Philip Couzens is in the process of obtaining material from the States and will be glad to pass information on to those who are interested. He can be contacted at

The U.K. PILOT User's Group
Oundle School,
Oundle,
Peterborough
England.

The sample output that follows illustrates the extreme flexibility that the user has in using a well constructed PILOT program. I think PILOT will have a major impact on computer assisted learning in the U.K. as well as providing an excellent vehicle for interactive games. Of course it is very young and there is not the wealth of material that there is for BASIC – but how often do you use someone else's BASIC program, not very often I expect. PILOT will flourish all the better if it is well supported over here as it is in the States, for example People's Computers publish at least on PILOT program each issue. I hope we see the same sort of developments over here. Try it – you will enjoy it.

Bibliography:

a) PILOT "People's Computer Company" Vol 5 No 3
b) PILOT "Creative Computing" Vol 3 No 3
c) 8080 PILOT "Dr. Dobb's Journal" Vol 2 Nos 4 and 5 a complete source listing of PILOT for the 8080
d) Tiny PILOT "Kilobaud" March/April 1978 flow charts for a version of PILOT
e) Common PILOT manual, South West Technical Products full manual for an M6800 PILOT


```
FILE=$AMPLE
Pick a number between 1 and 5.
3
    In this program you are going to see how the
computer can be used to teach. Teaching programs
consist of long and complex combinations of simple
operations, some of which all computer languages
handle easily, but many of which only languages
designed for teaching handle easily. For example,
all languages easily handle what i'm doing now,
typing text. Not all languages handle the following
few lines with the same ease.

Please type in your full name using all uppercase.
letters:
JOHN A COLL
Now we can be on a first name basis, John.

    Probably the most important feature of a
computer  language for teaching is its ability to
recognize a variety of student responses. The
language I'm using now is called COMMON PILOT; I'm
using it in part because of its powerful answer
processing capabilities. Push RETURN when you're
ready to see them in action.

Thanks John, I needed that to clear my screen.


         A. Numerical Answer Processing

    Sometimes we want the student to give an
exact numerical answer. We'll try one like that.

How many years is it between presidential
elections in the United States?
4
That's right; it is always 4 years between
presidential elections.
Odds are that you answered with the single digit
4; and any decent computer language could recognize
that. To give COMMON PILOT a chance to show off a
bit, this time imbed the digit in a sentence, like
"Presidential elections are held every 4 years."
How many years is it between presidential
elections in the United States?
ELECTIONS ARE HELD EVERY FOUR YEARS
I can recognize the words for the digits, but it's
a bother
Please give me a digit.
4
That's right; it is always 4 years between
presidential elections.
OK, John, hit RETURN, and I'll clear the screen.

    Many times we don't need an exact
numerical answer, just one that is close enough. To
illustrate I'll accept any answer to the next
question which is within 10% of the correct value.

   What is the approximate road distance between
Chicago and New Orleans?
1200
Farther than that; guess again.
1400
That's within 10%. Hit RETURN.

         B. Textual Answer Processing

Not all questions can be answered with numbers, and I don't
want to be restricted to multiple choice questions. Therefore
I'm going to have to recognize words too. Sometimes I want a
particular word answer spelled exactly right. For example, in
setting the table for a meal the knife should be placed on
the RIGHT side of the plate.


Of course.


     Other times I'm willing to be a bit
flexible about the spelling of words. That is, I
must be able to make the computer recognize a
variety of misspellings. This is difficult to program
in most computer languages but easy in Common Pilot.
 To illustrate, the capitol of the state of Washington
is Olympia. Give me the name of the capitol of Washington,
but misspell it.
OLYMPIA
Sorry, that won't do here; it's spelled
correctly. You can do  worse than that! Try again.
ULIMPYA
That I would recognize as meaning Olympia.
Do you want to try again?
OOL  I MEAN YES
```

To illustrate, the capitol of the state of Washington is Olympia. Give me the name of the capitol of Washington, but misspell it.
OOL EARLS COURT
That was too atrocious for recognition.
Do you want to try again?
YES

To illustrate, the capitol of the state of Washington is Olympia. Give me the name of the capitol of Washington, but misspell it.
OLIMPIA
That I would recognize as meaning Olympia.
Do you want to try again?
NO
I also must be able to recognize alternative answers, so that the student need not use a particular restricted vocabulary. To illustrate here, let's find out how our tastes in music agree.
What kind of music do you like?
PUNK
Well John, we don't agree on that one.
Do you want to try another one?
YES
What kind of music do you like?
CLASSICAL
Excellent taste, in my eyes.
Do you want to try another one?
YES
What kind of music do you like?
OPERA
Excellent taste, in my eyes.
Do you want to try another one?
YES
What kind of music do you like?
RHUBARB MUSIC
You've succeeded in stumping me by including one that isn't on my list.
Do you want to try another one?
NO
Fine John, please hit RETURN.

Now what I'm going to do is allow you to sample a number of short sections of instructional programs. There will be 3 each at the elementary and at the intermediate level. You'll be given a chance to repeat most examples at the end of the example. I realize that you may not be interested in doing all of them, so before I lose you, I'll give you the commercial. If you want to learn more about COMMON PILOT, you can obtain a manual from

Micropi
2445 N. Nugent
Lummi Island, Wa. 98262

You can press the RETURN after you've written that down.

MICROPI READY
UNIT=1
FILE=HORMUZ

THIS SAMPLE PILOT PROGRAM DOES NOT MAKE FULL USE OF THE COMPUTATIONAL FEATURES OF MICROPI PILOT BUT IT DOES DEMONSTRATE HOW A CAI PROGRAM CAN TAKE ON A PERSONALITY EVEN TO THE POINT OF RESPONDING TO BACK-TALK. THE USER IS ENCOURAGED TO TRY WRONG ANSWERS TO GET A FEELING FOR THE POTENTIAL FOR VERY CONVERSATIONAL CAI.
PLEASE PUSH RETURN TO BEGIN...

S A L A A M !!!

May Allah in his all-knowing wisdom bring happiness upon your many days. I, a most unworthy soul, am a scribe . . . a solver of problems . . . a composer of words and letters. A thousand pardons, young person, but are you male or female?

NEITHER
Young one, it may have been your fate to be neutered so early in life, but once you were either male or female. Which was it?
YES
My sympathies, young person, for I can readily understand the problem facing you. One can decidedly not go by outward appearances these days. My suggestion is this: I have noticed the generous occurance of segregated resting and washing rooms in this area.
They are generally of two varieties:

Type A, labelled: Men, Gents, Male, Hombre, or Messieur
Type B, labelled: Women, Ladies, Female, Senora, or Mesdames.

If to date you have been inclined to use Type A you can in all safety say you are male, and, if using Type B has been 'your thing' (I am learning your language!), then say you are female.
LES
Young one, sex aberrations do not enter into the question of whether you are male or female. Either ask for help, or, if that does not work (to quote the learned British programmer) "bye".

FEMALE

A thousand curses upon my miserable soul, that I should fail to recognize so fair a desert flower. Come, let us move to the shade of this date palm, where the light is easier on my weary eyes. . .That's right, sit there . . . by my feet.

Now, O Small Wind of the Night, by what name are you known in your father's tent?

SUSAN

Susan, did you say? Now I remember (forgive an old man his slow memory.) You are the one who is to become my apprentice. Tell me, what do you know of problem solving?
NOT A LOT
You underestimate, surely.
Have you ever heard of the merchants of Baghdad and the brigands of Hormuz?

NO I DONT THINK SO

Well, it appears that . . . . . .
In the days of your father's father's father, a group of merchants, (each merchant with his camel and his three slaves), was attacked in this very oasis by a group of Isfaghan brigands. In the confusion, a small number of slaves fled into the desert, each on a camel. The brigands made off with two camels and six slaves for each brigand, and the merchants were left with one slave for every two merchants and of course no camels at all.
Now, to return . . .

Do not mutter! It is no shame; though the story is common, many have not heard it. It seems that,
In the days of your father's father's father, a group of merchants, (each merchant with his camel and his three slaves), was attacked in this very oasis by a group of Isfaghan brigands. In the confusion, a small number of slaves fled into the desert, each on a camel. The brigands made off with two camels and six slaves for each brigand, and the merchants were left with one slave for every two merchants and of course no camels at all.
Now, Susan, the problem is to determine the smallest number of Brigands who could have taken part in this (Allah curse my tongue for calling it such) adventure.
Do you think you can solve this problem by yourself?
I DOUBT IT

Then, with my inadequate knowledge and the grace of our forefathers, I will try to bring light upon the dark recesses of your being, and teach you the art (truly a fine art) of problem solving. Let us first look upon the formation of the events leading up to the skirmish.
How many groups have we to begin with?
I DONT KNOW
You do not see my meaning, I fear. We have here the merchant group, encamped by an oasis well, and we have there a group of brigands, lurking in the night dunes. Which makes how many groups?
I THINK THERE ARE TWO
Excellent--
and how many groups do we have after the skirmish?
3
Excellent, my friend!

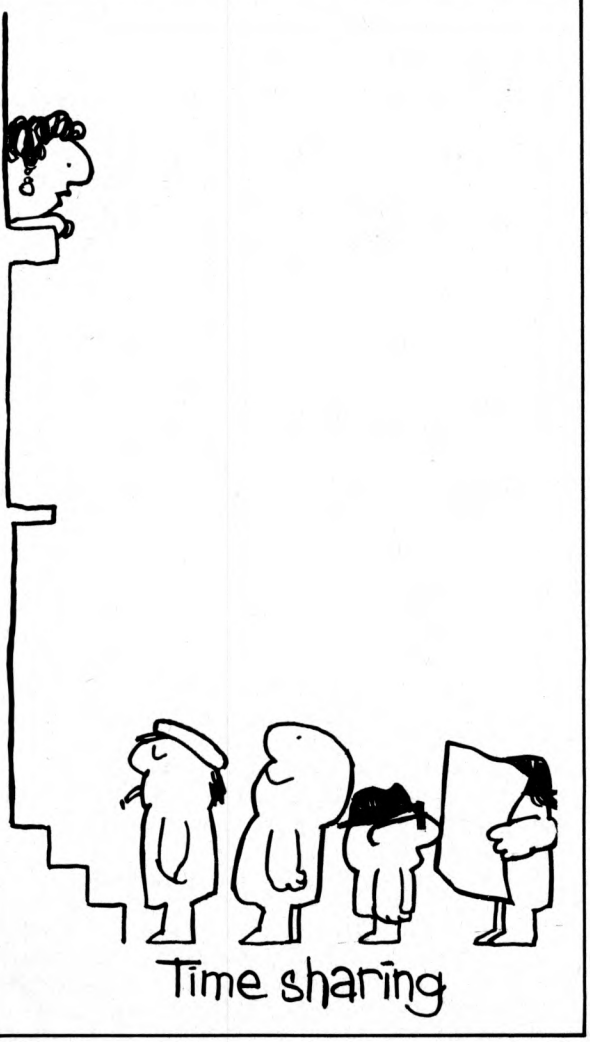
So, this gives us before the battle, two groups:

brigands	merchants
	camels
	slaves

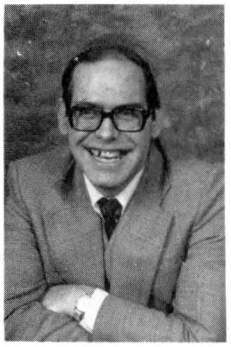
and after the battle three groups:

brigands	merchants	slaves
slaves	slaves	camels
camels		

Now, suppose we were to say the brigands are b in number, and the merchants m in number. How many camels do we start with, if we assume the merchants have the only camels invloved?
LOTS
No. If there are m merchants, each with a camel, how many camels?
3M
No. Each merchant had only one camel, so how many camels are there, we don't know the ACTUAL number.
M
Excellent, Susan.
And how many slaves did the merchants have?



Computing in the Primary School



Derrick R. Daines, Deputy Head of Carsic Primary School, Sutton-in-Ashfield.

My class of 9, 10 and 11-year-olds were widly excited as they lined up for their first-ever personal contact with a computer.

"I'm going to ask it what team it supports!" one boy said, while a little girl asked, "If I ask it how tall I am, will it tell me?" Then there was the fat girl who declared roundly, "I'm going to tell it to bring me sausage and beans!"

Fear was also evident on several faces and some hanging-back. There was the little boy who – although fascinated – always ensured that someone else was between him and the terminal, and who for weeks had to be encouraged, "It's alright – if you get it wrong, there's isn't a hand that will come out and grab you by the throat, you know."

The reactions both amused and dismayed me. If these were indicators of the impressions that children have of computers, then we all – and particularly the writers of fiction for the mass media – have much to answer for, and the sooner we start correcting it, the better. It is no wonder that most people regard the computer with awe and reverence. One has only to say that some data or other has been 'computerised' for it to be accepted without question. In authority, the computer ranks higher than God.

An illustration of this – both amusing and pathetic – was when I used my computer for our school Gala Day. I had programmed it to select at random three sentences from thirty six, presenting it as a 'Computerised Horoscope.' It was intended for amusement, but a large number of people went away fervently believing in what they had read on the monitor. It is about time that the computer was debunked. The owner of a micro-computer is the person to do it and particularly if he or she is a teacher too.

A few short weeks after their introduction to the machine, an Inspector was in my classroom assessing my work and the childrens' understanding. He asked, "if I ask the computer to do a sum for me, could it do it?" and quick as a flash, a 10-year-old boy shot back, "Is it programmed?" – an answer that well justified the time spent. That boy knew more about computers than the large number of adults who condemned us as 'lazy' because we had a computer in our classroom.

Don't get me wrong – I am not 'knocking' the computer at all. On the contrary, I am a computer nut and put myself into debt to buy the parts and build my computer, which is an SWTPC 6800 with 12k of memory. No, I too have my flights of soaring imagination, drool over bits and pieces, and write programs of no earthly use just for the fun of it, but I do say that we must remove the mystique attached to computers.

In these days when Local Education Authorities are agonising over whether or not to supply a few Secondary Schools and Colleges of Further Education, let me come right out and say that all schools ought to have one – or two – or three. Nor do I expect Infant Schools, for reasons which will become apparent later.

One of the problems of mystique – and of getting LEA'S to supply calculators (let alone computers) is that the very first people to 'latch on' to the possibilities of computers were the mathematicians and then very quickly there sprang up a new discipline – computer studies. The situation now is that more time is spent in school studying the computer than is using the computer as a tool for the benefit of pupils or school! – a situation that surely puts the cart before the horse and adds yet more to the mystique. Worse, it alienates those whom it could best benefit.

Happily, the microprocessor and the micro-computer bid fair to alter all that and it must be shouted from the housetops that – far from being just the tool of mathematicians or the plaything of Computer Studies – they can be of inestimable benefit to all aspects of school life.

An illustration. The most tedious part of my job (I speak personally) is testing the reading ability of children. At the very least, it must be done twice a year and for maximum efficiency ought to be done monthly. We use Schonell Reading Chart and if you have ever listened to a child slowly stumbling through Janet and John or Ladybird you will have some small idea of how boring I find it. The class must be kept quiet (something to be marked later!) while I try to concentrate and keep score of how many words the child reads correctly. Then his/her raw score is taken in conjunction with sex to get a Reading Age, which in turn is compared with birth date. Try as I might, my eyes inevitably and inexorably start closing . . . . .

What a job for the computer! All that is required is a multiple-choice answer: "Which word rhymes with BEAST? Type 1, 2, 3, or 4." The hours and hours of my time it would save!

The sports department: I attended a gymnastics competition recently in which over 200 children took part, each one being given a mark for floor exercise, nother for vault, both counting towards the individual trophies and also for the team trophies (best six in each team, each class.) Apart from the judges, 6 runners were involved, taking marks to a team of 6 recorders and an announcer – myself. At the end of the compeition a long hiatus ensued while the recorders battled with calculators and bits of paper to produce the first six in each class and the best teams. Apart from totally eliminating this delay – and the possibility of human error – the computer would have coped with the continuous stream of enquiries such as, "Please sir, how many marks did I get for my exercise?" and "Who's winning up to now?" Not only that, but it would have released the 6 recorders for other duties, and let everyone get home earlier.

The school secretary would surely benefit from a line to a computer. In large schools – and more especially those on split sites – it is not merely difficult to find a

particular person, it is virtually impossible. With a complete timetable as data base and a real-time interface, the problem is solved. Instantly. Or consider those fantastic questions of administration that constantly crop up – "how many red-haired children have we got that ride bikes?" Impossible question? Couldn't happen? It does! All the time (That one, for example, came from the police.) Such problems are relatively simple to solve if the school is small, but nowadays schools of well over 1000 pupils are not at all uncommon, and a computer is rapidly becoming an urgent necessity. If our customers were adults instead of child pupils, we surely would have had computers years ago. Considered as a business, the inefficiency of our schools is staggering.

English, Sport, Secretary – totally divorced from Maths and Computer Studies. Any others? Yes – there is very little of school life that would not benefit given the chance; recording marks and calculating averages is only the very first step.

Hiding a teaching point in a game is a technique as old as the hills. I taught my 10-year-olds vectors by playing Startrek; also overcoming the fear of the machine. I taught coding by the use of ASCII. I taught music theory by getting the computer to sing. I taught spelling by playing Hangman. I taught principles of acceleration and gravity by playing Lunar Lander. I taught arithmetic tables with a random-number selector. Why go on? The range of teaching points that can be got over in this way are virtually limitless, bounded only by the imagination of the teacher. (Any subject can be fitted into the format of the multiple-choice question, as the Open University has shown.)

Language development is perhaps a surprising spin-off, but thr Dienes' Logic Blocks have shown the way and I use them extensively. For the benefit of the uninitiated, Professor Dienes' blocks come in sets that differ from each other in various ways – colour, shape. size or thickness – each block being different from every other block in one or more particular. Thus we may have a large red thick triangle, a small thin yellow circle and so on. In the Infant School and early Juniors, work is verbalised and a wide variety of sorting games are played; but as the child gets older Venn diagrams are introduced. All the time language is being used and developed. Perhaps coincidentally, the words (and the accompanying logical thought) that are most in use are very familiar to computer users – AND, OR NOT – and are vital stepping stones in the child's development. Even at ages nine and ten, for example, AND and OR can be confused.

The same logic blocks are used for sorting on branched charts – all the reds along this branch, all the yellow on this – subdividing until each piece finds its' own place. In this way the child learns to recognise similarities and dissimilarities, a technique which is at the very root of the scientific method.

Pathways with branching gates are the next step, the gates being labelled as to their function, and the child is encouraged to make a careful note of the effect of changing the functions. It is tolerably obvious that this work leads easily and naturally to flowcharts and also to what at a more sophisticated level is called Critical Path Analysis.

Flowcharts, with their two-choice decision boxes, shake hands with other ideas introduced at about nine, the most popular of which is the binary card slection box. Children are enormously intrigued by it and, since it involves physical activity of a sort, will play with it for hours. It does not require much mental effort to see the connection between the binary slection box and regular binary coding and, given the parameters, any of my ten-year-olds will produce binary coding of – say – bicycles, women's fashions, watches or any of the goods in a mail-order catalouge. This is no mean feat, as any computer buff who has attempted coding such material will testify.

Binary arithmetic proper can safely be introduced at eight or nine, initially by means of lights directly wired to switches. The need for a code is soon recognised however, when the children attempt to send brief messages to each other and I steer them off Morse onto the regular ASCII code. This presents very few difficulties and the children are delighted if the computer is programmed very simply to turn their ASCII code into a line of text.

When it is realized that the use of Imperial weights and measures involves arithmetic in a number of different bases, it should come as no surprise to learn that children of 10 can waltz their way through hexadecimal. What will be surprising to many is that they are then ready to try their hand at writing their first programs.

First attempts arise naturally from flowcharts. These have been of the usual introductory type – making a cup of tea, getting a book from the library and so on – and the notion of a stored program has been introduced by reference to the Jacquard loom, etc. We make a brief foray into the use of a matchbox computer and are then ready for the real thing.

Difficulties? Yes, of course there are difficulties, one of which is that all children insist on pressing the key far too firmly! I wince sometimes to see the keyboard flex, although a plastic reinforcing bar has done much to alleviate this. Then there is the agonising slowness involved in finding their way around the standard layout of the keyboard. I make no concessions about this. Since it is the layout that they will have to use later in life, they might as well start right away, although it would have been easy for me to rewire the keys in alphabetical order. However, I have been delighted at the speed with which they have become accustomed to it.

The greatest difficulty might have been foreseen but wasn't. It is that children simply have no idea what they want the machine to do – what to write a program about. So, I impose one. Something very simple of course, like getting the computer to print out the first 10 or 12 elements of the 12 times table. So it is that during their last term in the junior school children of 11 years old have the thrill of seeing the computer print out their very own program. Frankly, I don't know if they are more thrilled than I am, but their faces tell a lot. I know this– they have a flying start on their way to the understanding and manipulation of the modern world; both boys and girls.

I have touched on logic once or twice. Now the world is not too hot on the use of logic right now, as the merest glance at any newspaper will show. The computer however has this over-riding characteristic, that it is completely logical and if your program has any illogicality contained in it, it will inevitably show up. This involves an enormous discipline for the programmer. It might be thought that such a discipline will have a natural and automatic carry-over into everyday life, but studies have shown that this is not necessarily true unless and until the correlation is pointed out. By the very nature of the teaching method outlined briefly in this article, there is ample opportunity for the teacher to point out the correlation at every step – by the use of everyday examples, etc.

It all goes to support my firm belief that after the fun and games are over, after the children have left my care and gone into the larger world of the Secondary School and beyond, some of the logic – some of the careful approach to problems – will have rubbed off onto them and will affect, even if in only a small way, their own approach to the problems of life, hopefully making them that little bit happier. Can anyone ask more?

Submarine Chase

A. J. HARDING

This article describes a computer game written in Tandy Level 1 Basic for the TRS-80 microcomputer. As this dialect of BASIC is a very simple one, the programme may easily be re-written in any version of the language. As listed, it requires about 2.8K bytes plus an undefined (Level 1 does not require dimensions) amount of memory for an array. The optional REM statements are not included in this assessment, nor are any instructions for play which may be desired at the start of the game. However, these may be added because, for reasons of readability, no Level 1 shorthand or multiple statements are used and their incorporation would yield enough space for REM statements and instructions. Readers with a wider version of BASIC will be able to re-write in substantially less space. For instance as Level 1 does not include a square root provision this chore requires seven lines of code (5500 to 6100) rather than the simple statement D = SQR(U), available in most dialects. Consequently, any microcomputer with 4K of RAM, BASIC and a video display will be able to run the game.

No particular skill in programming or originality of conception is claimed. Similar games may already have been written and the author has no great interest in the finer nuances of programming. It is rather good fun to play and it works!

THE GAME

The player is the Captain of a destroyer patrolling an area in which there is an enemy submarine. The area is displayed on the screen. It is divided into 3300 blocks which are designated by the numbers 0 to 110 horizontally and 0 to 30 vertically. He drops depth charges by entering coordinates, such as 55, 15, which would be the centre block of the area. After each depth charge is dropped the computer tells him the bearing of the submarine from the point of the drop. For instance "10 North West" would mean that the submarine is 10 blocks up and to the left from the drop point. The submarine however is moving, so the bearing indicated will be from the point of the last drop. In calculating the next drop therefore, the player must make allowance for the submarine's motion. A bearing of "10 North" would mean that at the time of the drop the submarine was up 10 blocks but a new entry merely calling out that new block would not sink the submarine due to its motion in the meantime. At the start of the game the player is asked to enter a degree of skill at which he wishes to play, being an integer 1, 2 or 3. An entry of "1" – Novice – will cause the submarine to move at a slow speed of one block at a time, "2" – Moderate – at a speed of two blocks per entry and "3" – Expert – at three blocks speed. A near miss will cause the submarine to be disabled and henceforth remain stationary – a sitting duck! The number of blocks proximity to the submarine, which causes disablement, is also controlled by the degree of skill entered. A "1" skill level means that a depth charge within three blocks horizontally and vertically will disable the enemy; "2" two blocks and "3" one block. A direct hit of course, sinks the submarine and the player has won. Every depth charge position is displayed on the screen, and remains there, but the submarine's position of course, is not.

As mentioned, the submarine is moving whilst depth charges are being dropped. If the submarine reaches the border of the area without being hit or disabled, an "escape", it is immediately returned to the area but whilst out it was able to take on more fuel, of which it is short. If the submarine escapes five times then it has taken on board sufficient fuel to escape the area completely and the player has lost. Whether the player wins or loses he may, at the end of the game, enter a "2"

particular person, it is virtually impossible. With a complete timetable as data base and a real-time interface, the problem is solved. Instantly. Or consider those fantastic questions of administration that constantly crop up – "how many red-haired children have we got that ride bikes?" Impossible question? Couldn't happen? It does! All the time (That one, for example, came from the police.) Such problems are relatively simple to solve if the school is small, but nowadays schools of well over 1000 pupils are not at all uncommon, and a computer is rapidly becoming an urgent necessity. If our customers were adults instead of child pupils, we surely would have had computers years ago. Considered as a business, the inefficiency of our schools is staggering.

English, Sport, Secretary – totally divorced from Maths and Computer Studies. Any others? Yes – there is very little of school life that would not benefit given the chance; recording marks and calculating averages is only the very first step.

Hiding a teaching point in a game is a technique as old as the hills. I taught my 10-year-olds vectors by playing Startrek; also overcoming the fear of the machine. I taught coding by the use of ASCII. I taught music theory by getting the computer to sing. I taught spelling by playing Hangman. I taught principles of acceleration and gravity by playing Lunar Lander. I taught arithmetic tables with a random-number selector. Why go on? The range of teaching points that can be got over in this way are virtually limitless, bounded only by the imagination of the teacher. (Any subject can be fitted into the format of the multiple-choice question, as the Open University has shown.)

Language development is perhaps a surprising spin-off, but thr Dienes' Logic Blocks have shown the way and I use them extensively. For the benefit of the uninitiated, Professor Dienes' blocks come in sets that differ from each other in various ways – colour, shape. size or thickness – each block being different from every other block in one or more particular. Thus we may have a large red thick triangle, a small thin yellow circle and so on. In the Infant School and early Juniors, work is verbalised and a wide variety of sorting games are played; but as the child gets older Venn diagrams are introduced. All the time language is being used and developed. Perhaps coincidentally, the words (and the accompanying logical thought) that are most in use are very familiar to computer users – AND, OR NOT – and are vital stepping stones in the child's development. Even at ages nine and ten, for example, AND and OR can be confused.

The same logic blocks are used for sorting on branched charts – all the reds along this branch, all the yellow on this – subdividing until each piece finds its' own place. In this way the child learns to recognise similarities and dissimilarities, a technique which is at the very root of the scientific method.

Pathways with branching gates are the next step, the gates being labelled as to their function, and the child is encouraged to make a careful note of the effect of changing the functions. It is tolerably obvious that this work leads easily and naturally to flowcharts and also to what at a more sophisticated level is called Critical Path Analysis.

Flowcharts, with their two-choice decision boxes, shake hands with other ideas introduced at about nine, the most popular of which is the binary card slection box. Children are enormously intrigued by it and, since it involves physical activity of a sort, will play with it for hours. It does not require much mental effort to see the connection between the binary slection box and regular binary coding and, given the parameters, any of my ten-year-olds will produce binary coding of – say – bicycles, women's fashions, watches or any of the goods in a mail-order catalouge. This is no mean feat, as any computer buff who has attempted coding such material will testify.

Binary arithmetic proper can safely be introduced at eight or nine, initially by means of lights directly wired to switches. The need for a code is soon recognised however, when the children attempt to send brief messages to each other and I steer them off Morse onto the regular ASCII code. This presents very few difficulties and the children are delighted if the computer is programmed very simply to turn their ASCII code into a line of text.

When it is realized that the use of Imperial weights and measures involves arithmetic in a number of different bases, it should come as no surprise to learn that children of 10 can waltz their way through hexadecimal. What will be surprising to many is that they are then ready to try their hand at writing their first programs.

First attempts arise naturally from flowcharts. These have been of the usual introductory type – making a cup of tea, getting a book from the library and so on – and the notion of a stored program has been introduced by reference to the Jacquard loom, etc. We make a brief foray into the use of a matchbox computer and are then ready for the real thing.

Difficulties? Yes, of course there are difficulties, one of which is that all children insist on pressing the key far too firmly! I wince sometimes to see the keyboard flex, although a plastic reinforcing bar has done much to alleviate this. Then there is the agonising slowness involved in finding their way around the standard layout of the keyboard. I make no concessions about this. Since it is the layout that they will have to use later in life, they might as well start right away, although it would have been easy for me to rewire the keys in alphabetical order. However, I have been delighted at the speed with which they have become accustomed to it.

The greatest difficulty might have been foreseen but wasn't. It is that children simply have no idea what they want the machine to do – what to write a program about. So, I impose one. Something very simple of course, like getting the computer to print out the first 10 or 12 elements of the 12 times table. So it is that during their last term in the junior school children of 11 years old have the thrill of seeing the computer print out their very own program. Frankly, I don't know if they are more thrilled than I am, but their faces tell a lot. I know this– they have a flying start on their way to the understanding and manipulation of the modern world; both boys and girls.

I have touched on logic once or twice. Now the world is not too hot on the use of logic right now, as the merest glance at any newspaper will show. The computer however has this over-riding characteristic, that it is completely logical and if your program has any illogicality contained in it, it will inevitably show up. This involves an enormous discipline for the programmer. It might be thought that such a discipline will have a natural and automatic carry-over into everyday life, but studies have shown that this is not necessarily true unless and until the correlation is pointed out. By the very nature of the teaching method outlined briefly in this article, there is ample opportunity for the teacher to point out the correlation at every step – by the use of everyday examples, etc.

It all goes to support my firm belief that after the fun and games are over, after the children have left my care and gone into the larger world of the Secondary School and beyond, some of the logic – some of the careful approach to problems – will have rubbed off onto them and will affect, even if in only a small way, their own approach to the problems of life, hopefully making them that little bit happier. Can anyone ask more?

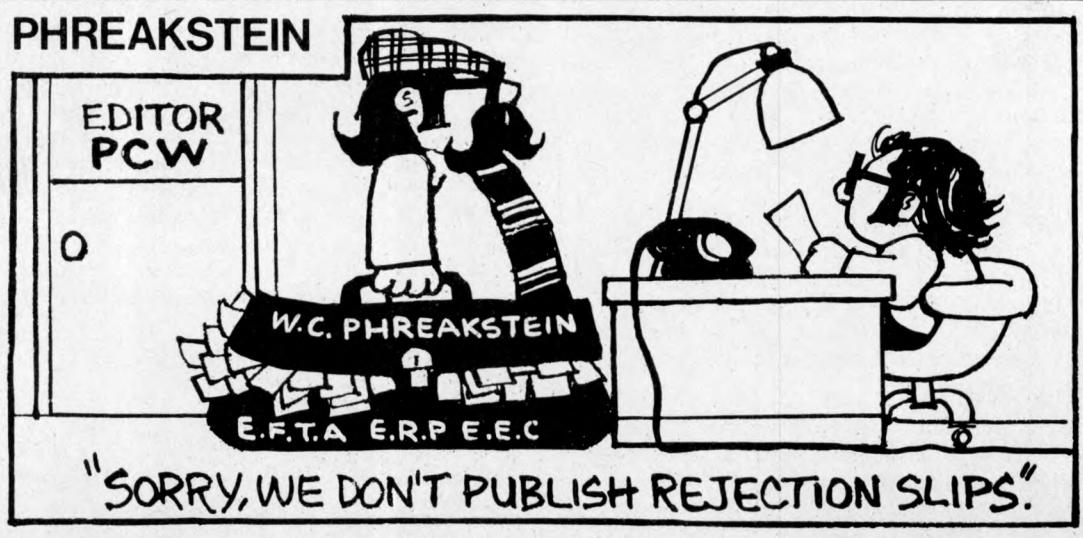
```
5     REM * SUBMARINE CHASE GAME *
10    Z = 0
20    W = 0
30    R = 1
40    A(0) = 2
50    X = RND (110)
60    Y = RND (30)
70    CLS
80    PRINT
90    PRINT "ENTER 1 TO 3 REPRESENTING YOUR DEGREE OF SKILL AS FOLLOWS:"
100   PRINT
110   PRINT TAB (9); "1 = NOVICE : 2 = MODERATE : 3 = EXPERT"
120   INPUT S
130   IF (S >= 1) * (S <= 3) THEN 180
140   PRINT "THE NUMBER YOU ENTERED WAS NOT 1, 2 OR 3.TRY AGAIN."
145   REM *SHORT DELAY*
150   FOR C = 1 to 1000
160   NEXT C
170   GOTO 70
180   ON S GOTO 190, 210, 230
190   E = 3
200   GOTO 250
210   E = 2
220   GOTO 250
230   E = 1
250   CLS
260   F = 61
270   FOR C = 0 TO 30 STEP 6
280   PRINT AT F, C
290   F = F + 128
300   NEXT C
310   A = 115
320   B = 0
330   FOR C = 0 TO 5
340   SET (A,B)
350   B = B + 6
360   NEXT C
370   A = 0
380   B = 34
390   FOR C = 0 to 11
400   SET (A,B)
410   A = A + 10
420   NEXT C
430   PRINT AT 768, "0"
440   F = 771
450   FOR C = 10 TO 110 STEP 10
460   PRINT AT F, C
470   F = F + 5
480   NEXT C
490   IF R > 1 THEN 9300
495   REM * SUBMARINE OUT OF AREA ? * *  + IS LOGIC OR *
500   P = 0
510   IF (X < = 0) + (Y < = 0) THEN P = 1
520   Q = 0
530   IF (X > = 110) + (Y > = 30) THEN Q = 1
540   X = ABS (X)
550   Y = ABS (Y)
560   A(R) = X
570   A(A(0)) = Y
580   R = R + 2
590   A(0) = A(0) + 2
600   PRINT AT 1009, "ESCAPES ="; Z;
610   PRINT AT 832, "ENTER DEPTH CHARGE COORDINATES (0 TO 110,0 to 30)."
620   FOR C = 4 TO 16 STEP 2
630   RESET (C,42)
640   NEXT C
650   INPUT H,V
660   W = W + 1
670   PRINT AT 905,"YOUR LAST ENTRY =";H;", ";V
675   REM   -- * IS LOGIC AND ---
680   IF (H < 111) * (V < 31) GOTO 730
690   PRINT AT 832, "THE COORDINATES ENTERED ARE OUT OF AREA."
700   FOR C = 0 to 2000
710   NEXT C
720   GOTO 610
730   SET (H,V)
735   REM * TEST DIRECT HIT *
740   IF (H = X) * (V = Y) THEN 2000
745   REM  * DECIDE IF NEAR MISS *
750   IF S = 0 THEN 810
760   L = X + E
770   M = X - E
780   N = Y + E
790   O = Y - E
800   IF ((H > = M) * (H < = L)) * ((V > = O) * (V < = N)) THEN S = 0
810   IF S = 0 THEN PRINT AT 936, "SUBMARINE DISABLED !"
820   GOSUB 5000
830   IF P = 1 THEN 990
840   IF Q = 1 THEN 1400
845   REM  * COMPUTE SUB'S NEXT MOVE *
850   T = RND (4)
860   ON T GOTO 870, 900, 930, 960
870   X = X + S
880   Y = Y + S
890   GOTO 500
900   X = X + S
910   Y = Y - S
920   GOTO 500
930   X = X - S
940   Y = Y + S
950   GOTO 500
960   X = X - S
970   Y = Y - S
980   GOTO 500
985   REM  * Z IS ESCAPE COUNTER *
990   Z = Z + 1
1000  IF Z > 4 THEN 3100
1050  REM  * ADD 1 TO STOP ANY IN/OUT LOOP *
1100  X = X + 1
1200  Y = Y + 1
1300  GOTO 870
1400  Z = Z + 1
1500  IF Z > 4 THEN 3100
1600  X = X - 1
1700  Y = Y - 1
1800  GOTO 960
1900  END
2000  CLS
2100  PRINT AT 464, "CONGRATULATIONS ! YOU SUNK IT !"
2200  PRINT AT 593,"YOU DROPPED";W;"DEPTH CHARGES."
2300  PRINT
2400  PRINT
2500  PRINT
2600  PRINT
2700  INPUT "ENTER 1 FOR A NEW GAME OR 2 TO SEE THE SUBMARINE'S COURSE";C
2800  IF C = 2 THEN 9000
2900  GOTO 10
3000  END
3100  CLS
3200  PRINT AT 464,"SORRY YOU LOST - HE GOT AWAY!"
3300  GOTO 2200
3400  END
4900  REM  * CALCULATE BEARING *
5000  I = H - X
5100  I = ABS (I)
5200  J = V - Y
5300  J = ABS (J)
5400  U = (I * I) + (J * J)
5500  D = U/2
5600  G = 0
5700  K = (U/D - D)/2
5800  IF (K = 0) + (K = G) THEN 6200
5900  D = D + K
6000  G = K
6100  GOTO 5700
6200  D = D + .5
6300  D = INT (D)
6400  A$ = "  "
6500  B$ = "  "
6600  IF H > X THEN A$ = WEST
6700  IF H < X THEN A$ = EAST
6800  IF V > Y THEN B$ = NORTH
6900  IF V < Y THEN B$ = SOUTH
7000  FOR C = 73 TO 93
7100  RESET (C,46)
7200  NEXT C
7300  PRINT AT 960,"THE SUBMARINE'S LAST BEARING WAS";D;B$;" ";A$;
7400  RETURN
8900  REM  DISPLAY SUBMARINE'S COURSE *
9000  CLS
9100  GOTO 260
9200  SET (A(R),A(A(0)))
9300  R = R - 2
9400  A(0) = A(0) - 2
9500  IF R < 1 THEN 9700
9600  GOTO 9200
9700  PRINT AT 896,"PRESS 1 FOR A NEW GAME."
9800  INPUT A
9900  GOTO 10
10000 END
```

Print Board.

Store posns.

Erase.

INPUT C.



Hop on a Nasbus to 32K of memory now

The Nascom – 1 is designed with expansion in mind. This is made possible by using the best products available. The Z80 microprocessor incorporated in the basic system is so powerful it can support 64K bytes of memory and 256 ports. To utilize this capability, we have designed the buffered 77 – way Nasbus.

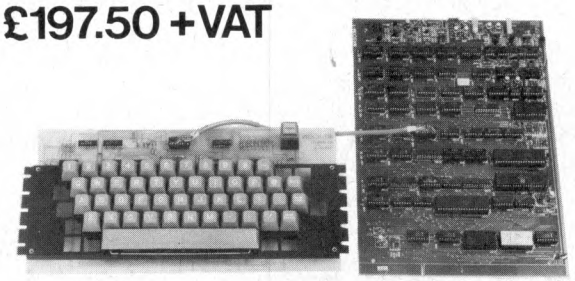
With this arrangement, the way is clear for considerable expansion, starting with our new memory expansion board. It has 16 memory sockets and two EPROM sockets. Therefore, you can fill it with 4K dynamic RAM up to a maximum 8K or with 16K dynamic RAM up to a maximum of 32K. A 2K Tiny BASIC in EPROM has been developed for the board.

To go with the board, we have produced a very flexible I.O board with three PIOs each giving two, 8bit ports, plus a UART for serial interface.

As you start building up your Nascom system you will need a convenient means of storing boards. Our new, custom-designed, Vero frame will allow for a Nascom – 1 to link through a buffer board to a 77-way Motherboard. There is then the option of eight or more expansion boards. To power this capability there is a new 8.5 amp power supply especially designed for the frame.

No other system offers so much at such a low cost. And it all starts with the basic Nascom – 1 kit which for just £197.50 offers an intelligently usable system with video and cassette interface, a full alpha-numeric keyboard and a mighty CPU chip. So if you want the best – make it a Nascom system.

Nascom-1 Kit still only £197.50 +VAT



UK National Distributors

Camera Centre, Barrow-in-Furness, Cumbria
Crystal Electronics, Torquay, Devon
Electrovalue, Egham & Manchester
Eley Electronics, Glenfield, Leicester
Henry's Radio, London W2
Lock Distribution, Oldham, Lancs
Lynx Electronics, Chesham, Bucks
Microdigital, Liverpool L2
Teleplay, New Barnet, Herts

Stop press...

Microcomputer Seminar

Nascom Microcomputer's highly successful seminar is coming to Bristol. The programme will be similar to London and Manchester, both of which were sold out. The day includes five lectures, demonstrations and an open forum. Venue is the Dragonara Hotel, Bristol, Saturday, October 14th, 09.50 to 17.30.

Admission: £4.50 (inc. VAT). Lunch will be available at £4.00 (inc. VAT) per head if there is sufficient demand.

Please send me

..... tickets to your seminar at £4.50 each and further details on Nascom-1 expansion products/Nascom-1 Kit/Int. Nascom Microcomputer Club.*

*delete as applicable

Name _____

Address _____

_____

Tel. No _____

Nascom Microcomputers
92 Broad Street, Chesham, Bucks.
Tel: (02405) 75151

Cut out coupon and post to Nascom Microcomputers. Cheques and PO's should be made payable to Nascom Microcomputers.

PCW 1078

MINOTAUR

A NEW COMPUTER GAME IN BASIC

John D. Lee and Timothy D. Lee

In ancient Greek mythology, the minotaur was the son of The Queen of Crete, and was compelled by the King to live in a labyrinth of caves because he was so ugly. The King decreed that every five years six maidens and six warriors from conquered Athens should be sent into the labyrinth. Since they never returned, it was rumoured that they were eaten – or worse! In fact they became hopelessly lost, and tired, and fell easy prey to the blood thirsty and carnivorous minotaur. The unlucky victims were selected by drawing lots, and when the intelligent son of the King of Athens was picked he devised a cunning plan. As he entered the labyrinth he unwound a ball of string, which enabled him to find his way out later. He finally caught up with the minotaur, slayed the beast, and lived to tell the story.

Initial placing of the men and minotaur on the board

For the purposes of the game, the labyrinth has been replaced by a board of variable size. The size of the board S is chosen by the user for each run, but must be at least 7 x 7 to obtain a sensible game and must not exceed 15 (the limiting size of the array B(15,15)). Initially S men are placed in random positions on the board, and the minotaur is placed in a random position towards the middle of the board. The positions of the men and the minotaur on the board are then printed showing men as an asterisk and the minotaur as M.

How the minotaur moves

Next the minotaur moves. The normal move is one square in an orthogonal direction (up, down left or right) into an empty square. If, however, the minotaur can move diagonally by one square into a square occupied by a man, then it will always do so in preference to the normal move, and the unfortunate person is devoured.

Moving your men

Next you are invited to move two of your men in an attempt to surround the minotaur (see winning and losing). To move your man you type in his initial Cartesian coordinates, and then the coordinates of the square to which he is moving. Several checks are performed on the input coordinates to ensure that they are integers in the range 1 to S, that a man occupies the initial location, that the final location does not already contain a man, and that the Pythagorean distance between the two squares is not greater than 3. If by misfortune you move a man to a square occupied by the minotaur, the man is eaten instantly and a warning message is printed. Having successfully moved one man, the computer invites you to move a second man (or of course the same man for a second move).

Printing the board

The board is then printed out showing the current positions of any remaining men. The position of the minotaur is not printed because it is so dark inside the cave that he cannot be seen!

Winning and losing

The cycle minotaur's move, your two moves, print the board is repeated. To win the game, the minotaur must be left so that it cannot move. This implies that it is surrounded orthogonally by four men if in the middle of the board, or three men if at the edge of the board, or by two men if in a corner. Remember that if the minotaur is completely surrounded orthogonally it still may be able to move diagonally to eat a man. Obviously it is easier to capture the minotaur at the edge of the board, which is why the minotaur always starts towards the middle.

Unless the minotaur is captured within a given number of moves (calculated taking account of the board size as INT(S*S/4)) you have lost the game. Alternatively you will lose if through incompetence all your men are eaten! If you admit defeat, you may type 0,0 when asked to type coordinates. The board is then printed showing the current positions of men and minatour. The board is also printed when the game is won or lost. You are then asked if you would like another game.

Some program refinements and explanations

To help you survive the ravages of this hungry monster, you are invited to place an extra man on the board every fifth move, corresponding to an extra batch of victims every fifth year.

The minotaur's movements and eating habits are apparently unpredictable. The normal moves are stored in the M array as four pairs of numbers describing the displacements right, up, left and down (1,0; 0,1; –1,0; 0,–1). This list is actually stored twice, and the starting point in the list is determined by a random number. The squares to which the minotaur may move may be this means be searched in any one of the following four orders, R,U,L,D; U,L,D,R; L,D,R,U or D,R,U,L. Provided a man cannot be eaten, the first available empty square is chosen – and if no such square exists the minotaur is surrounded and the player has won. The eating moves are stored in the E array and are chosen in a similar random manner.

The random number sequence is initialised by asking for a time check. On some machines a random starting point in the random sequence is automatically chosen, whilst on other machines statements RANDOMIZE or RND (-1) may have the same effect. The time check ensures a different starting board on any machine, but may be omitted if unnecessary.

All the arrays and string used are declared in the first statement. The strings are declared assuming that one location contains one character. Whilst this will work on any machine which can handle strings, it may be wasteful, and on some machines the string declaration may be omitted.

The BASIC matrix functions MAT ZER, MAT READ and MAT = are used. If matrix functions are not available, these can easily be replaced by simple coding. Apart from this, the program is written in the most elementary sub-set of BASIC, and should be easy to implement on virtually all computers which provide BASIC.

A listing of the source program, together with part of a sample run are provided.

```
10 DIM B(15, 15), M(16), E(16), F(16), Q$(10), P$(3), R$(10)
20 PRINT TAB(20); "MINATCUR"
30 PRINT TAB(20); "========"
40 PRINT
50 PRINT "TYPE SIZE IN RANGE 7 - 15"
60 INPUT S
70 IF (S - 7) * (S - 15) > 0 THEN 50
80 IF S = INT(S) THEN 110
90 PRINT "RETYPE AN INTEGER BETWEEN 7 AND 15"
100 GOTO  50
110 PRINT
120 PRINT "WCULD YCU LIKE INSTRUCTIONS ? ";
130 PRINT "TYPE YES OR NO AND PRESS RETURN."
140 INPUT Q$
150 IF Q$ = "NC" THEN 350
160 IF Q$ = "YES" THEN 190
170 PRINT "REPLY '"; Q$; "' NCT UNDERSTCOD. ";
180 GOTO  130
190 PRINT
200 PRINT "THE MINATCUR IS AN ANIMAL, INVISIBLE AFTER THE FIRST TURN"
210 PRINT "WHICH MUST BE CAPTURED BY YOU WITHIN";INT(S * S / 4); "DAYS."
220 PRINT "THE MINATOUR LIVES CN A"; S; "BY"; S; "BOARD "
230 PRINT "TC CAPTURE THE MINATOUR YOU MUST LEAVE IT SC THAT IT CAN"
240 PRINT "NCT MCVE . NORMALLY THE MINATOUR WILL MCVE CRTHOGONALLY"
250 PRINT "(IE. 1 SQUARE UP, DCWN, LEFT OR RIGHT ) INTC AN EMPTY SPACE"
260 PRINT "HCWEVER THE MINATCUR WILL MCVE DIAGCNALLY INTC A SQUARE"
270 PRINT "OCCUPIED BY ONE OF YOUR MEN AND BY SC DOING EAT YOUR MAN!"
280 PRINT "AT THE BEGINNINC OF THE GAME YOU HAVE"; S; "MEN AT YOUR"
290 PRINT "CCMMAND AND ON YOUR MOVE YOU CAN MOVE TWO OF THEM UP TC"
300 PRINT "THREE SQUARES EACH . EVERY FIFTH MCVE A NEW MAN WILL JCIN"
310 PRINT "YOUR FORCES AND YOU WILL BE ASKED WHERE ON THE BOARD YOU"
320 PRINT "WCULD LIKE TC POSITICN THIS MAN."
330 PRINT "AT ANY TIME YOU MAY QUIT BY TYPING 0,0"
340 PRINT
350 PRINT "TIME CHECK - HOW MANY MINUTES PAST THE HOUR"
360 INPUT T
370 FCR I = 1 TO T * T
380 LET X = RND(0)
390 NEXT I
400 PRINT "GCCD HUNTING    AND GCOD LUCK    ( YCU MAY NEED IT !)"
410 PRINT
420 MAT READ M
430 DATA 1, 0, 0, 1, -1, 0, 0, -1, 1, 0, 0, 1, -1, 0, 0, -1
440 MAT READ E
450 DATA 1, 1, 1, -1, -1, 1, -1, -1, 1, 1, 1, -1, -1, 1, -1, -1
460 MAT  F = E
470 MAT  B = ZER
480 LET P$ = "M  "
490 REM ****RANDOMLY PLACE YOUR MEN
500 LET I = 0
510 LET M1 = INT(RND(0) * S + 1)
520 LET M2 = INT(RND(0) * S + 1)
530 IF B(M1, M2) <> 0 THEN 510
540 LET B(M1, M2) = 1
550 LET I = 1 + 1
560 IF I < S THEN 510
570 REM **** RANDOMLY PLACE MINATQUR
580 LET M1 = INT(RND(0) * (S - 4)) + 3
590 LET M2 = INT(RND(0) * (S - 4)) + 3
600 IF B(M1, M2) <> 0 THEN 580
610 LET B(M1, M2) = 10
620 FOR I = 1 TO S * S / 4
630 REM ***** PRINT BOARD
640 LET M = 0
650 FCR X = 1 TC S
660 PRINT (S - X + 1); TAB(4);
670 FCR Y = 1 TO S
680 IF B(Y, (S - X + 1)) = 1 THEN 740
690 IF B(Y, (S - X + 1)) = 10 THEN 720
700 PRINT ".  ";
710 GOTO  760
720 PRINT P$;
730 GOTO  760
740 PRINT "*  ";
750 LET M = M + 1
760 NEXT Y
770 PRINT
780 NEXT X
790 FOR I1 = 1 TC S
800 PRINT TAB(2 * I1 + 1); I1;
810 NEXT I1
820 PRINT
830 IF P$ = "M  " AND I > 2 THEN 1590
840 LET P$ = ".  "
850 REM **** PLACE MAN IF 5TH, 10TH ETC. MCVE
860 IF INT(I / 5) * 5 <> I THEN 970
870 PRINT "WHERE WCULD YCU LIKE A NEW MAN?"
880 GOSUB  1710
890 IF Q = 1 THEN 1220
900 IF B(X, Y) <> 1 THEN 930
910 PRINT "WAKEY-WAKEY - YOU ALREADY HAVE A MAN THERE RE-INPUT"
920 GOTO  880
930 IF B(X, Y) <> 0 THEN 960
940 PRINT "YUM-YUM - THE MINATOUR HAS JUST HAD A GOOD MEAL!"
950 GOTO  980
960 LET B(X, Y) = 1
970 REM **** SEE IF MINATCUR CAN EAT A MAN
980 PRINT
990 LET E1 = 1
1000 LET K = INT(RND(0) * 4) * 2 + 1
1010 FOR J = K TO K + 6 STEP 2
1020 IF M1 = -E(J) THEN 1140
1030 IF M1 + E(J) = S + 1 THEN 1140
1040 IF M2 = -E(J + 1) THEN 1140
1050 IF M2 + E(J + 1) = S + 1 THEN 1140
1060 IF B((M1 + E(J)), (M2 + E(J + 1))) <> E1 THEN 1140
1070 LET B(M1, M2) = 0
1080 LET M1 = M1 + E(J)
1090 LET M2 = M2 + E(J + 1)
1100 LET B(M1, M2) = 10
1110 IF E1 = 0 THEN 1250
1120 PRINT "YUM-YUM - THE MINATOUR HAS JUST HAD A GOCD MEAL"
```

946. M = M-1
1126 m = M-1.

```
1130 GOTO  1250
1140 NEXT J
1150 IF E1 = 0 THEN 1200
1160 REM **** GO BACK AND SEE IF MINATOUR CAN MOVE
1170 MAT  E = M
1180 LET E1 = 0
1190 GOTO  1000
1200 PRINT "YOU HAVE SURROUNDED THE MINATOUR - WELL DONE !"
1210 PRINT "THIS IS WHAT THE BOARD LOOKS LIKE."
1220 LET P$ = "M  "
1230 MAT  E = F
1240 GOTO  630
1250 FOR J = 1 TO 2
1260 IF M = 0 THEN 1540
1270 PRINT "WHICH MAN WOULD YOU LIKE TO MOVE (X,Y) ?"
1280 GOSUB  1710
1290 IF Q = 1 THEN 1220
1300 IF B(X, Y) = 1 THEN 1330
1310 PRINT "STOP CHEATING - YOU HAVE NOT GOT A MAN AT (";X;",";Y; ") !"
1320 GOTO  1270
1330 LET X1 = X
1340 LET Y1 = Y
1350 PRINT "TO WHERE (X,Y) ?"
1360 GOSUB  1710
1370 IF Q = 1 THEN 1220
1380 IF SQR((X1 - X) * (X1 - X) + (Y1 - Y) * (Y1 - Y)) <= 3 THEN 1420
1390 PRINT "HAVE YOU READ THE RULES - YOU CAN ONLY MOVE A MAN 3 ,NOT";
1400 PRINT INT(1000 * SQR((X1-X) * (X1-X) + (Y1-Y) * (Y1-Y)) + .5) /1000
1410 GOTO  1270
1420 IF B(X, Y) <> 1 THEN 1450
1430 PRINT "THIS SQUARE IS ALREADY OCCUPIED BY ONE OF YOUR OWN MEN !"
1440 GOTO  1270
1450 IF B(X, Y) = 0 THEN 1490
1460 PRINT "YUM-YUM - THE MINATOUR HAS JUST HAD A GOOD MEAL !"
1470 LET M = M - 1
1480 GOTO  1500
1490 LET B(X, Y) = 1
1500 LET B(X1, Y1) = 0
1510 NEXT J
1520 NEXT I
1530 GOTO  1560
1540 PRINT
1550 PRINT "ALL YOUR MEN HAVE BEEN EATEN !"
1560 PRINT
1570 PRINT "YOU HAVE FAILED YOU MISERABLE SKUNK !"
1580 GOTO  1210
1590 PRINT
1600 PRINT "WOULD YOU LIKE ANOTHER GAME ?"
1610 IF Q$ = "NO" THEN 1630
1620 PRINT "TYPE YES OR NO AND PRESS RETURN."
1630 LET Q$ = "NO"
1640 INPUT R$
1650 IF R$ = "NO" THEN 1850
1660 RESTORE
1670 IF R$ = "YES" THEN 350
1680 PRINT "REPLY '"; R$; "' NOT UNDERSTOOD. ";
1690 GOTO  1620
1700 REM **** SUBROUTINE TO CHECK CO-ORDINATES
1710 INPUT X, Y
1720 LET Q = 0
1730 IF X <> INT(X) THEN 1750
1740 IF Y = INT(Y) THEN 1780
1750 PRINT "CO-ORDINATES MUST BE WHOLE NUMBERS"
1760 PRINT "PLEASE RE-TYPE CORRECTLY."
1770 GOTO  1710
1780 IF (X - 1) * (X - S) > 0 THEN 1810
1790 IF (Y - 1) * (Y - S) <= 0 THEN 1840
1800 IF ABS(X) + ABS(Y) = 0 THEN 1830
1810 PRINT "CO-ORDINATES MUST BE IN THE RANGE 1 -"; S
1820 GOTO  1760
1830 LET Q = 1
1840 RETURN
1850 END
```

```
                    MINATOUR
                    ========

TYPE SIZE IN RANGE 7 - 15
? 7

WOULD YOU LIKE INSTRUCTIONS ? TYPE YES OR NO AND PRESS RETURN.
? YES

THE MINATOUR IS AN ANIMAL, INVISIBLE AFTER THE FIRST TURN
WHICH MUST BE CAPTURED BY YOU WITHIN 12 DAYS.
THE MINATOUR LIVES ON A 7 BY 7 BOARD
TO CAPTURE THE MINATOUR YOU MUST LEAVE IT SO THAT IT CAN
NOT MOVE . NORMALLY THE MINATOUR WILL MOVE ORTHOGONALLY
(IE. 1 SQUARE UP, DOWN, LEFT OR RIGHT ) INTO AN EMPTY SPACE
HOWEVER THE MINATOUR WILL MOVE DIAGONALLY INTO A SQUARE
OCCUPIED BY ONE OF YOUR MEN AND BY SO DOING EAT YOUR MAN!
AT THE BEGINNING OF THE GAME YOU HAVE 7 MEN AT YOUR
COMMAND AND ON YOUR MOVE YOU CAN MOVE TWO OF THEM UP TO
THREE SQUARES EACH . EVERY FIFTH MOVE A NEW MAN WILL JOIN
YOUR FORCES AND YOU WILL BE ASKED WHERE ON THE BOARD YOU
WOULD LIKE TO POSITION THIS MAN.
AT ANY TIME YOU MAY QUIT BY TYPING 0,0

TIME CHECK - HOW MANY MINUTES PAST THE HOUR
? 23.5
GOOD HUNTING  · AND GOOD LUCK  · ( YOU MAY NEED IT !)

 7  .  *  .  .  .  .  .
 6  .  .  .  .  .  .  .
 5  .  .  *  .  .  *  .
 4  .  .  .  .  .  .  .
 3  .  *  M  .  .  .  .
 2  .  .  .  .  *  .  .
 1  .  .  *  *  .  .  .
    1  2  3  4  5  6  7
```

```
WHICH MAN WOULD YOU LIKE TC MOVE (X,Y) ?
? 4,1
TO WHERE (X,Y) ?
? 4,2
WHICH MAN WCULD YCU LIKE TC MCVE (X,Y) ?
? 2,7
TC WHERE (X,Y) ?
? 2,4
   7 .  .  .  .  .  .  .
   6 .  .  .  .  .  .  .
   5 .  .  *  .  .  .  .
   4 .  *  .  .  .  .  .
   3 .  .  .  .  .  .  .
   2 .  .  .  *  *  .  .
   1 .  .  *  .  .  .  .
     1  2  3  4  5  6  7

YUM-YUM - THE MINATCUR HAS JUST HAD A GOOD MEAL
WHICH MAN WOULD YCU LIKE TO MOVE (X,Y) ?
? 5,2
TO WHERE (X,Y) ?
? 3,3
WHICH MAN WOULD YOU LIKE TO MCVE (X,Y) ?
? 3,1
TO WHERE (X,Y) ?
? 2,2
   7 .  .  .  .  .  .  .
   6 .  .  .  .  .  .  .
   5 .  .  .  .  .  *  .
   4 .  *  .  .  .  .  .
   3 .  .  *  .  .  .  .
   2 .  .  .  *  .  .  .
   1 .  .  .  .  .  .  .
     1  2  3  4  5  6  7

WHICH MAN WOULD YOU LIKE TO MOVE (X,Y) ?
? 2,2
TO WHERE (X,Y) ?
? 3,4
WHICH MAN WOULD YOU LIKE TO MOVE (X,Y) ?
```


PET 2001

TRS 80 Level 2

APPLE II

Software Now Available On Cassettes

01 INVESTMENT DECISIONS
02 STATISTICS (A)
03 STATISTICS (B)
04 MATHEMATICS (A)
05 MATHEMATICS (B)
06 PERSONAL FINANCE MANAGEMENT
07 SMALL BUSINESS MANAGEMENT
08 STOCK CONTROL
09 ADDRESS LISTING
10 ENGINEERING
11 APPOINTMENT SCHEDULING
12 STOCK EXCHANGE PORTFOLIO MANAGEMENT
13 SUBSCRIPTION LISTING
14 GAMES

Please reply to:-
Mr. D. C. JAMES,
EDINSMITH COMPUTER SERVICES Ltd.,
11, Stoke Hill,
Stoke Bishop,
BRISTOL, BS9 1JL.

TOUCHDOWN

'LUNAR LANDING GAME FOR CASIO PROGRAMMABLE CALCULATORS'.

N. Rushton

This version of the ever popular Lunar Landing game has been written to run on any of the following calculators made by Casio: FX201P, FX202P, PRO FX1. It occupies 125 programme steps and uses 8 of the calculator's 10 memories.

The object of the game is to land an imaginary spacecraft upon the lunar surface at a safe touchdown velocity. The programme commences with the spacecraft at altitude A, and heading towards the lunar surface with a velocity V. It is also accelerating under the lunar gravity. The calculator requests an amount of fuel to be burnt – the effect of this 'burn' is to deaccelerate the spacecraft as it descends. If the burn entered is an acceptable amount – the programme imposes a maximum limit and checks that there is fuel left to burn – then new values of A and V are calculated: and these are displayed along with the amount of fuel remaining (F). Then a new burn is requested. In addition, the status of A, V and F may be checked at any time by use of the MJ command.

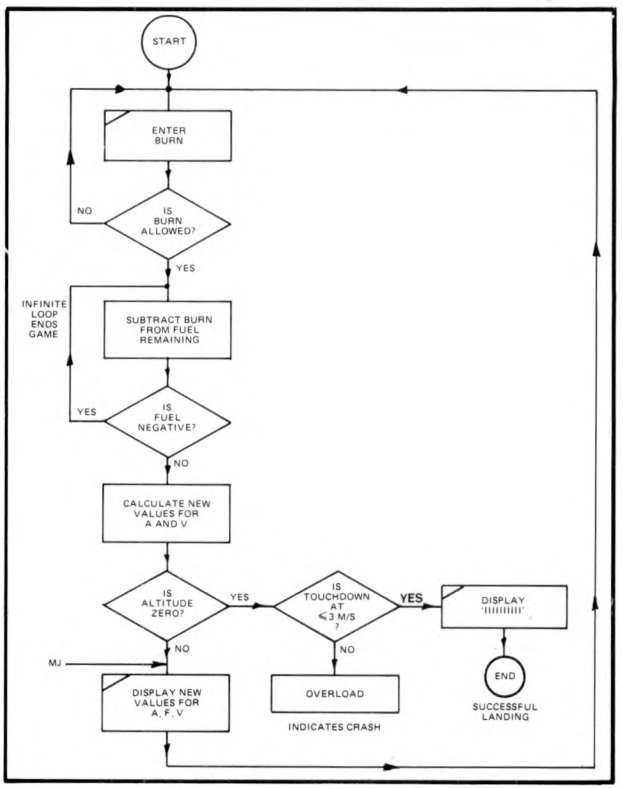
When the spacecraft reaches zero altitude i.e. touchdown, the programme checks that this occurred at an acceptable velocity. This velocity is 3 metres sec. or less, else a crash landing has occurred. This is indicated by the display of 'E' (i.e. an overload has occurred and the registers of the calculator have locked). A successful landing is indicated by a row of 1's appearing in the display. If the spacecraft runs out of fuel before landing then this is indicated by the programme entering an infinite loop and the display of a '–' sign.

Enter the programme (a full listing and a flowchart are provided below). **Before running,** switch to Manual and enter the following data:

In memory 0; Altitude A . . . suggested value 2500 metres.
In memory 1; Velocity V . . . suggested value 500 metres sec.
In memory 2; Fuel F . . . suggested value 600 units.
In memory 8; Maximum burn allowed . . . suggested value 75 units.
In memory 9; Successful landing code – 1111111111
Switch to Comp mode and run the programme.

Summary of running programme:
Enter burn when requested in memory 3. If the amount entered is greater than the maximum allowed, then a revised value is requested. Negative amounts should not be burned (it's cheating!). New values for A, V and F are given in memories 0, 1 and 2 respectively. These may be checked at any time by use of MJ key. If V is given as negative, then that indicates the spacecraft is going upwards and you would have been useless in Apollo.

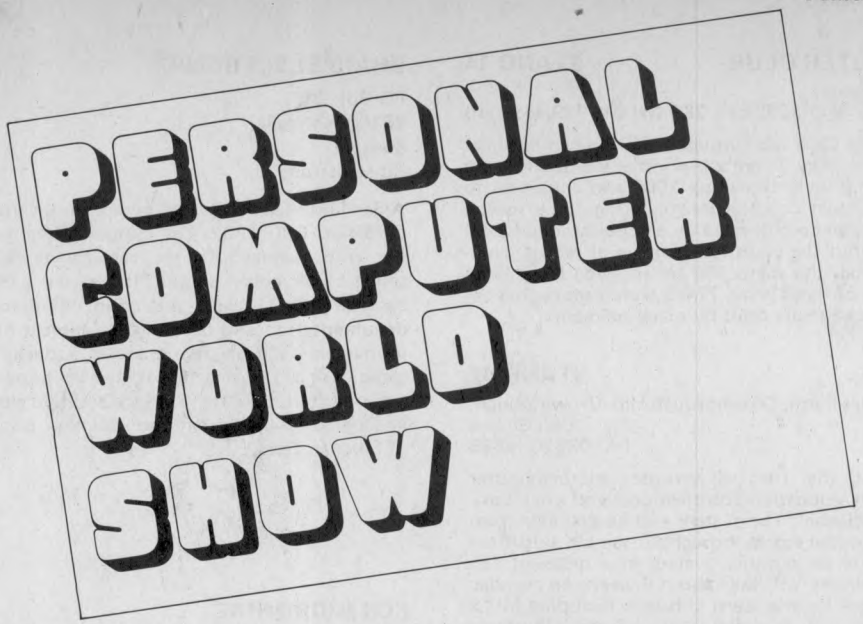
The objective is to land at 3 metres sec. or less.
'1111111111' . . . Successful landing.
'E' . . . Crash landing.
'–' . . . Run out of fuel.



LUNAR LANDING GAME.

```
ST #1:  ENT 3:
        IF 3 = 8:2:2:1:
ST #2:  2 = 2 – 3:
        IF 2 = KO:2:3:3:
ST #3:  4 = 3 – K2 – 1:
        6 = 1 – 4 ÷ K2:
        0 = 0 – 6:
        1 = 4 +/–:
        IF 0 = KO:4:4:5:
ST #4:  IF 1 = K3:7:7:6:
ST #5:  MJ ANS 0:1:2:
        GOTO 1:
ST #6:  9 = 9 e^x:
ST #7:  ANS 9:
```

PROGRAMME LISTING



GENERAL INFORMATION

Opening hours:
10.00 – 19.00 Thursday 21 September
10.00 – 19.00 Friday 22 September
10.00 – 17.00 Saturday 23 September

West Centre Hotel:
The Hotel is situated in Lillie Road, London SW6. The closest London Transport Underground Station is West Brompton (5 minutes walk), and buses, 30, 74 and 74b pass the Hotel.

Admission to the Show:
Admission is by pre-paid ticket (£1) or by programme at the door (£1). Free to conference pass holders.

Feature area:
A special feature area has been set aside and will be of particular interest to the visitor. It will display the efforts of the contenders in the Personal Computer World homebrew competition, with prizes of £200 each for the best application in four different categories. And the Personal Computer World **micro-chess championships.** Model railway enthusiasts can see a computer controlled layout developed by CAP Software. Leslie Solomon, Editor of Popular Electronics in the United States and regarded as the 'father' of home computing across the Atlantic will be there with his new voice-controlled Sol computer (which includes a chess programme that came 2nd in the world championship). He will also be demonstrating, for the first time in Europe, a new graphics breakthrough.

Refreshments and catering:
The West Centre Hotel is well provided with bars and restaurants. A special bar for visitors and exhibitors is situated in the foyer of the exhibition and is open daily from 12 noon.

The Centre Bar on the Hotel ground floor is easily accessible from the exhibition and is open from 11 am.

The Coffee Shop on the first floor is open throughout the day and offers coffee, snacks and quick, modestly priced meals.

The Carvery Restaurant offers a wide selection of joints and has the Baron Carver menu of three courses for £4.25 + VAT.

Telephones:
Six pay telephone booths are situated in the exhibition foyer.

Cloakrooms and lavatories:
These facilities are situated in the exhibition foyer.

Sales from stands:
Visitors are advised that for security reasons they must obtain a receipt for any purchase made from stands which will be demanded by the security officer at the gate.

EXHIBITORS

Company	*Stand*
Attaché	42
Amateur Computer Club	14
Belvedere Computer Services (Scarborough) Ltd	15
BHAIB Electronics	27
The Byte Shop Ltd	19 – 24
Bywood Electronics Ltd	37 + 38
CCS Microhire	49
Collins Consultants	36
Comart Ltd	2 + 3
The Computer Bookshop	4
Computer Workshop (South West Tech)	28 + 29
Crofton Electronics Ltd	1
Cytek (UK) Ltd	12
Datac Ltd	41
Keen Computers	16
L P Enterprises	8 + 9
Limrose Microprocessors Division	30
Microdata Gmbh	46
Micronics Ltd	10 + 11
Nascom Microcomputers Ltd	34 + 35
Newbear Computing Store	13
Pelco (Electronics) Ltd	39
Personal Computers Ltd	18
Research Machines Ltd	7
Sintrom Microshop	25
Sirton	40
Star Devices Ltd	17
Strumech Engineering Ltd	54
Tandberg UK Ltd	47
Tandy Corporation (Branch UK)	6
Technalogics	12

(List correct at time of going to Press).

AMATEUR COMPUTER CLUB — STAND 14

7 Dordells, Basildon, Essex.
R. W. Warren Tel: 01-877 3252 ext. 289 (w) 01-979 4193 (h)

The Amateur Computer Club was formed in 1973 when personal computing was in its infancy. From a few initial members, it has grown to a membership well exceeding 1000 and continues to expand rapidly. Like most organisations catering for a special hobby pursuit, it is largely co-ordinated by a newsletter and local centre groups throughout the country. Samples of recent newsletters are available from the stand. We are showing two home-made computers. One of these is the 77-68, a club sponsored design which has been successfully built by many members.

ATTACHÉ — STAND 42

Moncoland Ltd., Beeches Farm, Crowborough Hill, Crowborough, East Sussex
Roger R. Barnes Tel: 08926-4665

Moncoland announces the first all-purpose microcomputer designs for the hobbyist, educational/professional and small business user called the 'Attache'. The system will be available from November 1 in leading retail stores throughout the UK but those interested in purchasing an advance system at a discount can come to the Show. Delivery will take about 6 weeks on average. The system is supported by the latest software including MITS BASIC release S.O. as well as a full range of small business accounting packages at reasonable cost. In addition, the system supports via its S-100 compatability a large range of peripherals at reasonable prices.

BELVEDERE COMPUTER SERVICES (SCARBOROUGH) LTD. — STAND 15

9 Belvedere Place, Scarborough, Yorks Tel: 0723-63638

Belvedere Computer Services Ltd., Main distributors for 'Imsai' micro computer products, from hobbyist to distributed data processing terminals for the largest of companies. Hardware and software support, we will be displaying a range of software, stock control, invoicing, sales ledger, word processing, profit and loss accounts, purchase analysis, budgetory control and others. There are over 15000 Imsai computers installed worldwide. Come, look, talk to us on Stand 15. Ask for Mark Proudfoot.

BHAIB ELECTRONICS — STAND 27

PO Box 216,
76100 Norrtalje
Sweden
Bo Hellstrom Tel: 0176 18025

A Scandinavian low-priced computer kit using the powerful RCA COSMAC CPU 1802. The system offers a complete package for use with standard TV set, cassette recorder, loudspeakers and single 5 volt power supply. It includes a 64 character keyboard, cassette with TV games and basic programme and comprehensive documentation also covering the subject of writing programmes in machine language, basic and in a special language for easy TV game design. On card 12k of RAM externally extendable to 32k, audio output, cassette interface and two video interfaces, one for photographic display and one for text programme with 16 lines of 64 characters.

CCS MICROHIRE — STAND 49

Freepost,
Letchworth,
Herts. SG6 4YA

CCS Microhire will be explaining the merits of hiring a microcomputer to get hands-on experience before you buy. The company will also be showing how you can use a program to learn about programming in BASIC. CCS offers hiring arrangements for micros from one day to as long as you like at a starting price from £2 per day with the option of a deliver and collect service. What is probably one of the biggest ranges of microcomputers available from a single source in the UK includes middle-of-range machines such as APPLE II, COMMODORE PET, NASCOM 1, MICROS MSI 6800, RESEARCH MACHINES 380Z, SOL – 20, SWTC 6800, TANDY TRS – 80 and extends upwards to include ALTAIR and down to single board machines such as KIM 1 which will be available for research purposes.

Main distributors for IMSAI products, reliable low cost, Micro Computers, cassette or multi-floppy disk based systems. Ideal for all business, industrial, medical and educational use. Simple and quick to learn programming with BASIC, FORTRAN, and ASSEMBLER languages.

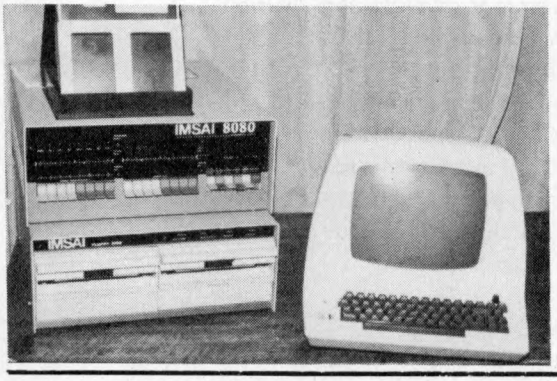
IMSAI 8080 computer – 10,000 sold in 18 months – the world's fastest and most successful selling machine. Flexibility that everyone can afford.

IMSAI latest product, the **VDP80** with 32K or 64K memory, 1 million alphabetic characters of disk memory and loaded with lots of super features.

Designed for you with thought.
Join the successful people – buy IMSAI.

Powerful programmes are available – designed so that you can tailor them to your needs without programming skill.





Profit/Loss accounts, Balance Sheet, Sales Ledger, Stock Control, Invoicing, Purchase Ledger, Budgetary Controls, Postal Lists, Production Control, and Word Processing.

All products are competitively priced for the finest budget. For details call or write:
Mark Proudfoot
0723-63638 or 67027

We are looking for suitable local and area distributors. Join our successful team supported by successful products.
IMSAI
The standard of excellence in Micro Computors.

P.s. All parts are guaranteed for 6 months, we have a repair service to back up our products.

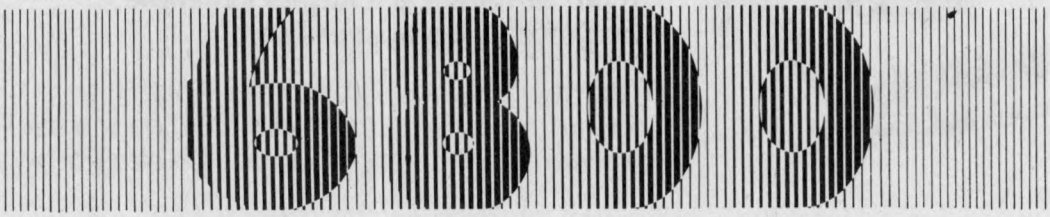
Belvedere Computer Services (Scarboro) Limited

9 Belvedere Place . Scarborough . North Yorkshire . YO11 2QX

Telephone 0723 63638 (67027 - 24 hours)

INTRODUCING

THE CROFTON



Micro Learning Modules

with 'FREE' Basic Interpreter

This exciting system has been developed to fulfil the very varied requirements of the purchaser. Whether you wish to learn machine programming, computer logic, basic concepts, programme development or control an industrial process, this package will meet your requirements.

VISIT US ON STAND 1 AT THE PERSONAL COMPUTER WORLD SHOW, WEST CENTRE HOTEL

The System Comprises - Mother board with plug in cards and is fully expandable.

On board facilities are also available to burn in your own Read Only Memory.

FOR FULL DETAILS SEND FOR OUR DATA SHEET LS1

OTHER CROFTON PRODUCTS INCLUDE:

Computers for the small business.	**Software.**	**CCTV Cameras.**
Computer peripheral equipment.	**Computer books.**	**CCTV Monitors.**
Computer boards for the hobbyist.	**Electronic Secretary.**	**Video Switchers.**
Educational Computer Kits.	**VHF/UHF Modulators.**	**Character Generator.**

Data sheets are available covering the above equipment.

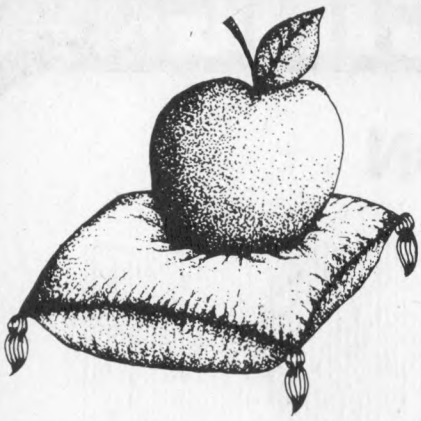


Crofton also provide a specialist Printed Circuit Board service for either prototypes or production boards. Full details will be supplied on request.

CROFTON ELECTRONICS LIMITED

35 GROSVENOR ROAD, TWICKENHAM, MIDDLESEX, TW1 4AD

Telephone: 01-891 1923



let apple soften you up!

(with software packages available from Keen Computers Ltd.)

The new disk units now make Apple II a powerful business machine. Our "off the shelf" software packages get you started straight away.

Use Apple's unique colour graphics to display sales trends, costing breakdowns, market prices etc.

Spend more time in promoting your business.

JOIN THE APPLE CORPS!

Apple disk (116 Floppy), controller and DOS – £395.00*

SOFTWARE 58 CASTLE BOULEVARD NOTTINGHAM 49588

keen computers ltd. (the apple experts)

COLLINS CONSULTANTS — STAND 36

Rotherglen, Gerrards Cross Road, Stoke Poges, Buckinghamshire SL2 4EJ
A. J. Collins — Tel: FULMER 2572/2465

The Stand will feature 3 configurations of system based on the APPLE II equipment.
Small system comprising 16K machine, TV Display and cassette recorder.
Application: Advertising Display.
Medium System comprising: 32K machine, TV Display, Diskettes and Daisy Wheel printer.
Application: Kitchen Planning and Word Processing.
Large System comprising: 48K machine, TV Display, Diskettes and fast printer.
Application: Business Applications.
The emphasis will be on the systems use of a versatile, cost effective equipment. Personnel will be available on the STAND to discuss individual applications, equipment requirements and resources needed for development.

COMART — STAND 2 + 3

P.O. Box 2, St. Neots, Cambs.
J. R. Lamb — Tel: 0480 215005

Comart are exhibiting their range of S100 Microcomputer Systems, sub-systems modules and software.
Computers on display include the System Three, Z2D and Z2 Cromemco range, Processor Technology's SOL together with memory, analog/digital interface cards and North Star Diskette system.
The choice of software for Comart's microcomputers is one of the most extensive in the industry: Assembler, Macro Assembler, Basic, Fortran and Cobol languages are all available.

COMPUTER BOOKSHOPS LIMITED — STAND 4

Temple House, 43/48 New Street, Birmingham B2 4LH
Margaret Maclean — Tel: 021 - 643 4577

Computer Bookshops Limited is a Trade and Mail Order distribution company, handling books, manuals, software and training programs for the industry, business user, education and the hobbyist.
The company deploys its expertise to enable customers to obtain the most effective use from their micro processors and computers. On the stand you will be able to see at a glance those books which are relevant to the hardware you are using and the stage of experience and interest you have reached.

COMPUTER WORKSHOP — STAND 28 +29

(South West Tech), 38 Dover Street, London W1
John Burnet — Tel: 01 - 491 7507

Computer Workshop is exhibiting its range of computer equipment of both small and large systems. At £4650 there is a CPU, one million characters of disc storage, VDU and a 60 cps printer with 132 column width. It has a disc operating system and sophisticated Basic. The operating system has facilities such as print job queueing and spooling while the computer is used for other purposes.
At the smaller end of the scale, is a CPU with 4K bytes of memory for £330. Additional memory is available at £70 per 4K; serial and parallel interfaces at £37 each; the VDU for £445; and a 40 column printer for £250.
Software such as a text editor and text processor, said to be as powerful as those found on larger mainframes is available from £25.
Most Computer Workshop equipment is manufactured in England by Southwest Technical Products.

CROFTON ELECTRONICS — STAND 1

35 Grosvenor Road, Twickenham, Middlesex
D. E. Pattinson — Tel: 01 - 891 1923

Being exhibited for the first time – An all BRITISH DESIGNED MOTOROLA 6800 MICRO SYSTEM for the small business and educational user. Including dual mini floppies, Keyboard and VDU. Other standard equipment from the Crofton range includes: Television Monitors suitable for display of both TV pictures and alpha numeric information, UHF Modulators allowing Domestic TV to be used as VDU. ASCII Free standing Keyboards – Mini Floppy Drives, Educational Computer boards, Character Generators, Computer Books. Hard Copy Printers and a whole lot of know how on interfacing the various peripheral devices to a micro computer system.





Goods are normally shipped within 24 hours subject to availability.
Barclay card & Access VAT at 8% for Hardware Components.
30p postage and packing unless otherwise stated.
Cheques to be made out to 'The Newbear Computing Store'

THE NEWBEAR COMPUTING STORE

Send for an up-to-date catalogue to:

The Newbear Computing Store

7 Bone Lane, Newbury.

Callers welcome Monday to Saturday 9.00a.m.-5.30p.m.
The Newbear Computing Store is a division of Newbury Laboratories Ltd.

Hardware Components Section

VIM 1

The new 6502 based micro from Synertek.
Fully assembled and tested £199.00
Send for further details + 8% VAT. Carriage £1.00

BEARBAGS (KITS)

1.	77-68 CPU PCB and components	£ 49.50
2.	77-68 LED's and switches	£ 14.95
3.	77-68 Power supply	£ 17.95
4.	77-68 19" 5u Rack and Backplane	£ 26.70†
5.	77-68 4K Ram PCB and components	£ 75.00
6.	77-68 Mon 1 PCB and components	£ 50.00
7.	4K Ram Exorcisor PCB and components	£ 71.50
8.	8K Ram Exorcisor PCB and components	£160.00*
9.	Petitevid VDU Kit	£ 85.00
10.	Kansas City Cassette interface	£ 18.95
11.	UHF Modulator	£ 4.50
12.	77-68 VDU PCB and components	£ 69.50
15.	PROMVERTER (Enables a 2708 to be used instead of MIKBUG)	£ 8.50

50p p+p unless otherwise stated † 1.50 p+p * 1.00 p+p

Unencoded Keyboard	£27.50 + £1.50 p+p
Licon Assembled & Tested Keyboard	£86.00 + £1.50 p+p
Hexadecimal Keypad (19 positions)	£11.25
Keyboard case	£20.33 + £1.50 p+p
19" 5u Rack	£23.40 + £1.50 p+p
U.V. Prom Eraser	£56.00*
9368 7 seg. Display Decoder	£2.45
FWD 500 ½" 7 seg. Display	£1.05
C-30 Cassette Memorex £0.75 5" Floppy Disc	£4.00
C-60 Cassette Memorex £0.90 8" Floppy Disc	£4.00
C-90 Cassette Memorex £1.20	

*£1.00 Postage & Packing

S-100 BUS

CROMEMCO Z-2 kit (CPU + Card Frame)	£395.00
BYTESAVER PROGRAMMING BOARD kit (2708)	£ 95.00
TU-ART Interface kit	£130.00
16K BYTE Dynabyte ram card (fully tested)	£415.00
North Star Mini Floppy kit	£490.00

MEMORIES

2101–1	£ 1.25
2102L–1	£ 1.35
2112	£ 3.04
2513	£ 6.50
SWATBUG	£16.00
4027	£ 4.50
2114	£ 8.25
2708	£ 6.99
MC6830L7	£13.65
MIKBUG	
4116 16K Dynamic Rams	£15.00

INTERFACING I.C's

SFF 96364	£16.20
MC1488P	£ 1.40
MC1489P	£ 1.40
75150P	£ 1.30
75150N	£ 1.20
75134	£ 2.30
4N33	£ 1.95
AY-5-1013	£ 4.50
6402	£ 4.50

MOTOROLA MICROCOMPUTING I.C.'s

MC6800	£9.26
MC6820	£4.63
MC6850	£6.74
MC6810AP	£3.61
MC8602P	£2.88
MC14536P	£3.69
MC3459	£2.53

ZILOG MICROCOMPUTING I.C's

Z80 CPU 2.5Mhz	£15.50
Z80 PIO	£10.00
Z80 CTC	£10.00
Z80A CPU 4 Mhz	£20.50
Z80A PIO	£11.55
Z80A CTC	£11.55

MICROPROCESSORS

SC/MP 11	£10.30
INS 8154N	£ 8.18
6502	£14.93
8080	£ 6.00

BOOKS

All books are subject to availability

		P. & P.
COMPUTER DESIGNS		
77-68 6800 Microcomputer	£ 7.50	.50
Spare diagram set for 77-68	£ 1.50	.50
WB–1 TTL Microcomputer	£ 6.50	.50
Spare diagram set for WB–1	£ 1.00	.50
FROM ADAM OSBOURNE ASSOCIATES		
Introduction to Microcomputers		
Volume 0: The Beginners Book	£ 5.95	.50
Volume 1: Basic Concepts	£ 5.95	.50
Volume 2: Some Real Products June 1977 Revision	£11.95	1.00
8080A/8085 Assembly Language Programming	£ 6.95	.50
6800 Assembly Language Programming	£ 5.95	.50
Some Common Basic Programs	£ 5.95	.50
6800 Programming for Logic Design	£ 5.95	.50
8080 Programming for Logic Design	£ 5.95	.50
Payroll with Cost Accounting in Basic	£ 9.95	1.00
SCELBI		
Understanding Microcomputers & Small Computer Systems	£ 7.56	.50
Scelbi 6800 Software Gourmet Guide & Cook Book	£ 7.95	.50
Scelbi 8080 Software Gourmet Guide & Cook Book	£ 7.95	.50
8080 Standard Assembler	£15.95	.75
The Scelbi Byte Primer	£ 9.95	1.00
The 8080 Programmers Pocket Guide	£ 2.35	.30
ZILOG		
Z80 Technical Manual	£ 4.00	.50
Z80 PIO Technical Manual	£22.50	.50
Z80 Programming Manual	£ 4.00	.75
MOTOROLA		
Understanding Microprocessors	£ 2.75	.30
M6800 Microprocessor Programming Manual	£ 4.50	.50
M6800 Microprocessor Applications Manual	£ 9.50	1.00
SYBEX		
Microprocessors C201	£ 8.00	.50
Microprocessors Interfacing Techniques	£ 8.00	.50
BASIC		
Instant Freeze-Dried Computer Programming in Basic By Jerald R. Brown	£ 4.95	.75
My Computer Likes Me When I Speak in Basic	£ 1.65	.30
Computer Programs That Work	£ 2.40	.75
Basic Software Library Volume 1:	£17.50	.50
Volume 2:	£17.50	.50
Volume 3:	£26.95	.50
Volume 4:	£ 7.95	.50
Volume 5:	£ 7.95	.50
Volume 6:	T.B.A.	.50
Volume 7:	£26.95	.50
What to do after you hit Return	£ 7.00	.75
NEW BOOKS		
First Book of KIM	£ 7.00	.50
Z80 Microcomputer Handbook	£ 7.25	.50
Using the 6800 Microprocessor	£ 6.25	.50
Micro 6502 Journal	£ 1.70	.50
Learning Basic Fast	£ 6.30	.50
How to Program Microcomputers	£ 6.30	.75
Getting Involved with your own Computer	£ 4.75	.50
8080A Bugbook	£ 6.95	.75

* VISIT OUR NEW STORE AT:
2 Gatley Road, Cheadle,
Cheshire. Tel. 061-491 0134 *

24 HOUR TURNROUND ON ORDERS
**Securicor delivery advised please phone for charges.*

Please add 8% VAT to all prices. P & P 30p unless otherwise stated. Barclaycard & Access welcome. Overseas orders issued with Pro-Forma Invoice. Send for catalogue to **NewBear Computing Store, Bone Lane, Newbury, Berks.** Callers welcome Mon – Sat, 9.00 – 5.30 but please phone us first on 0635 49223. New office – 2 Gatley Road, Cheadle, Cheshire (callers only) Tel: 061 491 0134

CYTEK (UK) Ltd — STAND 12

17 Exchange Hall, Corn Exchange Building, Manchester M4 3EY
C. N. Menhinick — Tel: 061 - 832 7604

Cytek (UK) Ltd is a Manchester based software and systems house, and is an authorised Pet computer dealer. Technalogics specialises in teletext and home computing. Collaboration has resulted in the first Pet compatible teletext decoder.
Cytek is showing new Pet software:— payroll, word processing, percentage costing, standard statistics and electronic cad. A teletype KSR43 matrix printer is also shown with the Pet.
Technalogics is launching its 6800 microprocessor based teletext decoder which doubles as a powerful and expandable home computer.
Also shown: dedicated decoder, battery portable colour TV pattern generator. In-house teletext system for information distribution available to order.

DATAC LTD — STAND 41

Tudor Road, Altrincham, Cheshire WA14 5TN
Michael J. Robinson — Tel: 061 - 941 2361/2

Datac Limited will be showing a wide range of low-cost digital matrix printers, both in fully-packaged and D.I.Y. form.
These cover the 16 to 80 column market and include full parallel (BCD), bit-parallel, character-serial (ASCII), and full serial (RS232C/V24 or 20 mA loop) data input interfaces. Also on show will be a range of low-cost Mini-Disk Drives and accessories.
A low-cost formatter board for the MDD will be available soon.
A range of small printer "kits" will be available for visitors to purchase on the stand and take away, thus saving carriage costs.

KEEN COMPUTERS LTD — STAND 16

58 Castle Boulevard, Nottingham NG7 1FN
Dr. Tim Keen — Tel: 49588

Keen Computers Ltd., specialise in the marketing of the Apple II computer and the production of software to specific needs. Financial, technical and statistical applications can be met by "off the shelf" packages. With many years programming experience on small machines, the Apple II was chosen because it is probably the best microprocessor based computer in this particular field; its large memory and versatile disk system means that it is ideal as a business machine. Keen sell a full range of peripherals and printers and have gained respect from many computer companies, often being called on for advice on machines other than Apple.

L.P. ENTERPRISES — STAND 8 + 9

313 Kingston Road,
Ilford,
Essex IG1 1PJ
V. Coen — Tel: 01 - 553 1001

L.P. Enterprises imports and distributes books and magazines from a large number of American publishers. It is displaying and selling a complete current range of books and magazines, including magazine subscriptions. There are books geared specifically for business applications with some appropriate software, as well as some systems software; e.g. CP/M. There are books for the novice as well as for the person who already knows how to handle microcomputers, but wishes to make a fuller use of them. The magazines cover the spectrum of hardware, software, applications and development. The literature ranges between informative reference material to lighthearted observations of man and machine, plus all the intervening stages.

LIMROSE MICROPROCESSOR DIVISION — STAND 30

241 - 243 Manchester Road,
Northwich,
Cheshire, CW9 7NE
Dr. R. S. Raizada — Tel: 0606 - 41696/7

Limrose will be displaying the MICROTUTOR 8080, the LMC 6800 - 2 Microcomputer and accessories at their stand. Also on display will be CREED 73 5-Bit Code Teleprinter machine interfaced to the MICROTUTOR 8080. This item is of particular interest to personal computer users as it enables them to obtain a hard copy from a very inexpensive printer.
A low-cost Visual Display Unit, VDU 7000, will also be exhibited, together with Limrose' other computer educational products.

NEWBEAR COMPUTING STORE — STAND 13

Bone Lane, Newbury, Berks. RG14 5SH
T. W. Moore — Tel: 0635 46898

This organisation supports Microcomputing in general and stocks a wide range of items to achieve this aim:— The systems section support S-100 bus microcomputers with Cromemco Z-2 and all

SIRTON PRODUCTS (SP)

We specialise in the S.100 Bus System with 8080 or Z.80 CPU'S.

MAINFRAME
Desk Top, with power supply, motherboard & fan etc. £187.00
SIRTON Mainframe — due December
SIRTON V.D.U. Self Contained Unit
16 lines 64 characters, 1K RAM case, power supply and UHF modulator, with Reverse Video and Flash etc. £97.50

KEYBOARD KITS
53 key — Tri-mode ASCII output (without case) £47.00
53 key — Tri-mode ASCII output in black/white SIRTON case £59.00
SIRTON tough-type ASCII output, with case £38.50
SIRTON tough-type ASCII output, with additional features £50.00

BOARD KITS
8K RAM Board low power 450 n Sec. (21L02-1) £94.50
8K RAM Board low power 250 n Sec. (21L02-1) £119.50
8080 CPU Board with Vector Interrupt Circuit £72.50
8080 CPU Board with jump-on-reset £95.50
Z80 CPU Board 2 MHz, 2708 Monitor, power-on-jump £94.50
Z80 CPU Board 4 MHz, 2708 Monitor, power-on-jump £99.50
Z80 Bare Board £31.00
2708 EPROM (16K) for 2708 or 2716 EPROMS £47.00
2708 EPROM (16K) with 8 EPROMS (2708's) £92.50
2708 EPROM Board with programmer (8K) £118.50
8K EPROM/1K RAM, with RAM £75.50
Cassette Interface (with Monitor) — built £135.00
Serial/Parallel I/O Board, 2 Serial/1 Parallel 'Kansas City' Interface £94.50
Video Interface, 16 lines, 32 or 64 characters per line £98.50
Motherboard (13 slot with active terminations) £48.50

HARDWARE
S.100 edge connectors gold plated solder tail £5.45
Transformer Pri 110/240 V; sec. 8 V @ 10 Amp & 25 V CT @ 2 Amp £12.75
Bridge Rectifier with integral bracket rating 18 Amp £3.95
S.100 Proto-type board £15.00

INTEGRATED CIRCUITS
8080A CPU Chip 2 μ Sec £6.95
21L02 RAM 1Kx1 Bit Low Power 450 n Sec £1.20
21L02 RAM 1Kx1 Bit Low Power 250 n Sec £1.40
2708 EPROM 1Kx8 Bit 450 n Sec £7.50
8212 I/O PORT 8 Bit £3.10
2513 Character Generator, Upper Case (5 volt) £5.25
Z.80 Monitor £14.00

Apply for prices of ready-built and tested items. Please add 8% VAT.
WRITE OR PHONE FOR LATEST CATALOGUE.

SIRTON PRODUCTS
13 Warwick Road, Coulsdon, Surrey CR3 2EF.
Tel: 01-660 5617

Post & Packing
Keyboards: £1.00 each — Kits: 80p per kit
Transformer: £1.00 each — Hardware/IC's: 30p per order



BFD-1
£522.00 ASS.
BFD-2
£785.00 ASS.
BFD-3
£1045.00 ASS.



SOROC IQ.120
£699.00
ASS.



SWTPC 6800
LEVEL 2
£300.00
KIT

GROW YOUR SYSTEMS FROM

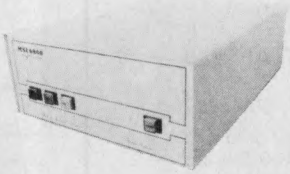
seed

CI.30/60
KIT £18.95
ASS. £25.00

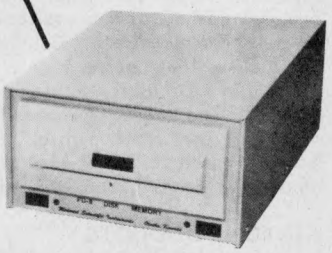
16K RAM £350
ASS.
8k RAM
£141.75
KIT



FULL SUPPORTING SOFTWARE
SUPER BASIC £48.00
COMPILER £185.00
EDITER/ASS. £36.00
ETC.



MSI 6800
£375.00 KIT
£565.00 ASS.



FD-8W £935.00
FD-8DW
£1640.00

Sole U.K. Distributor for M.S.I. and S.S.B.

STRUMECH ENGINEERING LTD.

Portland House, Coppice Side
Brownhills, Walsall, Staffs.
Phone: BRO 4321

the Comart products, together with Newbury Laboratories VDU's and DRI printers. For the '6800' our latest offspring "PANDA" has just arrived!! The Software and literature section has a unique range of 6800 and Z80 based software together with probably the largest range of computing books in the country. The Hardware components section has launched the Synertek VIM-1 as well as packaging its own kits as "Bearbags" which have now reached 15 in number. Many of these support the popular 77-68 6800 based hobbyist system. A considerable range of components specifically for microcomputer constructors is also available. These items are available by mail order or from our shops at Manchester and Newbury.

PELCO (ELECTRONICS) LTD — STAND 39

Enterprise House, 83/85 Western Road, Hove, Sussex
G. Dale-Smith — Tel: (0273) 722155

Pelco (Electronics) Limited are the exclusive distributors and representatives of Rockwell International microelectronic devices in the UK.
AIM 65 is the main feature on the stand. Developed by Rockwell as a personal computer and microprocessor teaching aid, AIM 65 is an enhanced version of KIM-1 using the same 6502 microprocessor but with 20 column printer, 20 character display and 54 key keyboard.

PERSONAL COMPUTERS LTD — STAND 18

18/19 Fish Street Hill, London EC3R 6BY
M. J. Sterland — Tel: 01-623 1434

The UK distributor for Apple Computers Inc.
The company will be showing its Apple II computer which is a very powerful 6502 microprocessor using Basic programming language with high-and-low resolution colour graphics facilities. The machine at the very beginning of its product life with spare input/output ports, ROM and RAM areas. Several cards to enhance the machine are already available.
The system is easy to use, reliable, simple to maintain and very well documented.
Applications for which the machine is being used include time-sharing and markets like education, engineering, scientific research, banking and to OEM computer companies.

RESEARCH MACHINES LTD — STAND 7

209 Cowley Road, Oxford
Mr. M. O'Regan — Tel: 0865 49793

The Research Machines 380Z exhibited, designed and manufactured in Oxford, is a general purpose microcomputer developed for professional users. The flexibility of the system and the range of software available has been of particular interest to those in Scientific Research and Education with a large number of systems in use in the field. Some of the peripherals offered for the 380Z are shown on the stand, and there are leaflets available covering different applications for the system.

SINTROM MICROSHOP — STAND 25

Based in Arkwright Road, Reading, The Sintrom Microshop welcomes callers to view a selection of microcomputers and peripherals on demonstration. Company policy is to hold stock of all products wherever possible.
A wide range of users are catered for, from the hobbyist through Scientific and Educational users to the business man requiring an inventory and stock control system. By carrying a carefully selected range of processors, floppy disks, kits, software and books, the Sintrom Microshop caters for all requirements.

SIRTON PRODUCTS — STAND 40

13 Warwick Road, Coulsdon, Surrey CR3 2EF
Mrs. M. Moon — Tel: 01-660 5617

We have on display an ASCII coded touch type keyboard and case. Both Z80 and 8080 CPU board kits for the S.100 Bus are featured together with supporting RAM, I/O and Mother boards. Just released will be a VDU interface kit complete with case etc., featuring 16 lines of 64 characters, reverse video, flashing characters, screen scroll and screen roll.
Mainframes will be available for S.100 system boards.

STAR DEVICES LIMITED — STAND 17

11 Winston Way,
Thatcham, Berks.
P. D. Stubley — Tel: Thatcham 68020

A British designed and built low cost QWERTY keyboard will be on display. The Standard unit offering the following features:—
7 bit parallel ASCII encoded output — Positive or negative strobe edges — Modified 2 key rollover + N key lockout — Audio tactile feedback — Cased — Automatic scan facility — LED's to show coding of selected character — Gold plated 0.1" pitch edge connector — Requires 5 Volts at only 200mA.
Optional extras are:— RS232 — Baud rate generator — internal generation of ± 12 Volts — Parity odd or even — 20 mA current loop — On board 5 Volt regulator — Active low outputs — Open collector outputs active high or low.

STRUMECH ENGINEERING LIMITED — STAND 54

Portland House, Coppice Side, Brownhills, Walsall, West Midlands
R. N. Hinton — Tel: B'hills 4321 Ext. 16

S.E.E.D. will be exhibiting many disc products for use with the M.S.I. 6800 processor along with printers and terminals including the new ACT-1 Keyboard. M.S.I. Processors are now available in two forms, a stand alone desk top system or for commercial applications rack mounted in a desk which can be customised to include our HD-8 10 megabyte disc drive. For other M.S.I. users a large range of hardware support in the form of RAM, EPROM, PROM/RAM, EXTENDER, WIRE WRAP boards and 2 prom programmers are also supplied to enhance your existing system. S.E.E.D. are sole U.K. distributors for M.S.I.

TANDBERG U.K. — STAND 47

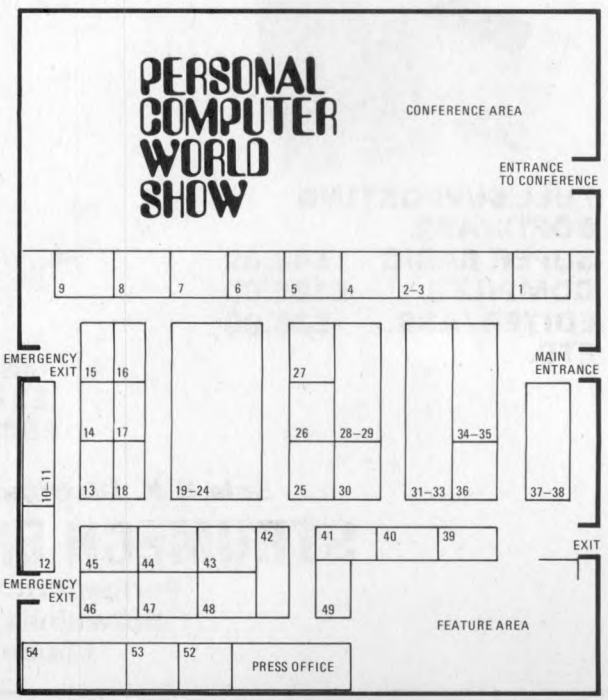
Mr. Cowing

Tandberg EC10 educational computer designed by Tandbergs Radiofabrikk of Norway has been introduced to meet the growing need for stand alone computer systems in Scandinavian schools.
The basic unit comprises a robust housing (VDU) with interchangeable boards. The CPU is an Intel 8080 microprocessor with configuration of boards to suit. Full size Diskette drive is built in. The screen measures 25 x 80 cm and the keyboard is the ECMA type. The power supply is the compensating type with overall protection and the language is extended BASIC. User Ram 35K. EC10 is complete with external sockets, V24 and 20 m/amp current loop, printer and time sharing sockets and card reader.
Baud rate is from 110-19,200.

TANDY CORPORATION — STAND 6

(Branch U.K.) Bilston Road, Wednesbury, West Midlands WS10 7JN
R. King — Tel: 021-556 6101

Designed and manufactured by Tandy Corporation, the TRS 80 microcomputer based on the Z80, is available in several combinations ranging from a 4K level 1 Basic up to 48K level 2 Basic. Storage facilities are available to 350K 'on line'. Peripherals available include an expansion interface, a line printer and a mini disc system (4 drive). Prices range from £499 to £4077 (inclusive of VAT). Basic systems comprising of 4K level one at £499 includes the CPU 12" video display cassette interface power supply and user manual. The units are available through any one of Tandy Corporation's 170 retail outlets in the UK. Further information is available from Tandy Corporation (Computer Division).



ASSEMBLY CODE PROGRAMMING FOR THE BEGINNER

Stephen Collins

Before I start explaining how, in my view, a beginner should go about writing assembly code programs, I would like to briefly explain why assembly code programming is sometimes necessary.

If you have not had any experience in writing assembly code programs, you may think, having learnt a high level of language such as BASIC or FORTRAN, that any problem can be solved using these languages and that low level languages are not really necessary. However, you must realise that computers do not directly understand BASIC, or any other high level language, but only the machine code for which they are designed. Therefore, high level languages have to be either interpreted by an interpreter, or translated into machine code by a compiler. All computers with the ability to accept and run programs written in a high level language have either a compiler or an interpreter residing in their main storage.

Because computers can only understand machine code programs, both compilers and interpreters have to be written in machine code; usually by a team of experienced programmers. However, machine code programming is very difficult and tedious, even for experienced programmers, and a compiler can take several man-years to write.

In order to eliminate the problem of writing programs directly in machine code (which involves a lot of decimal to binary conversions), assembly code programming was developed. Each machine code instruction is represented by an easily remembered mnemonic, such as ADD, SUB, JMP, etc., and the program is written using these mnemonics. Also, to avoid calculating offsets for jump instructions, symbolic labels are used so that the destination of jumps can easily be found.

Once the assembly code program is completed it is translated into machine code by a small program, usually written in machine code, called an assembler. The machine code version of the program can then be executed by the computer. The task of writing large machine code programs, such as compilers and interpreters, is made much easier by the use of assembly code programs which avoid many of the problems associated with their machine coded counterparts.

Having explained the purpose of assembly code programming, I should now explain how I began writing programs in assembly code. Part of my A-Level Computer-Science exam consisted of presenting an account of practical work on a programming project. I decided to write an interactive CESIL interpreter, in BASIC, so that lower sixth students could get some experience in interactive programming without having to learn a complicated language at the start of their course.

Having completed the program on a PDP 11/45 time-sharing system, I slightly modified the program for use on the college computer, an M6800 microprocessor, using a 12K BASIC interpreter. However, when I ran the program it was so slow that it would have been quicker to work out the problem by hand, instead of writing a program. It took about twenty seconds to check each CESIL line for syntax. In a simple CESIL program to print out the numbers from one to ten there was a lapse of approximately forty seconds between each number output.

In the hope of speeding up the program, I accepted the challenge of rewriting the complete program in assembly code, assembling it with the Co-resident Editor and Assembler available on the microprocessor.

So how should you go about writing assembly code programs for the first time? The first thing to do is to thoroughly familiarise yourself with the language to be used. Although this may seem obvious, it is more complex than just knowing the instruction set. You should know exactly what each instruction does, including which flags and registers are set by the outcome of each instruction. You should also know how conditional jumps are tested for, and how offsets are arranged. In addition to this, you should know how any internal registers, such as accumulators, index registers, stack pointers, program counters, etc., are arranged. This involves a knowledge of whether they are eight or sixteen bit registers, and whether numbers are stored in pure binary, binary coded decimal or in any other way.

Another important point that should be mentioned here is that any software supplied with the system should be studied. Many computer firms supply a large amount of software which is very useful when writing

assembly code programs. Therefore, if you first study this software, many routines, for example input and output routines, can simply be copied instead of rewriting them for each program.

However, you must ensure that you fully understand how the routines work. When using one of the routines of the monitor program on the college microprocessor, I did not realise that an accumulator was modified. Consequently, I could not discover the whereabouts of an error which had occurred while using this routine.

In brief then, you should thoroughly understand the language, have a rough idea of how the computer works and be familiar with any useful software before starting to program.

Once you are familiar with the language you can think about writing your first assembly code program. Initially, you should write a few short programs (of about ten to twenty instructions) to see exactly how the language works and to make clear anything not understood while reading the manuals. This also helps you to discover any important errors which you need to guard against when writing larger programs.

This was one of the mistakes that I made when I first started programming in assembly code. I started by writing the CESIL interpreter, which is a very difficult and complicated program for a beginner. Of course, when the program was finally completed I was faced with the task of debugging it. Having spent about five weeks dry-running flowcharts and debugging the program, all to no avail, I found the error, a very simple one, allowing me to continue and correct the remaining errors. Had I written a few shorter programs before tackling the interpreter, I am sure that this would not have occurred, since I would certainly have encountered the problem then, and known to guard against it when writing larger programs.

Having gained a little experience, you can start on a larger project. One word of warning here! DO NOT try to translate a program written in a high level language directly into a low level one. I tried this when writing the first version of CESIL. I had already written a program in BASIC and I thought it a good idea to translate it into assembly code. However, half of the routines turned out to be slightly modified BASIC interpreter routines, and consequently the first version was twice as long as it needed to be, and therefore very inefficient.

The first thing to do before writing a complete low level language program is to draw a flowchart. Many programmers look down on flowcharts as something producing more work and rarely use them. I, for one, very rarely draw flowcharts when programming in a high level language because I find that I can work out the flowchart in my head and then write the program from that. However, I soon found that this was not possible when writing programs in a low level language, since it is much more complicated. After several failures when writing the first routine of CESIL, I discovered that it was much easier to draw a flowchart first, and then to write the program from this.

But don't be put off by the thought that flowcharts take up a lot of time and use up reams of paper. I soon developed a shorthand method of flowcharting which, although only comprehensible to myself, took very little time to write and occupied little space on paper.

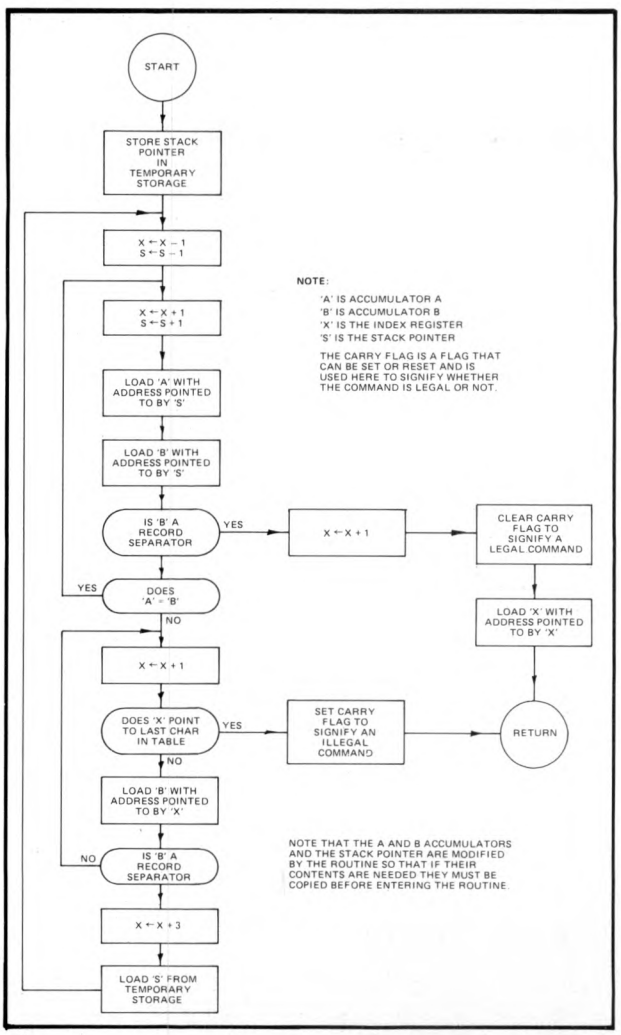
Having completed the flowcharts you can finally start writing the program in its initial form of separate subroutines. This was another method I discovered to facilitate debugging. I found that it was easier to write each subroutine separately, and debug it, rather than writing the complete program in one fell swoop and then trying to debug the lot. Of course, subroutines are not much use on their own and very rarely do anything when isolated from the main program. Consequently, I had to write small **driver programs** for each program in order to test them. Although this involves more work, it is far easier, in the long run, to complete the program this way; the amount of coding that has to be debugged is much smaller so the errors can be found more rapidly.

I found that this was the case when trying to debug the arithmetic routines for CESIL. I had written a small driver program to test the actual addition, subtraction, multiplication and division routines and I knew that they worked correctly. However, when I wrote a small CESIL program to subtract two numbers an incorrect result was produced. Knowing that the subtraction routine was not in error, I concluded that the error was either in the decimal to binary conversion routine, or in the binary to decimal one, and I soon corrected it.

One important point to remember when writing each routine separately is to realise that subroutines usually change various registers if these are used for temporary storage. Therefore, you must ensure that any registers used for this purpose are restored to their original state before an exit is made from the routine. If this is not done, incorrect data will be fed into other routines.

Finally, once all the separate routines have been written, dry-run and fully debugged, they can all be joined together, along with the main program, which can then be tested as a whole. Of course, a few more errors will still have to be corrected, but it should be very easy to finish the program.

So, having finished the program and thoroughly tested it, a final listing should be obtained. The program must then be fully documented, together with listings, flowcharts and other relevant information that is needed for another person to be able to operate, understand and perhaps modify the program, without having to spend a great deal of time running through it.



Although I cannot give examples of all the points mentioned above, I will give an account of how I developed one of the routines for CESIL.

The problem was to write a routine to decode commands and to load the index register with the address to which control should be transferred for each instruction.

The basic algorithm is to compare the first character of the string to be decoded with the first character of each string in the lookup table, in turn, until a match is made. Then the second, third and remaining characters are compared until either a record separator is read from the lookup table, signifying that a complete string has been recognised, or a mismatch is found. If this is the case the next string in the lookup table is examined and the process repeated.

Assume that, on entering the routine, the stack pointer points to the beginning of the string to be decoded and the index register to the start of the lookup table.

The flowchart shown is the one used when I wrote the routine for CESIL, although I have rewritten it in a normal style since you will probably not understand my shorthand method and will want to develop your own ways for flowcharting.

(PCW Stack pointer? Index register? Program Counter? Consultant Patrick Sutton will unravel some of these mysteries in a forthcoming issue. PCW)

Once the flowchart was completed and fully debugged I wrote the actual program.

A listing of the program is shown below. The driver program has not been included in the listing because it only consists of a single loop to input the string to be decoded and is not necessary to understand the routine.

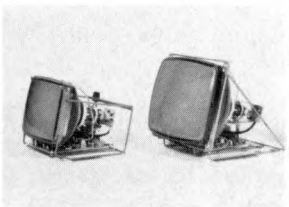
Before I finish, I would like to make it clear that the method I have described above for going about writing your first assembly code programs is certainly not the best method. I have only had a single year's experience in writing this type of program, using a small microprocessor, and so I have only scraped the surface of a very large topic. However, I have already encountered and successfully overcome a number of major problems that beginners are sure to encounter themselves. I hope that the tips in this article will help and encourage other programmers who have just started, or are thinking of starting, programming in assembly code.

```
00050                    *
00060                    *
00070                    ** DECODE ROUTINE
00080                    *
00090                    *MODIFIES A,B AND S
00100                    *
00110 1000 BF 0100       STS    TEMP     SAVE CONTENTS OF S.P.
00120 1003 09            DEX             DECREMENT X
00130 1004 34       DCODE1 DES             DECREMENT S
00140 1005 08       DCODE2 INX             INCREMENT X
00150 1006 32              PUL A          INCREMENTS S AND LOADS A
00160 1007 E6 00           LDA B  0,X     LOAD B WITH 'X'
00170 1009 C1 1E           CMP B  #$1E    IS IT A RECORD SEPARATOR
00180 100B 27 16           BEQ    DCODE4  IF SO,JUMP
00190 100D 11              CBA             IF NOT,COMPARE ACCUMULATORS
00200 100E 27 F5           BEQ    DCODE2  IF EQUAL,TRY NEXT CHARACTERS
00210 1010 08       DCODE3 INX
00220 1011 8C 032F         CPX    #COMEND  X POINTS TO LAST CHAR IN TABLE?
00230 1014 27 12           BEQ    DCODE5   YES,SO EXIT
00240 1016 E6 00           LDA B  0,X      LOAD B
00250 1018 C1 1E           CMP B  #$1E     IS IT A SEPARATOR
00260 101A 26 F4           BNE    DCODE3   IF NOT,TRY NEXT CHAR.
00270 101C 08              INX             MOVE POINTER TO NEXT
00280 101D 08              INX             ITEM IN TABLE
00290 101E BE 0100         LDS    TEMP     RELOAD S.P.
00300 1021 20 E1           BRA    DCODE1   REPEAT
00310 1023 08       DCODE4 INX             A COMMAND HAS BEEN RECOGNISED
00320 1024 0C              CLC             CLEAR CARRY TO SIGNIFY THIS
00330 1025 EE 00           LDX    0,X      LOAD X WITH ADDRESS
00340 1027 39              RTS             RETURN
00350 1028 0D       DCODE5 SEC             NO MATCH,SO SET CARRY
00360 1029 39              RTS             RETURN
00370                    *
00380                    *
00390                    *
00400                    *
```


TV MONITOR KITS

VISIONKITS

gives you professional TV equipment in kit form for all data display and picture applications, providing far superior results to converted TV receivers.



Top quality monitors exactly as produced by one of today's leading manufacturers. All parts plus comprehensive assembly instructions supplied, along with full application notes, 9" or 12" tube plus chassis kits are available with a separate printed circuit module kit to fit other size of tube.

SPECIFICATION:

Video – Composite 1V ± 6dB Input impedance 10K ohms
Power – + 12 volts stabilised Current 1 amp
System – 625 lines 50 fields and 525 lines 60 fields
Video response to 10MHz
Operating temperature – 0 – 50°C

PRICE: (Including VAT and inland postage and packing)

9" tube and chassis kit	£45.00
12" tube and chassis kit	£45.00
P.C. module kit	£55.00
Ready-built 9" monitor	£150.00
Ready-built 12" monitor	£150.00
Mains power supply for 12" monitor (built)	£40.00

Please allow 21 days for delivery
Send s.a.e. for copy of application notes only
Mail order only.

VISIONKITS

9 Claymill Road, Leicester LE4 7JJ

commodore

PET

personal computers
for home and business use

* * * * * * *

software available:
payroll, stock control, ledger, invoicing, mailing lists . . . . games.

* * * * * * *

See the PET and our range of programmable calculators at

Sumlock
Manchester

Royal London House, 196 Deansgate
Manchester M3 3WE. 061-228 3507

Binary Finery

Extensions to the Minmon

Neil Harrison

The MINMON is a 256 byte monitor program for Z80 based micros which was published in PCW Vol 1 No 1. It was written to provide the minimum facilities needed for machine code programming and to fit into a cheap 1702A EPROM. In view of the tumbling price of larger EPROMS such as the 2708 the 256 byte limit is far less important and and expanded form of the monitor can be considered. The MINMON provides two commands; the 'E' command to examine and alter memory locations and the 'G' or 'goto' command to start program execution at a particular address. The extensions to the monitor detailed in this article add four new commands, three for program storage and retrieval on tape and one to examine and alter Z80 I/O ports. In all, this adds a further 306 bytes to the length of the original program and the listing in Figure 1 has been assembled to start at 100 Hex, immediately after the MINMON.

Tape I/O

The tape storage routines use a data format which is almost universal in the world of 8080s and Z80s, the Intel format. Bytes of data are stored in hexadecimal as two ASCII characters and grouped together in blocks complete with a load address and a check byte. Figure 1 shows the structure of one block of data (a typical dump would consist of a number of such blocks). The check byte is formed by adding all the bytes in the block (except the 's') and negating the result. This means that when the block is read in the sum of all the bytes *including* the check byte will be zero if no errors have occurred.

A special block is used to indicate that the 'end of file' has been reached. In this block, shown in Figure 2, the data byte count is zero, and the address bytes contain an optional execution address for the program.

The contents of memory are stored on tape using the Dump command:—

> Dxxxx yyyy (carriage return)

User input is underlined; xxxx is the start address and yyyy the end address in hexadecimal of the memory area to be dumped. Nothing is written to tape until a Carriage Return is typed to allow time to start a tape punch or cassette recorder. When the dump is complete control returns to the monitor ready for a new command.

When the complete program is stored on tape an 'end of file' block should be written using the 'Z' command:—

Zxxxx (carriage return)

where xxxx is the optional program execution address in hexadecimal. If this is omitted zeros are written into the address bytes of the block.

Programs are loaded into memory from tape by simply typing the letter 'L' immediately after the tape reader or cassette player has been started. When the pro-

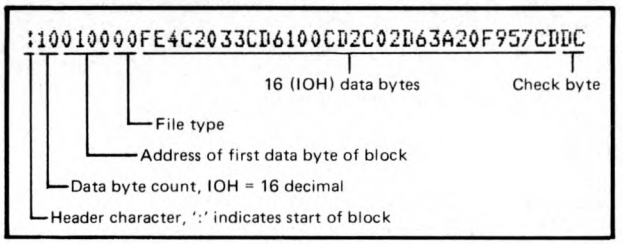


Figure 1 Typical data block

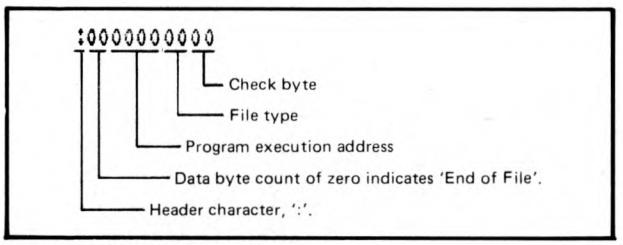


Figure 2 End of File block

gram has been successfully loaded the execution address is printed on the console in hexadecimal and control returns to the monitor. If an error is found an asterisk "*" is printed on the console and the load terminated. If this occurs it is only necessary to try loading again from the beginning of the block and not from the beginning of the whole program since previous blocks have already loaded correctly.

Playing with Ports

The fourth command in the MINMON extensions gives access to the Z80 I/O ports in much the same way that the Examine command does for memory:—

> Pxx
xx dd *nn*

'xx' defines one of the 256 Z80 ports in hexadecimal, 'dd' the data input from the port and 'nn' the user's reply which will be one of the following:—

1) Two valid hexadecimal characters will be used as data to be output to port 'xx'.
2) A 'space' character will print the next port number and data input from the port.
3) A minus character, '—' will print the previous port and input data.
4) Any other character will restart the monitor.

Example:—

>PO4	(examine data input from port 4)
04 DE (space)	(examine next port)
05 03 FF	(output 'FF' to port 5)
06 72 (space)	(examine next port)
07 8E —	(examine previous port)
06 72 (carriage return)	(return to monitor)

Adding the extensions to MINMON

Once the extra code shown in Listing 1 is written into memory above MINMON only two bytes of the original program need changing to make the new commands available. The instruction at address 004B (see original MINMON listing PCW Vol 1 No 1 page 27), changes from:—

004B C2 OB 00 JP NZ, BEGIN

to:—

004B C2 00 01 JP NZ, LOADER

I/O Routines

The tape character I/O routines PCHAR and RI in Listing 1, are included as examples and will almost certainly need changing to suit individual systems. Both routines alter only the accumulator and flags.

PCHAR outputs the data in the accumulator to the tape output device. It is written for a device which uses port 7 for data output and bit 7 of port 6 as a 'ready' bit.

RI gets data from the tape input device into the accumulator. Port 7 is used for data input and bit 1 of port 6 indicates 'data available'.

MINMON routines used by the Extensions

These extensions to the MINMON use a number of routines and addresses in the original monitor code. For those who wish to use all or part of the routines in Listing 1 without the MINMON here is a list of the subroutines and addresses and what they do.

Name	Address	Function
BEGIN	000B H	Entry to the MINMON command processor.
ERROR	005A H	Prints an asterisk '*' on the console and returns to the command processor.
CRLF	0061 H	Prints a new line (carriage return, line feed) on the console.
SPACE	006C H	Prints a 'space' on the console.
GXN	0074 H	Gets a hexadecimal character from the keyboard and returns it in the 4 least significant bits of A. If the character is not valid hexadecimal then returns with carry set and the character in A.
GXB	008A H	Gets 2 hexadecimal characters from the keyboard and returns them as a byte in A. If a non hexadecimal character is found the carry flag is set and the character returned in A.
GXW	009C H	Gets 4 hexadecimal characters from the keyboard and returns a 16 bit number in the HL register pair. If a non hexadecimal character is found the carry flag is set and the character returned in A.
TXB	00B4 H	Prints the contents of A as two hexadecimal characters on the console.
TXW	00CD H	Prints the contents of HL as 4 hexadecimal characters on the console.
CHROUT	00E8 H	Outputs an ASCII character from A to the console.
CHRIN	00F5 H	Gets an ASCII character from the keyboard to A.

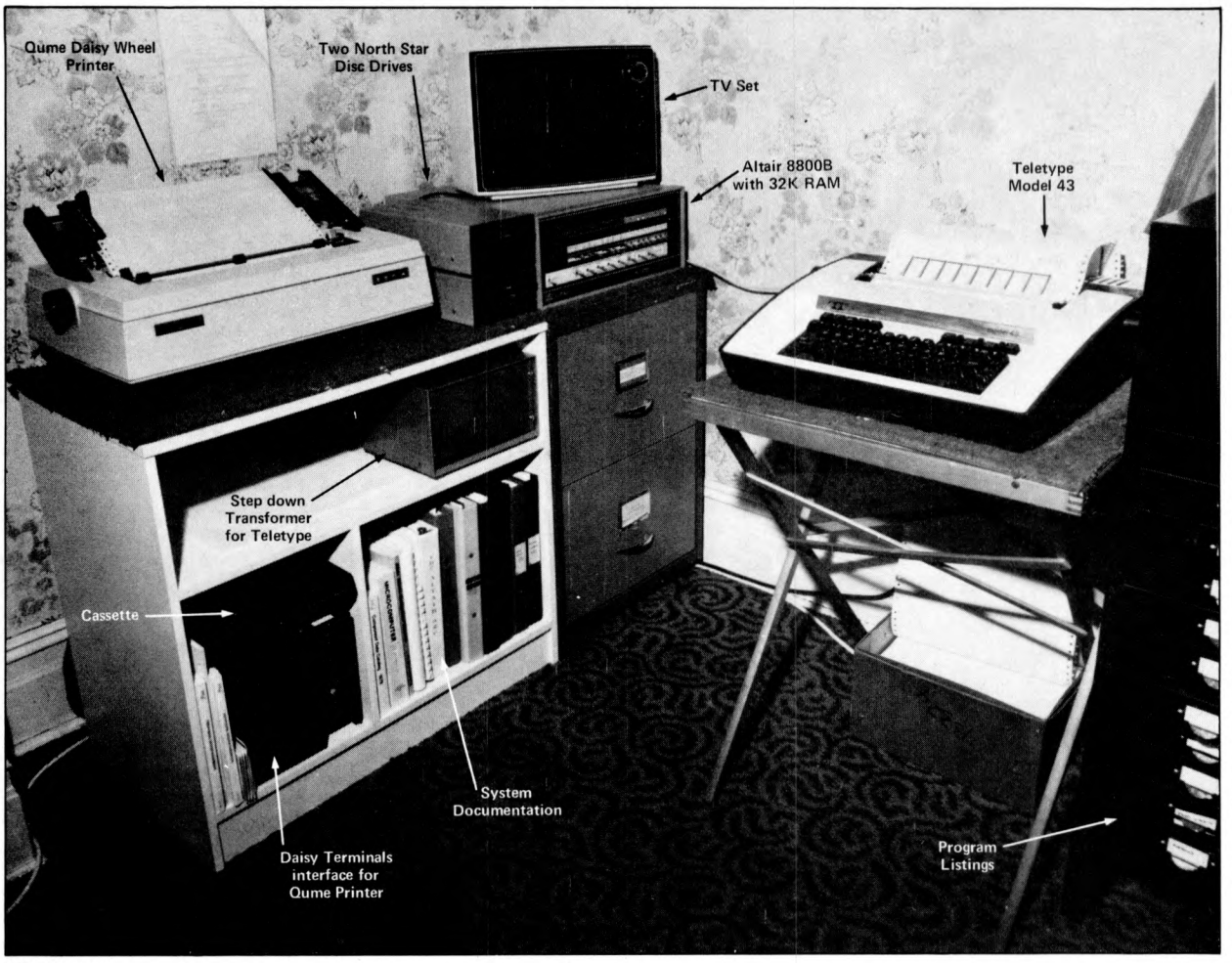
Further expansion

Why stop at six commands? More can always be added simply by changing the address in the Jump instruction at location 0202 H to the start of the next lot of extensions rather than BEGIN. The last command in your new routines should jump back to BEGIN when you've finished.

```
:10010000FE4C2033CD6100CD2C02D63A20F957CDDC
:10011000D40147CDD40167CDD4016F78B72812CD73
:10012000D401CDD401772310F9CDD40128D9C35AF5
:1001300000CDCD00C30B00FE442037CD9C0038EE2F
:10014000EBCD6C00CD9C0038E5CDCB01CD6100B787
:10015000ED52EB06107AB720097BFE103004B72869
:10016000D343CD91017ECDAF01231B10F8CDC60145
:1001700018E1FE5AC2EE01CD9C00300ACDCB011826
:1001800006D60D20A9676F0600CD9101CDC60118D6
:10019000A33E0DCD21023E0ACD21023E3ACD2102E1
:1001A0000E0078CDAF017CCDAF017DCDAF01AFF5B5
:1001B000814FF1F51F1F1F1FCDBC01F1E60FC69047
:1001C00027CE4027185B79ED4418E4CDF500FE0DED
:1001D000C8C35A00CDE501171717175F7DE501B366
:1001E0005F82577BC9CD2C02CD7700D018E3FE503B
:1001F000C20B00CD8A0038D94FCD610079CDB40053
:10020000CD6C00ED78CDB400CD6C00CD8A00380502
:10021000ED790C18E4FE2028F9FE2DC20B000D1814
:10022000D8F5DB06E68020F9F1D307C9DB06E60145
:0502300020FADB07C904
:0000000000
```

```
                        0001 ;*************************************************************
                        0002 ;
                        0003 ;       EXTENSIONS TO THE Z80 MINIMUM MONITOR PUBLISHED
                        0004 ;       IN PCW VOL 1 NO 1.
                        0005 ;
                        0006 ;       REFERENCES TO MINMON ROUTINES:-
                        0007 ;
(000B)                  0008 BEGIN:  EQU     000BH          ;START OF MONITOR
(005A)                  0009 ERROR:  EQU     005AH          ;ERROR ROUTINE
(0061)                  0010 CRLF:   EQU     0061H          ;CR & LF ON CONSOLE
(006C)                  0011 SPACE:  EQU     006CH          ;SPACE ON CONSOLE
(0074)                  0012 GXN:    EQU     0074H          ;GET A HEX NUMBER INTO A
(008A)                  0013 GXB:    EQU     008AH          ;GET A HEX BYTE INTO A
(009C)                  0014 GXW:    EQU     009CH          ;GET A HEX WORD INTO HL
(00B4)                  0015 TXB:    EQU     00B4H          ;TYPE A IN HEX
(00CD)                  0016 TXW:    EQU     00CDH          ;TYPE HL IN HEX
(00E8)                  0017 CHROUT: EQU     00E8H          ;CHARACTER OUTPUT ROUTINE
(00F5)                  0018 CHRIN:  EQU     00F5H          ;CHARACTER INPUT ROUTINE
0000                    0019         ORG     100H
                        0020 ;
                        0021 ;       L - LOADS AN INTEL FORMAT FILE FROM TAPE
                        0022 ;
0100 FE4C               0023 LOADER: CP      'L'            ;LOAD COMMAND?
0102 2033               0024         JR      NZ,DUMP        ;NO, SKIP TO 'DUMP'
0104 CD6100             0025         CALL    CRLF           ;NEW LINE
0107 CD2C02             0026 LOAD1:  CALL    RI             ;GET A CHARACTER FROM TAPE
010A D63A               0027         SUB     ':'            ;LOOK FOR A START CHAR
010C 20F9               0028         JR      NZ,LOAD1       ;KEEP LOOKING TIL WE GET ONE
010E 57                 0029         LD      D,A            ;ZERO THE CHECKSUM
010F CDD401             0030         CALL    RBYTE          ;GET BYTE COUNT INTO A
0112 47                 0031         LD      B,A            ;PUT BYTE COUNT INTO B
0113 CDD401             0032         CALL    RBYTE          ;GET HI BYTE OF ADDR
0116 67                 0033         LD      H,A            ;PUT IT IN H
0117 CDD401             0034         CALL    RBYTE          ;GET LO BYTE
011A 6F                 0035         LD      L,A            ;PUT IT IN L
011B 78                 0036         LD      A,B            ;GET BYTE COUNT
011C B7                 0037         OR      A              ;SET FLAGS
011D 2812               0038         JR      Z,END          ;IF Z, END OF FILE
011F CDD401             0039         CALL    RBYTE          ;IGNORE FILE TYPE
0122 CDD401             0040 LOOP:   CALL    RBYTE          ;GET A DATA BYTE
0125 77                 0041         LD      (HL),A         ;PUT IT IN MEMORY
0126 23                 0042         INC     HL             ;UP THE POINTER
0127 10F9               0043         DJNZ    LOOP           ;LOOP TIL WE'RE DONE
0129 CDD401             0044         CALL    RBYTE          ;GET THE CHECKSUM
012C 28D9               0045         JR      Z,LOAD1        ;0 MEANS ALL'S WELL
012E C35A00             0046 ERROR1: JP      ERROR          ;BAD NEWS
0131 CDCD00             0047 END:    CALL    TXW            ;TYPE LAST ADDRESS
0134 C30B00             0048 BEGIN1: JP      BEGIN          ;RESTART MINMON
                        0049 ;
                        0050 ;       D - DUMPS MEMORY TO TAPE IN INTEL FORMAT
                        0051 ;
                        0052 ;       D<FIRST ADDR.> <LAST ADDR.> (CR)
                        0053 ;
0137 FE44               0054 DUMP:   CP      'D'            ;DUMP COMMAND?
0139 2037               0055         JR      NZ,ZEND        ;NO, SKIP TO EOF CMD.
013B CD9C00             0056         CALL    GXW            ;GET FIRST ADDRESS
013E 38EE               0057         JR      C,ERROR1       ;ANY ERRORS?
0140 EB                 0058         EX      DE,HL          ;PUT FIRST IN DE
0141 CD6C00             0059         CALL    SPACE          ;SPACE OVER
0144 CD9C00             0060         CALL    GXW            ;GET LAST ADDRESS
0147 38E5               0061         JR      C,ERROR1       ;ANY ERRORS
0149 CDCB01             0062         CALL    CRWT           ;WAIT FOR A CR
014C CD6100             0063         CALL    CRLF           ;NEW LINE
014F B7                 0064         OR      A              ;CLEAR CARRY
0150 ED52               0065         SBC     HL,DE          ;CALC. BYTE COUNT
0152 EB                 0066         EX      DE,HL          ;FIRST IN HL, COUNT IN DE
0153 0610               0067 BLOCK:  LD      B,10H          ;B=FULL BLOCK COUNT
0155 7A                 0068         LD      A,D            ;GET HI BYTE OF COUNT
0156 B7                 0069         OR      A              ;SET FLAGS
0157 2009               0070         JR      NZ,BLOCK1      ;IF NZ, FULL BLOCK
0159 7B                 0071         LD      A,E            ;GET LO BYTE OF COUNT
015A FE10               0072         CP      10H            ;ENOUGH FOR FULL BLK?
015C 3004               0073         JR      NC,BLOCK1      ;DO A FULL BLOCK
015E B7                 0074         OR      A              ;SET FLAGS
015F 28D3               0075         JR      Z,BEGIN1       ;IF 0, WE'VE FINISHED
0161 43                 0076         LD      B,E            ;PART BLOCK COUNT
0162 CD9101             0077 BLOCK1: CALL    PHEAD          ;PUNCH HEADER
0165 7E                 0078 DATALP: LD      A,(HL)         ;GET DATA FROM MEMORY
0166 CDAF01             0079         CALL    PBYTE          ;PUNCH IT IN HEX
0169 23                 0080         INC     HL             ;UPDATE POINTER
016A 1B                 0081         DEC     DE             ;REDUCE BYTE COUNT
016B 10F8               0082         DJNZ    DATALP         ;LOOP TIL DONE
016D CDC601             0083         CALL    PCRC           ;PUNCH CRC & CR/LF
0170 18E1               0084         JR      BLOCK          ;ROUND AGAIN
                        0085 ;
                        0086 ;       Z - END OF FILE MARK, CONTAINS AN OPTIONAL
                        0087 ;       PROGRAM EXECUTION ADDRESS:-
                        0088 ;
                        0089 ;       Z(CR)  EOF WITH EX ADDR = 0000
                        0090 ;       Z<EX ADDR> (CR)
                        0091 ;
0172 FE5A               0092 ZEND:   CP      'Z'            ;IS IT Z COMMAND
0174 C2EE01             0093         JP      NZ,PORT        ;NO, TRY PORT
0177 CD9C00             0094         CALL    GXW            ;GET OPTIONAL ADDR.
017A 3805               0095         JR      C,NOADDR       ;C MEANS NO VALID ADDR
017C CDCB01             0096         CALL    CRWT           ;WAIT FOR A (CR)
017F 1806               0097         JR      ZBLOCK         ;WRITE THE BLOCK
0181 D60D               0098 NOADDR: SUB     0DH            ;WAS IT A (CR)?
0183 20A9               0099         JR      NZ,ERROR1      ;IF NOT, ERROR
0185 67                 0100         LD      H,A            ;FREE 0 INTO H
0186 6F                 0101         LD      L,A            ;FREE 0 INTO L
0187 0600               0102 ZBLOCK: LD      B,0            ;ZERO BYTE COUNT
0189 CD9101             0103         CALL    PHEAD          ;PUNCH THE HEADER
018C CDC601             0104         CALL    PCRC           ;PUNCH CRC & CR/LF
018F 18A3               0105         JR      BEGIN1         ;RESTART THE MINMON
                        0106 ;
                        0107 ;       PHEAD - PUNCH AN INTEL FORMAT BLOCK HEADER
                        0108 ;
0191 3E0D               0109 PHEAD:  LD      A,0DH          ;CARR RET
0193 CD2102             0110         CALL    PCHAR
0196 3E0A               0111         LD      A,0AH          ;LINE FEED
0198 CD2102             0112         CALL    PCHAR
019B 3E3A               0113         LD      A,':'          ;HEADER BYTE
019D CD2102             0114         CALL    PCHAR          ;PUNCH IT
01A0 0E00               0115         LD      C,0            ;CLEAR CRC
01A2 78                 0116         LD      A,B            ;GET BYTE COUNT
01A3 CDAF01             0117         CALL    PBYTE          ;PUNCH IT IN HEX
01A6 7C                 0118         LD      A,H            ;GET HI BYTE OF ADDRESS
01A7 CDAF01             0119         CALL    PBYTE          ;PUNCH IT IN HEX
01AA 7D                 0120         LD      A,L            ;GET LO BYTE OF ADDRESS
01AB CDAF01             0121         CALL    PBYTE          ;PUNCH IT IN HEX
01AE AF                 0122         XOR     A              ;FILE TYPE = 0
```

```
                        0123 ;
                        0124 ;       PBYTE - PUNCH A BYTE FROM A IN HEX
                        0125 ;
01AF F5                 0126 PBYTE:  PUSH    AF              ;SAVE A & F
01B0 81                 0127         ADD     C               ;UPDATE CHECKSUM
01B1 4F                 0128         LD      C,A             ;PUT IT BACK IN C
01B2 F1                 0129         POP     AF              ;RESTORE A & F
01B3 F5                 0130         PUSH    AF              ;AND SAVE AGAIN
01B4 1F                 0131         RRA                     ;ROTATE TOP
01B5 1F                 0132         RRA                     ; FOUR BITS
01B6 1F                 0133         RRA                     ;  DOWN TO
01B7 1F                 0134         RRA                     ;    THE BOTTOM
01B8 CDBC01             0135         CALL    PNIB            ;PUNCH AS HEX CHAR
01BB F1                 0136         POP     AF              ;GET A & F BACK
                        0137 ;
                        0138 ;       PNIB - PUNCHES BOTTOM 4 BITS OF A AS A HEX CHAR.
                        0139 ;
01BC E60F               0140 PNIB:   AND     0FH             ;LOSE TOP 4 BITS
01BE C690               0141         ADD     90H
01C0 27                 0142         DAA                     ;INTELS DAA TRICK
01C1 CE40               0143         ADC     40H
01C3 27                 0144         DAA                     ;AND AGAIN
01C4 185B               0145         JR      PCHAR           ;PUNCH IT
                        0146 ;
                        0147 ;       PCRC PUNCHES THE CRC BYTE
                        0148 ;
01C6 79                 0149 PCRC:   LD      A,C             ;GET CRC
01C7 ED44               0150         NEG                     ;NEGATE IT
01C9 18E4               0151         JR      PBYTE           ;PUNCH IT IN HEX
                        0152 ;
                        0153 ;       WAITS FOR A (CR) FROM THE KEYBOARD. ANY OTHER
                        0154 ;       CHARACTER CAUSES AN ERROR.
                        0155 ;
01CB CDF500             0156 CRWT:   CALL    CHRIN           ;GET A CHARACTER
01CE FE0D               0157         CP      0DH             ;IS IT CR?
01D0 C8                 0158         RET     Z               ;RETURN IF SO
01D1 C35A00             0159 ERROR2: JP      ERROR           ;OTHERWISE ERROR
                        0160 ;
                        0161 ;       READS TWO HEX CHARS FROM TAPE AND CONVERTS
                        0162 ;       THEM TO A BYTE IN A AND UPDATES THE CHECKSUM IN D
                        0163 ;
01D4 CDE501             0164 RBYTE:  CALL    NIBBLE          ;GET 4 BITS IN A
01D7 17                 0165         RLA                     ;PUT THEM
01D8 17                 0166         RLA                     ;  AT THE
01D9 17                 0167         RLA                     ;    TOP OF A.
01DA 17                 0168         RLA
01DB 5F                 0169         LD      E,A             ;SAVE IN E
01DC CDE501             0170         CALL    NIBBLE          ;GET 4 MORE BITS
01DF B3                 0171         OR      E               ;MAKE A BYTE
01E0 5F                 0172         LD      E,A             ;SAVE IT IN E
01E1 82                 0173         ADD     D               ;NEW CHECKSUM
01E2 57                 0174         LD      D,A             ;BACK INTO D
01E3 7B                 0175         LD      A,E             ;RESTORE OUT BYTE
01E4 C9                 0176         RET
                        0177 ;
                        0178 ;       GETS A CHARACTER FROM THE READER AND
                        0179 ;       CONVERTS IT TO BINARY IN THE BOTTOM FOUR
                        0180 ;       BITS OF A.
                        0181 ;
01E5 CD2C02             0182 NIBBLE: CALL    RI              ;GET A CHAR
01E8 CD7700             0183         CALL    GXN+3           ;CONVERT IN MINMON
01EB D0                 0184         RET     NC              ;RETURN IF NO ERROR
01EC 18E3               0185         JR      ERROR2          ;ITS AN ERROR
                        0186 ;
                        0187 ;       P -  ALLOWS YOU TO EXAMINE & CHANGE I/O PORTS
                        0188 ;
                        0189 ;       P<PORT> (CR)
                        0190 ;
01EE FE50               0191 PORT:   CP      'P'             ;PORT COMMAND?
01F0 C20B00             0192         JP      NZ,BEGIN        ;IF NOT RESTART MINMON
01F3 CD8A00             0193         CALL    GXB             ;GET PORT NUMBER
01F6 38D9               0194         JR      C,ERROR2        ;ANY ERRORS?
01F8 4F                 0195         LD      C,A             ;PORT NUMBER INTO C
01F9 CD6100             0196 PORT1:  CALL    CRLF            ;NEW LINE
01FC 79                 0197         LD      A,C             ;GET PORT NUMBER
01FD CDB400             0198         CALL    TXB             ;TYPE IT
0200 CD6C00             0199         CALL    SPACE           ;SPACE OVER
0203 ED78               0200         IN      A,(C)           ;GET PORT DATA
0205 CDB400             0201         CALL    TXB             ;TYPE IT
0208 CD6C00             0202         CALL    SPACE           ;SPACE OVER
020B CD8A00             0203         CALL    GXB             ;GET DATA OR CMD.
020E 3805               0204         JR      C,PORT2         ;IF C, COMMAND
0210 ED79               0205         OUT     (C),A           ;OUTPUT DATA
0212 0C                 0206 NEXTP:  INC     C               ;NEXT PORT.
0213 18E4               0207         JR      PORT1
0215 FE20               0208 PORT2:  CP      ' '             ;IS IT SPACE?
0217 28F9               0209         JR      Z,NEXTP         ;IF SO DO NEXT
0219 FE2D               0210         CP      '-'             ;IS IT '-'?
021B C20B00             0211         JP      NZ,BEGIN        ;IF NOT RESTART MINMON
021E 0D                 0212         DEC     C               ;PREVIOUS PORT
021F 18D8               0213         JR      PORT1
                        0214 ;
                        0215 ;       THE I/O ROUTINES PCHAR AND RI WILL ALMOST
                        0216 ;       CERTAINLY NEED CHANGING FOR YOUR SYSTEM.
                        0217 ;
                        0218 ;       'PCHAR' SENDS A CHARACTER TO THE TAPE
                        0219 ;       OUTPUT DEVICE.
                        0220 ;
                        0221 ;       THIS ROUTINE IS FOR A TAPE O/P DEVICE WITH DATA
                        0222 ;       ON PORT 7 & AN ACTIVE LOW READY BIT IN PORT 6,
                        0223 ;       BIT 7. ONLY A & FLAGS ARE CHANGED.
                        0224 ;
0221 F5                 0225 PCHAR:  PUSH    AF              ;SAVE THE CHARACTER
0222 DB06               0226 PCHAR1: IN      A,06            ;GET STATUS PORT
0224 E680               0227         AND     80H             ;TEST READY BIT
0226 20F9               0228         JR      NZ,PCHAR        ;LOOP TIL READY
0228 F1                 0229         POP     AF              ;RESTORE THE CHARACTER
0229 D307               0230         OUT     07,A            ;SEND IT
022B C9                 0231         RET
                        0232 ;
                        0233 ;       'RI' GETS A CHARACTER FROM THE TAPE INPUT
                        0234 ;
                        0235 ;       THIS ROUTINE IS FOR A DEVICE WITH DATA ON
                        0236 ;       PORT 7 AND AN ACTIVE LOW READY BIT IN PORT
                        0237 ;       6, BIT 1. ONLY A & FLAGS ARE CHANGED.
                        0238 ;
022C DB06               0239 RI:     IN      A,06            ;GET STATUS PORT
022E E601               0240         AND     01H             ;TEST READY BIT
0230 20FA               0241         JR      NZ,RI           ;LOOP TIL READY
0232 DB07               0242         IN      A,07            ;GET THE DATA
0234 C9                 0243         RET
0235                    0244         END

0000 ERRORS

END OF ASSEMBLY
```



The System
The note on the wall contains a list of instructions on how to switch on and run the computer. This is for the benefit of Brian Crank's children, should they wish to play games in his absence.

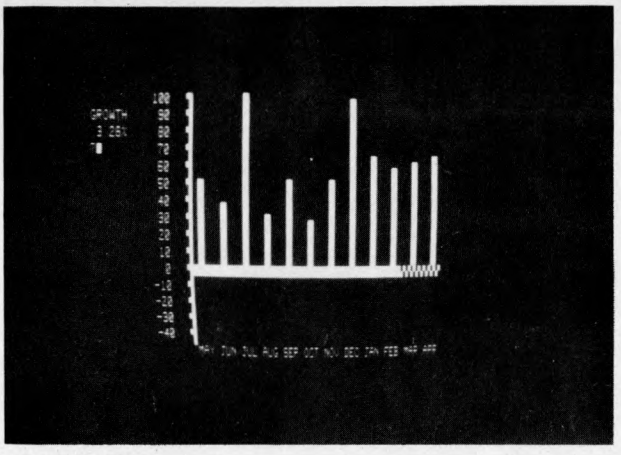
Boris Sedacca

PCW SMALL BUSINESS CASE STUDY

The Company: Brian Crank Associates
The Business: Technical public relations
The Background: Brian Crank Associates was formed three years ago. Growth has since been steady and the bulk of the company's turnover comes from about half a dozen large accounts.

The System:
Altair 8800B microcomputer with 40K RAM
Special interfaces (see diagram)
Twin North Star mini-floppy disc drives
Teletype 43
Qume 40 Daisywheel printer 40 cps and controller
Television and Polymorphic VTI board for video output
Cassette drive
A similar configuration at current market prices will cost in the region of £7,000 (Compelec Electronics are sole UK agents for MITS Altair). Brian Crank assembled the machine himself from a kit and added peripherals as the system developed. The audio cassette machine might appear somewhat superfluous and, indeed, this is now rarely used. However, it was invaluable in the early stages because in the beginning, it was the only bulk storage device in the system.

The Altair is something of a legend, it is the forefunner of personal computing. It is manufactured in the USA by a company called MITS which has now been taken over by another company called Pertec, a manufacturer of magnetic storage peripherals. This takeover has been beneficial because the Altair can now be interfaced to 10MB hard disc drive, (not in DMA mode however) which makes it particularly attractive to a business user whose storage requirements need



Sample Bar Chart

upgrading to a larger capacity.

Since it inception in 1974 many developments have taken place. Perhaps the most significant is the creation of a separate company called Microsoft Inc, set up to market and further develop the Altair BASIC. Microsoft Inc BASIC interpreter software is becoming increasingly popular on microcomputers, particularly those based on the Intel 8080 microprocessor.

An alternative is however available from a company called North Star which is preferred by some users because it supports hardware floating point, and because of some differences in features offered by North Star BASIC.

Perhaps more significant from the hobbyist's point of view is the creation of the S100 bus which is something of an industry standard nowadays and used extensively by microcomputer manufacturers, even those who make use of microprocessors other than the Intel 8080.

Brian Crank left the RAF some 11 years ago to begin a career with "Wireless World" as an electronics engineer. He later joined the editorial staff and wrote various technical articles describing equipment of his own design. Among his articles published was the design for a very limited digital computer for educational purposes made up of discrete components (about 400 transistors!). He then became deputy editor of "Wireless World".

"I feel that one important design project in which I involved myself in those days is what I then called the 'Logic Display Aid', which I believe to be the forerunner of today's logic analysers offered by companies such as Hewlett-Packard", says Brian. "Hewlett-Packard's equipment has more bells and whistles, of course, but my design featured all the basic principles. It would display the Karnaugh Map, Truth Table or Venn diagram, on a convential oscilloscope, of any combinational logic circuit. It was described in a series of articles in 'Wireless World' during 1969."

So Brian was not green as far as electronics was concerned. However, he had no previous experience of programming at all when he first bought his computer. Learning the commands was easy enough, but the production of really efficient programs was another matter, and the first programs which he wrote were more exploratory than functional in nature.

Today, eighteen months later, the picture is quite different. The configuration is quite impressive and Brian has managed to apply sound management principles to what was originally no more than a few boxes of components and diagrams.

"The machine's operation is quite simple, really. I have modified North Star DOS (the disc operating system) so that BASIC is loaded automatically and comes up running as soon as 'RUN' is selected. A file directory is printed automatically.

"I use the teletype's keyboard in simplex mode, thereby isolating its printer. I then have the choice of three output peripherals: the teletype printer which produces print of matrix characters; the QUME Daisy-wheel printer which operates much faster with high-quality print, or, if hard-copy (paper) output is not required, the information may be displayed on the television screen. These peripherals are accessed by simple commands."

Brian Crank's software (using North Star BASIC) comprises four main applications suites: financial transactions; word processing; price lists; and analysis of reader reply cards.

One minifloppy diskette contains one year's financial transactions. "When I first developed the software for the financial applications, I had to update files with separate runs of the various programs comprising the suite. This was a tedious job at times so I decided to apply database management techniques in order to achieve file independence and integration. I got hold of a book called 'Principles of Data Base Management' by James Martin and this helped me no end. It is an excellent book and I thoroughly recommend it to anyone who wants to develop his own database management system."

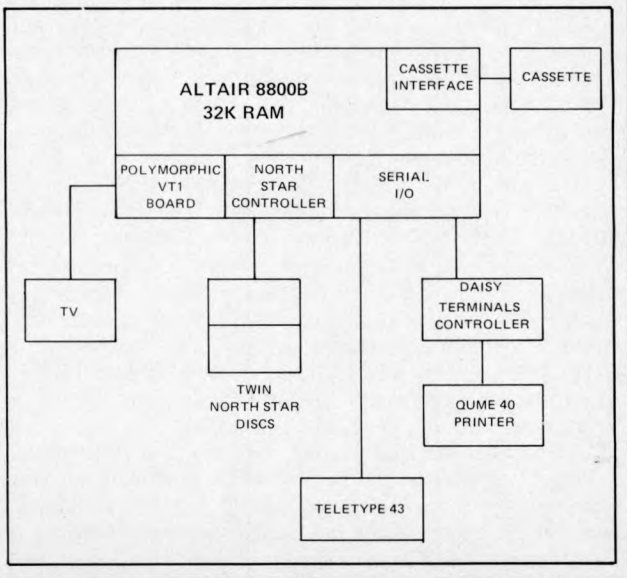
Once the database program has been loaded into memory from the diskette the following options are displayed on the screen:

1) Enter details of new invoice
2) An invoice is paid
3) Enter new expenditure
4) Call editor
5) Change program

If option 1) is selected, details of a new invoice are entered interactively with requests for information appearing on the screen and data verified at each stage. More specifically, the first request is for an invoice number; this begins with an alphabetical character which identifies the client, then a number for the invoice itself, and at the end of the number, an alphabetical character again which gives the income category to which the payment will be posted.

The rest of the details are entered in the same way and if anything is entered incorrectly the program will automatically go into editing mode for amendments to be made to the record number, the invoice details, date of purchase or for a search through the records by month or customer.

When payment is made against an invoice, the cashbook is automatically updated together with the customer's account in the ledger; a feature of the database software.



When a new client is entered, this is done by entering a new invoice number. The system does not recognise the new number and interactively asks for details. From this, a new chain is created for future file searches. "Customers are now changed on-line. Previously I had to change the program itself when I wanted to enter a new customer, but I overcame this by introducing a program module to deal with this."

At the end of the accounting period, typically on a monthly basis, reports may be produced by calling up the following menu onto the screen:

1) Amend the database (calls the database management program)
2) Print cashbook – expenditure
3) Print cashbook – income
4) Print client statements
5) Call the anlysis program
6) Print list of suppliers

A report of the expenditure by suppliers of which there are about 75 each identified by number is selected by keying in 2). This will display in chronological order each item of expenditure as it is incurred during that month, according to the following headings; supplier, VAT, Total and analysis. Analysis refers to type of expenditure such as freight and delivery, postage, travel expenses, stationery, telephone, photography, etc.

Income is listed by client with details, of invoice number, VAT, and invoice totals. Clients statements contain similar details plus date of invoice and age (days outstanding).

The analysis program collects figures from the database on a monthly basis. Categories which can be analysed include billings, income, expenditure, monthly cash flow, accumulated monthly cash flow, profitability, billing analysis and expenditure analysis. From this, one may elect to have the financial figures displayed numerically and graphically, or graphically only, by means of bar charts.

The word processing suite is used for occasions when invitations are sent out for events, exhibitions, press conferences, etc. Variables such as date, name of venue and address are entered, together with the time of the start of the event, the date the letter is to be sent, the name of the client and, finally, details of what the client will announce.

"Names and addresses of the people to whom invitations are to be sent are held on a disc file. The system automatically generates letters of invitation on standard letter headed paper, evelope labels for posting, and a list of invitees from which I may then follow up with an invitation. I also use the system for the printing of address labels for press releases and for captions to accompany photographs. For this latter activity, costs are automatically calculated and stored on disc. At the end of each month the file is passed to another program which automatically generates the invoices."

The computer and the Daisywheel printer form an excellent combination for handling price lists. In Brian's system, software has been written to handle product price lists. Such details as device type, description and price are sorted on disc in a base currency. The details of each form one disc record and there can be any number of records since the software will accommodate lists which occupy several discs. The editing section of the program allows new lines to be inserted, others to be deleted, and line to be altered at will.

Price lists can be printed out on the Daisywheel printer in any currency as long as the conversion factor between the base currency on disc and the desired currency is known. Price list pagination is automatic and page numbers and headings are printed. Page length can be changed at will to suit any stationery format. If carbon ribbon is used in the Qume, the output is so good that it can be used as camera-ready copy for bulk off-set litho printing or photo-copying.

The main advantage claimed by Brian for this system is the elimination of tedious proof-reading. Once the price list data has been entered and checked, it is only necessary to proof-read any changes that are subsequently made. Brian believes it is in this sort of application that the small computer is a real asset.

The suite for reader replies divides products into groupings. The number of cards received is input to the system, and for each product category the number of editorial mentions is counted. The average number of enquiries per mention is then calculated and compared with the actual enquiries per magazine, and on this basis the system allocates points and produces a scoreboard.

"Then, of course, I also have a number of games, including most of the usuals: 'Lunar Lander', 'Hunt the Wumpus', 'Market', 'In Out' (a word game for children), 'Stars', 'Weight Watching', 'Bio-rythm', 'Star Trek', 'Mastermind', and the 'Game of Life', all of which my daughters play from time to time. The 'Star Trek' program is quite interesting in that it is written in Palo Alto Tiny BASIC – an excellent small interpreter.

"A Heuristics Speech Lab has recently been added to the system, but the possibilities offered by this have not yet been investigated.

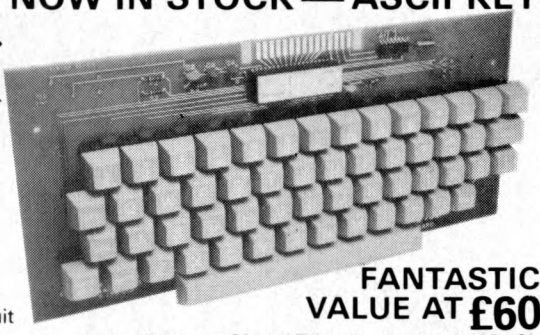
"I feel that the value of the microcomputer to the smaller business is incalculable and I would offer the following advice to the directors of small companies who cannot, as yet, write programs. The best way to learn is by experience. Take the plunge and buy a system. I did and have not regretted it."



NEW

56-STATION ASCII KEYBOARD

NOW IN STOCK — ASCII KEYBOARD MODEL KB756



Full 128 character set with ROM encoder (Upper and lower case + control shift).
Fully TTL — compatible — power requirements + 5v-12v.
Supplied with full technical data, code chart, pin connections, circuit diagram and application notes.

FANTASTIC VALUE AT £60

+ £1.50 p&p + 8% VAT (mail order total £66.42)

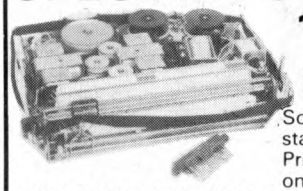
SELF-SCAN ALPHANUMERIC PANEL DISPLAY

16/18 position display with 64 character repertoire, 5 × 7 dot matrix. Input 6-bit BCD-code, power requirements + 5v, -12v. Character size 0.40" × 0.28". Overall dimensions 8¼" × 2¼" × 1¼". Supplied with full technical data. Price **£55.00** + 75p P&P + 8% VAT (Mail order total £60.21).

BALL MIRATEL VIDEO MONITOR

9" diagonal P4 phosphor tube. Bandwidth 12 MHz (-3dB). Input voltage 220V 50/60 Hz 24W. Output voltage + 15V DC (short circuit protected) + 12kV DC; 12.6V rms. Supplied complete with high & low voltage power supplies, amplifier, and attractive moulded plastic housing including space for keyboard. Price **£95.00** + carriage + VAT.

SPECIAL PURCHASE - MITE 123P Alphanumeric printer mechanisms



BRAND NEW SURPLUS ONLY £75
+ £3.50 P &P + 8% VAT
(Mail order total £84.78)

Solenoid-operated page printer using standard reversible typewriter ribbon. Prints standard 64-ASCII character set on 8½" paper (80 characters per line, 6 lines to the inch). Maximum speed 11 cps. Power requirements 115VDC. Compact, light-weight unit 9¾lbs, 12" × 9" × 2¾". Supplied complete with full technical manual.

We also specialise in: DEC minis — PDP8 and PDP11 processors, add-on memory, peripherals and spares. Hard copy terminals — ASR 33 and KSR 33 Teletypes, Data Dynamics 390, Texas Silent 700. Send for complete lists.

ELECTRONIC BROKERS LTD.
49-53 Pancras Road, London NW1 2QB.
Tel: 01-837 7781. Telex: 298694.

HAZELTINE VISUAL DISPLAY UNIT

* Teletype Compatible
* 12" Diagonal Screen
* TTY Format Keyboard
* 12 lines of 80 characters
* 64 ASCII Character Set
* 5 × 7 Dot Matrix
* Switch-selectable Transmission Speeds up to 9600 baud
* Switch-selectable Parity
* Standard CCITT V.24 Interface



MODEL H-1000 PRICE £350 + carriage and VAT

Also available: —

Model H-1200: Specification as for H-1000 except 24 lines of 80 characters displayed. Price **£425.00** + carriage + VAT.

Model H-2000: Buffered/Editing model with direct cursor addressing, dual intensity video, and detachable keyboard with separate numeric and edit clusters. 27 lines of 74 characters. Price **£495.00** + carriage + VAT.

A copy of trading conditions supplied on request

B•BUG SUPER SMART NEW MONITOR FOR NASCOM OWNERS

Features include:— FULLY COMPATIBLE with existing monitor/hardware: TAPE I/O 4 TIMES FASTER with extensive error checking (see cassettes below); INTELLIGENT COPY command for programme relocation; ARITHMETIC function for calculation of hex addresses and offsets; HEX KEYBOARD facility — throw away your ASCII tables; SUPER SHIFT allows all displayable characters to be entered directly from the standard keyboard; FLAG DISPLAY prints out all the flags that are set C, Z, etc; EXTENDED REGISTER DISPLAY shows the I,IX,IY etc registers as well; USEFUL SUBROUTINES include:— ASCII to packed BCD and vice-versa — CURSOR MOVEMENT — TABLE SEARCH — VARIABLE INTERRUPTABLE DELAY — RANDOM NUMBER GENERATOR — CHARACTER STRING OUTPUT — AUTO RUN facility allows a tape to be loaded (at high speed) and then execute automatically.

"B.BUG" is supplied in 2 x 2708 EPROMS which plug into your existing sockets. NO HARDWARE MODIFICATIONS are necessary. "B.BUG" is currently ex-stock but it is anticipated that nearly every Nascom owner in their right mind will want one, orders will be handled in STRICT ROTATION.

B•BUG in 2 x 2708's is only £24.50 including documentation.

C10 DATA CASSETTES only 38 pence each (Nascom, PET, TRS-80 etc.)

Please add 30p p + p for orders under £10. Prices include VAT.

VIEWFAX LTD.

KING EDWARD BUILDING, CORPORATION STREET, BIRMINGHAM B4 6SE.

PCW OPEN PAGE

The Amateur Computer Club View



Mike Lord

Among the more established ACC local groups, the *North West Group* now has well over 100 members and is still expanding rapidly. Having recently held their first AGM and adopted a formal constitution, they are now considering holding meetings every fortnight, instead of monthly, and are thinking of moving from their present meeting place at the National Computing Centre. For further details contact Ken Horton, 50 Lymefield Drive, Worsley, Manchester M28 4WA.

Bob Cottis, prime mover behind the *Thames Valley Group,* is moving to the land of the North West Group, so enthusiasts in the *Reading/Maidenhead* area are now invited to get in touch with Dave Howland, 4 Kent Lodge, Courtlands, Shoppenhanger Rd., Maidenhead tel: Maidenhead 36976 for the latest details of this group's activities.

It seems that until now *Bradford* Bit Bashers haven't been getting together. To remedy this Barry Waite of 315 Toller Lane, Bradford 9, tel: 498 750, has offered to act as catalyst for a new group in that area.

Phil Wheeler G8LSC would like to meet other computer hackers in the *Bromley/Orpington* area. No definite plans at the moment, but he hopes to get a few different micro systems together in one place to talk about and play with them. His address is 1 Irene Road, Orpington, Kent, tel: Orpington 23800.

The *Coventry* Group continues to flourish, with TRS80, 77-68 and MK14 machines being featured at a recent meeting. They are hoping to be able to hold their meetings at the Coventry (Lanchester) Polytechnic -- anyone wanting to go along should ring John or Roy Diamond on Coventry 454061 to confirm arrangements.

The *Southampton* ACC is a new group, but has strong links with the well established Southampton University ACC. Potential members are invited to contact P. D. Maddison, 13 Westridge Road, Portswood, tel: (0703) 558557.

The *Scottish* Amateur Computer Club meets on the first Wednesday of each month in the Glencairn Hotel, Royal Circus, Edinburgh. They also publish an excellent newsletter and are planning to establish a 'colonial' branch in Glasgow. For more details contact Stewart Stevenson, Lindisfarne, New Well Wynd, Linlithgow, tel: Linlithgow 2657.

The *Harrow* Group is now well under way, having set up a schedule of regular meetings and workshops, under the guidance of Alan Secker, 209 Albury Drive, Pinner, Middx HA5 3RH, tel: 01-428 0844.

It looks like there is a need for another group in the *Sittingbourne* area of Kent; J. M. Baron of 27 Wises Lane, Borden Sittingbourne, tel: Sittingbourne 70160 will be acting as contact man for gregarious computer hobbyists in that area.

Finally, the ACC itself is organising a series of meetings in the London area, based on presentations from the suppliers of systems such as TRS-80, Apple, PET, 9900. By the time you read this the programme will have been settled and details will have been published in the ACC Newsletter. If you're not a member of the ACC (and why not ?) send an SAE, or ring, Mike Lord at 7 Dordells, Basildon, Essex, tel: (0268) 411125.

PCW Readers: This is the Open Page, so invitations such as the one following are open. Take good advantage of them!

NORTH LONDON HOBBY COMPUTER CLUB

On Wednesday October 5th 1978, at 6.30 p.m. we are holding the inaugural meeting of the above club. This will be held in **Room 4** in the **Old Building** at **Holloway Road**, just opposite Holloway Road underground station on the Piccadilly Line. I would be pleased if you could mention this fact somewhere in your magazine so that as many interested people as there are might come.

The Department and the Polytechnic have made available many resources for this venture. Within the Department there are two PETS (with a third coming), four SWTPC 6800 computer systems, with floppy discs, printers and VDUs and some KIM and Motorola microcomputer systems. Most of these will be available for use, as will some PETs and SWTPC systems in other departments.

As we envisage the club at the moment, little "homebrew" activities are anticipated before Christmas, with any meetings centering around talks by manufacturers and discussions on programming, etc. However, from the new year we anticipate three sets of activities running concurrently, or sequentially (it all depends on how many people turn up!). These are short courses on programming, Basic and machine level; a homebrew section using the facilities of the department — up to 35 people can solder and test at the same time — and introductory talks and discussions for those anticipating their own systems.

As you can see, we are preparing a varied programme that should be of interest to a wide variety of people. Obviously, students from the Poly will be coming along, but we want to emphasise that this is a club open to *all* interested. The Poly will be providing some backup, especially with expert staff and other facilities. This is all part of the **Community Development Programme** that has recently been instituted.

I hope that you will be able to help us get this off the ground. Those organising it are members of the Amateur Computer Club, as well as lecturers in digital electronics.

Robin Bradbeer (Acting Club Secretary) Senior Lecturer,
Dept. of Electronics & Communication
Polytechnic of North London

MUSE NEWS

The Summer Course was held at Westhill College, Birmingham from July 14th to 16th and was attended by 75 people, which was the most that the College could manage at this time. Courses ranged from beginners BASIC to machine code for the M6800 and the 380Z, and packages on disk systems, and a good time was had by all.

There were demonstrations by current users of the PET, the 380Z, the TRS-80, the SWTPC, and the Apple.

There were three main issues that emerged during the course. The first was that there was no clear leader in all respects amongst the current micro-computers, and that most of them had something to recommend them. The second was that Computer Assisted Learning (CAL) was becoming increasingly important to practising teachers and that graphics was a part of this. The third was the increasing concern that money was about to be spent on education, and that it would be wasted due to the fact that practising teachers were by and large not involved in the decisions about how to spend it. These issues were discussed in the working groups that took up six sessions and in the one general session.

At the AGM the Committee was re-elected with the exception of Graham Batty who wished to retire. Bob Trigger was elected Treasurer in his place.

The subscription has been raised to £5 per annum, due to the high cost of the MUSE letter.

The most recent MUSE letter has been published and consists of 96 pages of software, reports, articles, and hardware. It is available from the Secretary at the cost price of £1.

The next meeting will be at Trent College on 3rd November on the subject of A-level projects. Write to Michael Ling, Trent College, Long Eaton, Nottingham for details.

For details of how to join MUSE write to the Secretary — Charles Sweeten, 18 South Road, Oundle, Peterborough. Membership is aimed at institutions with an interest in Secondary education, but many others have joined and are welcome. Membership is not limited to those who have their own computer.

BUSINESS systems for as little as £3037. Tailor made software commissioned. SWTP products for personal and educational use also. Books and magazines available. Barclaycard accepted. **VERWOOD SYSTEMS** — Telephone Rugby 87629 for catalogue.

PROGRAMMING IN BASIC BY QUALIFIED LECTURER. A COURSE OF 4 EVENINGS FOR £25. FOR DETAILS WRITE INCLUDING AN SAE TO S. WILLIAMS. 114, BEECH ROAD, ST. ALBANS, HERTS.

Star Trek

Now available on cassette a version of startrek utilising the graphic capabilities of the Commodore PET. Destroy the klingon invasion force before they destroy Enterprise. Complete operating instructions £4.50).

Send cheque, postal orders payable to M. Green, 57 Ross Road, South Norwood, London SE25 6SB.

PAYROLL AND VAT FOR YOUR COMMODORE PET.

Just two of the programs available for your PET: PAYROLL calculates P.A.Y.E. and displays or prints payslips. VAT calculates your VAT returns and formats results as the form. PAYROLL — £20. VAT — £17.50. SAE for list: COMPUTAWARE, 479 WELLINGBOROUGH RD., NORTHAMPTON. 714821.

The low cost of microcomputer is now putting computer power within the reach of the small business

MCS specialise in the programming installation and sale of microcomputer systems.

We are North London Agents for the renowned Apple II computer

Let MCS show you how a microcomputer can save you time and money in your business

For a complete system or just a demonstration contact us right now, or meet us on stand 18 at the PCW show

MICROSOLVE

Microsolve Computer Services Limited
252 Hale Lane, Edgware, Middlesex HA8 8NT
Telephone: 01-958 4347

IDES MICROCOMPUTER SYSTEM

The IDES processor is a realistically priced development system for National's SC/MP when used alone. It may be expanded to become a powerful process controller or your own 'personal computer'.

IDES PROCESSOR:-

- 7 segment displays for every register (32 digits)
- Run program in single or slow step
- HARDWARE breakpoint registers allows bugtraps on program, data or ROM!
- MOVE or COPY blocks in memory
- PROGRAM from RAM to 2708 or 2758 EPROM*
- LOAD and STORE to tape CASSETTE*

KIT £99.50 + VAT

** TV interface, 64 chars. by 16 lines £120 KIT
* Cassette storage interface £75 KIT
* EPROM programmer interface £45 KIT
* Process control oriented BASIC £95 KIT
* Also RAM, ROM boards etc.

For details send large SAE to:-

MUTEK

QUARRY HILL, BOX, WILTS. TEL (022 121) 3289

Gordon Bell
Technical Director, MSL

PET PREENING

A NEW ART FORM?

Readers fortunate enough to own or have acces to a Commodore PET machine may like to try the very small program shown below.

```
10 POKE 32768 + INT (999 * RND (2)), INT (255 *
RND (3) 20 GOTO 10
```

The program may not be very sophisticated or of immediate practical value but it does produce a rather interesting pattern on the VDU screen and shows some of the original capabilities of the PET machine.

For those who do not appreciate the working of the program a short description follows.

The POKE statement takes the form POKE X, Y where the decimal value of Y is stored as an ASCII character at the decimal address X. In the program, therefore, a random character is generated with a value between 0 to 255. This character is then stored in amemory location calculated as 32768+ a random number in the range 0 to 999. This range of a thousand memory locations is that used in the PET for the VDU display so characters deposited in this area will appear on the screen. A simple unconditional loop ensures continuous operation of the program. For maximum effect it is best to clear the screen before typing the RUN command.

Having now outlined the principle of operation, it will be interesting to see what variations on this approach can be devised by Personal Computer World readers.

"B[illegible] the buffer speed my sales ledger was up in 20 minutes."

What is important is whether the system works for you and how long it takes to get you on the air.

We are distributors of the full range of apple products and other peripherals with software back-up.

Collins Consultants

For information or demonstration ring Collins Consultants, Tel: (02816) 2572 or Templeman Software, Tel: (0789) 66237

224 ARTICLES
Kilobaud has more articles than any other microcomputer magazine. During 1977, for instance, there were 224 articles in Kilobaud—880 pages of articles—that's like a very large encyclopedia of microcomputing. There will be even more articles in 1978.

FOR THE BEGINNER
Though Kilobaud covers both the technical and programming sides of microcomputers, each article is written with the beginner in mind. No other magazine makes it as easy to understand microcomputers.

THE BEST PROGRAMS
Only Kilobaud offers programmers both the publication of their programs in the magazine plus a large royalty for the program if it's issued on cassettes. Is it any wonder all of the really good programs are being published in Kilobaud? You'll find the best in games, diagnostics, teaching, music, business, etc., programs in Kilobaud. We're very heavy on programs.

HOBBY OR BUSINESS?
Both businessmen and hobbyist want the same thing: to understand microcomputers . . . and this is the purpose of Kilobaud. KB also has articles aimed at the businessman to help him know what is available and what it will do.

THE PUBLISHER
Kilobaud is published by the same people who put out 73 Magazine—the largest of the amateur radio magazines.

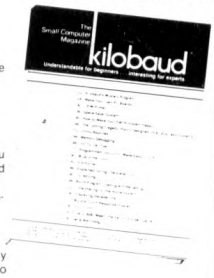
MONTHLY COLUMNS
Keep up with the latest developments and readers' programs through the letters column . . . a KIM column . . . one on the TRS-80 . . . a column on BASIC . . . etc. The editorials will keep you up to date on money-making ideas, the progress of the industry and more. The New Products column is particularly useful, including the results of tests of the latest equipment in the Kilobaud microcomputer laboratory—the most complete in the industry.

SUBSCRIBE TO KILOBAUD
You can get Kilobaud fast. Copies are flown to Europe immediately upon publication and mailed directly from the United Kingdom, so your copies are current. Send subscription orders for United Kingdom and Europe, £20 sterling per year, to:

L.P. Enterprises/313 Kingston Road/Ilford/Essex IG1 1PJ/England.
Barclay Card, VISA, Diner's Club and American Express, Access Card honored.

If you only read ONE American computer magazine...

it should be



Getting It Together

Mike Banahan

The previous articles have taken a quite broad look at assembly language: this article becomes specific about one particular assembler later on but, before that, let's 'stand back' and have a think.

Why use assembly language at all? That isn't as daft a question as it might sound. There's little doubt, I agree, that for small programs written by an experienced machine code programmer, it can be efficient. Efficient, that is, if you need the resulting program to be as small as possible, or as fast (at run-time) as possible. How often though, are speed or size really the limiting constraints on a program? In the commercial world programmer time is often the most expensive part of producing and running programs – that's why high level languages like COBOL are so popular. They produce results – so what if the code they produce is massive or slow. In home computing a similar situation can be found. The people with BASIC implemented are the ones who are producing working programs, because writing in assembly language is such a slow process.

In some circles it's even believed that good compilers can produce MORE efficient code than assembly language programmers. Admittedly, it only applies to very large programs where mere humans tend to get into the 'can't see the wood for the trees' syndrome. However there are some very clever compilers about and it may not be long before, at least commercially speaking, the assembly language programmer becomes about as useful as an alchemist in a oil refinery. The place of assembly language is gradually being taken over by compiled languages such as BCPL, which give the flexibility of assembly language coupled with the advantages of built-in input output routines and the power of a high-level language. Incidentally, if anyone has implemented BCPL on the Z-80, I'd be very interested to get hold of it.

For the time to come though, assembly language will play an important part in the field of home computing. Compilers are big, clever ones very big, and few home computer fanatics will be able to afford the memory to run them in. Does your system support 250k of COBOL compiler? I thought not. For you, and the likes of me who just enjoy playing with machine code, here it comes.

INTRODUCTION TO A Z-80 ASSEMBLER.

A little history may help to explain, if not excuse, the state of this program. It's in what enthusiastic salesmen tend to term 'an experimental state, Sir'. Which is to say, though not necessarily right, it isn't proved wrong either. It has actually managed to assemble itself without any glaring errors, but the whole exercise of writing it was geared more to producing something that worked a bit than something definitely right. This was because the assembler was developed on a system which was shortly to become unavailable, and it was more important that it could assemble itself than assemble everything. My reasoning was that if it could assemble itself, then it could assemble a corrected version of itself when bugs were found. In the event, shortage of time meant that it couldn't even be proved to assemble itself correctly (cries of 'shame'), but it does try bravely. Checked against the alphabetic list of opcodes given in the Zilog programming manual (a nasty job typing them up, too), all looks O.K. So it will probably only be the error detection that has any major faults.

The assembler deals with many of the opcodes separately which means that mods to one section of the program should not have a dire effect on other parts; except of course when the mods are made to the shared routines.

Of these shared routines undoubtedly the most unpleasant is the mathematical evaluation section which needs to be re-written. It works enough for the self-assembly process but stops at that point.

The assembler was written as an attempt to copy, on the Z-80, a cross assembler already existing and running on a PDP-11. The cross– assembler was written by John McFerran of Bradford University to whom most of the credit for the syntax should go. I would like to express my appreciation for his co-operation and help, and also

that of Doctor Peter Comerford, without which this task could not have been attempted. It wasn't possible to incorporate all of John's ideas into this assembler; a pity, because anything which shakes the writers of assemblers out of their apathy would be welcomed.

SYNTAX.

The following notes on the syntax should be read whilst examining the relevant parts of the assembler. They are meant to explain, not define the syntax.

1. STATEMENTS

Statements consist of the OPCODE followed by operands where valid. A statement is terminated by semicolon; or carriage return. The load opcode is the only exception, it consists of destination, leftarrow,source. Some opcodes take an indefinite number of operands, e.g. PUSH AF,BC,DE,HL .....

2. ASSEMBLER DIRECTIVES

There are a number of directives, mainly used for the reservation of storage, either as 8-bit quantities (words), or 16-bit ones (addresses).

.WORD and .ADDR take arguments which are evaluated and assembled as consecutive words in store allowing tables of constants to be produced.

.BLKW and .BLKA take one argument and reserve that many locations — if no argument is specified, a value of 1 is assumed. The locations so reserved have no specific value written into them.

.ASCII and .ASCIZ operate like .WORD, but allow strings of characters as parameters. The strings are bounded by quote marks ". To include quote marks in the string, double quotes must be used " ". .ASCIZ inserts a null word at the end of the block, .ASCII does not. .END tells the assembler that it has read all of the input text. It takes one argument, the program start address, which is used by the loader. More about that later.

3. IDENTIFIERS

Identifiers may consist of up to 6 characters in length, starting with an alphabetic one. Labels terminate with colon: variables with an equals sign =. The value associated with a variable may be changed at any time, the value associated with a label is the address at which it was declared, and cannot be changed.

4. EXPRESSIONS

The arithmetic capability of this assembler is not what it might be. It should have been able to deal with addition subtraction, multiplication and division. Well at least it adds and subtracts! A special expression is used when the value of a character is wanted. SEMICO='; will assign the ASCII value of the character after the ' mark to the variable SEMICO. The assembler has now been modified from the original octal-only to accept any base, due to popular demand. The astute amongst you (i.e. other than the very dim) will notice the use of the SETRAD pseudo-op at the head of the assembler listing, and the fact that the numbers in the source code are still octal. SETRAD was included for the sake of those who become bored by working in the same number system for any length of time; it allows them to change the current radix to anything between 2 and 16 (decimal). Beware of expecting SETRAD 10 to have any effect at all. The 10 will be evaluated in whatever the current radix is and do absolutely nothing. To make sure that a number is taken in the desired radix use the other form of radix control which is (radix)uparrow(number), e.g. 10↑22 will give the value 22 base 10 (decimal). The number preceding the uparrow is ALWAYS evaluated as a decimal quantity no matter what you've been up to with SETRADS.

5. INDEXING

Unlike the standard syntax of (IX+OFFSET), this assembler uses OFFSET (IX).

6. CONTRACTIONS

To clear or test the accumulator, the opcodes CLA and TSTA are included. They correspond to XOR A and OR A.

7. NUMBERS AND ADDRESSES

Where confusion might arise, numbers must be preceded by a # mark, otherwise the assembler assumes an address is meant. A←#7; loads A with the number 7, A←7; loads A from address 7.

8. COMMENT

Comment is now preceded by backslash \. All text following the backslash, up to and including the next backslash or end of line is ignored. Space or tab followed by comment is treated as space.

As it stands this is a three pass assembler, because the development system only had a teletype for output: the binary and listing passes are separate. The teletype reader must be controllable by the processor. Blocks of 64 characters at a time are read — this is deliberately to allow for easy conversion to block — structured input devices such as cassette: it means the code already exists to handle the blocks.

The binary output is in the following format:

```
1
0
0
number of bytes in block
high order start address
low order start address
bytes
of code
```

ONE BLOCK
THE CHECKSUM IS *NOT* PART OF THE BLOCK

checksum— negated 8-bit addition of all bytes in block.

THE ASSEMBLER

Here it is in all its glory. As an added bonus it comes in the form of a listing produced by yet another experimental assembler, which gives the first 9 bytes of code generated by a line. It isn't possible to provide the assembler's listing of itself but a sample of its' output is given to show that it really does work.

If anyone is keen enough to want an object copy of the assembler it could probably be arranged. Indeed if anyone will lend me an 8k Z-80 system I'll happily set the whole beast animated and working.

(PCW Pages 1 to 6 are published this issue; other sections follows in future issues **PCW).**

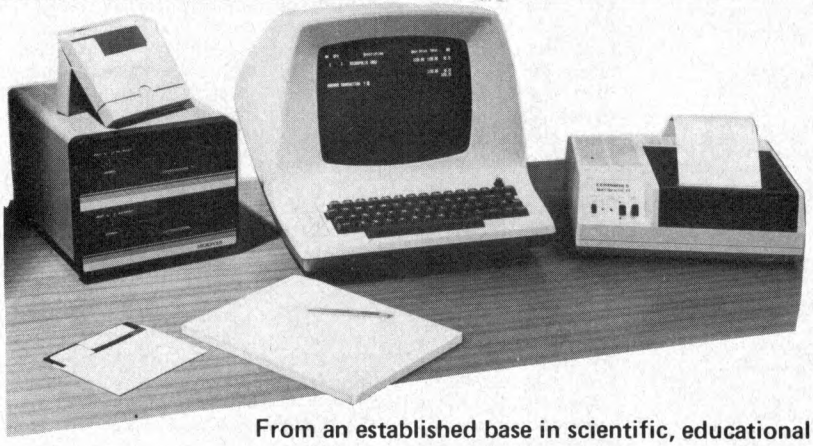
```
>

Z-80 ASSEMBLER V01
PASS 01
*
PASS 02
*
0000                            TITLE MIT-80 LOADER
0000                    .=0000
0000 3E 04 D3 FF        START:  A=#4;OUT TTYCN1,A
0004 CD 63 00 F7 00     BEGIN:  CALL CHIN;STATUS? Z,.-3         ;WAIT WHILE
0008 3D 00 3F                   INC A;IF NZ,BEGIN+3             ;RECEIVE FOR TITLE
000B 06 00                      B=#0
000D CD 63 00 F7 00             CALL CHIN;STATUS? NZ,BEGIN+3;IF NZ .-3      B1
0012 4F                         C=A\    A IS NOW ZERO
0013 CD 6C 00                   CALL CHECK                      '
0016 47                         B=A\    LO BYTE
0017 CD 6C 00 67                CALL CHECK;H=A
001B CD 6C 00 6F                CALL CHECK;L=A
001F 78 E6 00 00 11             A=B;SUB #0;IF Z,TXFR
0024 47                         B=A
0025 CD 6C 00 77 D3     LOOP:   CALL CHECK;(HL)=A;INC HL
002B 10 F9                      IFNZ LOOP\      C HOLDS CHECKSUM
002D CD 63 00 81 3C             CALL CHIN;ADD C;INC A;IF NZ,BEGIN
0032 18 CA                      JR BEGIN
0034 AF D3 FF           TXFR:   CLA;OUT TTYCN1,A\       STOP LOADER
0037 00 FF FF                   0FFFF=HL\       TO BE IN RAM
003A 21 74 00                   HL=#DONE
003D 18 06                      JR OUT
003F C3 11 F1           STOP:   JP 0F111\       BACK TO MONITOR
0042 21 5F 00           BEGIN:  HL=#NASTY
0045 0E 11              OUT:    C=#TTYSTA
0047 D1 70 E6 41 00     OUTLP:  IN A,(C);J A,.
004C 7E B7 00 FF                A=(HL);STATUS? Z,STOP
0050 23                         INC HL
0051 D3 1C 18 F2                OUT TTYTI,A;JR OUTLP
0055 45 52 52 4F 52     NASTY:  .ASCIZ"ERROR",15,1F
005D DF D1 E6 40 00     CHIN:   IN A,TTYSTA;AND #40;IF Z,.-
0063 DF DC                      IN A,TTYTI
0065 C9                         RET
0066 CD 63 00           CHECK:  CALL CHIN
0069 57 81 4F                   E=A;ADD C;C=A
006C 7A C9                      A=E;RET
006E 4C 4F 41 44 45     DONE:   .ASCIZ"LOADER",15,1F
0077                    TTYSTA=0011;TTYTI=0C;TTYCN1=01F
0077                            .END START

SYMBOL TABLE

.       =0077     BEGIN  :0042     CHECK  :0066     CHIN   :0063
DONE    :006E     ERROR  :0048     NASTY  :0055     OUT    :0045
OUTLP   :0047     RET    :0000     START  :0000     STOP   :003F
TTYCN1  =001F     TTYSTA =0011     TTYTI  =000C     TXFR   :0034

PASS 03
*
```

```
0000                    ;       THIS LISTING HAS BEEN EDITED TO SHOW
0000                    ;       CHARACTERS WHICH WOULD OTHERWISE HAVE  OT
0000                    ;       PRINTED ON THE PARTICULAR PRINTER USED.
0000                    ;       LEFTARROW IS NOW SHOWN AS <-
0000                    ;       DOUBLE QUOTES AS %
0000                    ;       UPARROW AS &
0000                    ;       AND BACKSLASH AS !
0000                    ;       TITLE Z-80 RESIDENT ASSEMBLER
0000                            SETRAD 8
0100                            .=400#! SET ORIGIN
0100
0100 310001CDCC0E       BEGIN:  SP<-#STACK#CALL CRLF#!  SET UP STACK
0106 CDA80E                     CALL PRINT#!    AND PRINT A MESSAGE
0109 5A2D38302041535345         .ASCIZ %Z-80 ASSEMBLER V01#,CR,LF
011E AF3C32B701                 CLA#INC A#PASFLG<-A
0123 010800216015118415         BC<-#10#HL<-#DOTMES#DE<-#SYMTAB
012C EDB0                       LDIR#!  RENEW ENTRY FOR 'DOT'
012E                    ;               IN SYMBOL TABLE
012E CDB801                     CALL PASS
0131 C7                         RST 0#  IBET IT DON'T GET HERE
0132                    ;       IF IT DOES, RE-ENTER MONITOR
0132
0132                    GRADIX: .BLKW
0133 0100               LBLOW:  .ADDR 1#        !LOW ADDRESS IN LINE BUFFER
0135                    ATLENG: .BLKW 1#        !LENGTH OF CURRENT ATOM
0136
0136
0136                    GETATM:
0136 E5D5F5C5                   PUSH HL#PUSH DE#PUSH AF#PUSH BC
013A 21AC150607                 HL<-#ATBLCK#BC<-#7#     !CLEAR SOME SPACE
013F 36002310FB                 (HL)<-#0#INC HL#DBNZ .# !HANDY LATER
0144 3A5E0DB72009               A<-EMPTYF#TSTA#BR NZ,GET.0#!    BUFFER EMPTY?
014A CD5F0D                     CALL TAPEIN#!           IF SO GET ANOTHER
014D 2148DE223301               HL<-#INBUFF#LBLOW<-HL
0153 2A330111AC15       GET.0:  HL<-LBLOW#DE<-#ATBLCK#! SET POINTERS
0159 7EFE202003         GETSP0: A<-(HL)#CMP#SPACE#BR NZ,GETSP1
015E 2318F8                     INC HL#BR GETSP0#!      NO LEADING SPACES
0161 FE092003           GETSP1: CMP#TAB#BR NZ,GETCOM
0165 2318F1                     INC HL#BR GETSP0#!      OR TABS
0168 FE5C2008           GETCOM: CMP#'!#BR NZ,GET.1
016C 237EFE0D2805       GETC1:  INC HL#A<-(HL)#CMP#CR#BR Z,GET.1
0172 FE5C20F6                   CMP#'!#BR NZ,GETC1
0176 23                         INC HL
0177 7EFE0D2004         GET.1:  A<-(HL)#CMP#CR#BR NZ,GET.3
017C 08                         EX AF
017D AF325E0D                   CLA#EMPTYF<-A#! EMPTY!
0181 08                         EX AF
0182 CD9A0E             GET.3:  CALL WOTSIT#!   WHAT SORT OF CHARACTER?
0185 E601                       AND#TERMIN#!    LOOK FOR TERMINATOR
0187 2004                       BR NZ,GET.2
0189 EDA018EA                   LDI#BR GET.1#!  PUT INTO ATBLOCK &INC POINTERS
018D 23223301           GET.2:  INC HL#LBLOW<-HL#!      POINT TO NEXT
0191 0832B81108                 EX AF#TERMIT<-A#EX AF#! TERMINATOR IN BOTH TERMIT
0196                                                    !AND 'A
0196 21AC15EB87         GET.R:  HL<-#ATBLCK#EX DE,HL#TSTA#!     CLEAR CARRY
019B ED52                       SBC HL,DE#!     HOW LONG IS THE ATOM?
019D 70323501                   A<-L#ATLENG<-A#!        ATLENG CONT'S LENGTH OF ATOM

01A1 FE09                       CMP#11
01A3 3808                       BR C,.+15
01A5 21E212CBCECDEE13C3         HL<-#ATLONG#SET 1,(HL)#CALL ERRPRN#J BEGIN#!    SYMBOL TABLE CORRUPTED
01B0 C1F1D1E1                   POP BC#POP AF#POP DE#POP HL
01B4 C9                         RET
01B5
01B5                    NULFLG: .BLKW 1#!       CHECK FOR ANY NULLS EVER
01B6                    FIRSLN: .BLKW
01B7                    PASFLG: .BLKW
01B8
01B8 CDFD12             PASS:   CALL ERRST#!    CLEAR ERROR FLAGS
01BB 3E10323201                 A<-#20#CRADIX<-A#!      SET DEFAULT RADIX
01C0 3A8701FE04CC1214CA         A<-PASFLG#CMP#4#CALL Z,BLANK#J Z,BEGIN#!        TRAILER ON OBJECT TAPE
01C8 FE03CC0F13                 CMP#3#CALL Z,SYMPRN#!   PRINT SYMBOL TABLE

01D0 CDA80E504153532000         CALL PRINT#.ASCIZ%PASS %
01D9 CDCB12CDCC0E               CALL PR2#CALL CRLF
01DF CDA80E2A00                 CALL PRINT#.ASCIZ%*%
01E4 CDC10E                     CALL CHIN#!     TYPE ANYTHING TO CONTINUE
01E7 CDCC0E                     CALL CRLF
01EA 3E4032130E                 A<-#100#T.POS<-A#!      INPUT-BUFFER POINTER
01EF 3A8701FE03CC1214           A<-PASFLG#CMP#3#CALL Z,BLANK#!  LEADER FOR OBJECT TAPE
01F7 AF676F22BA15               CLA#HL<-A#L<-A#OURDOT<-HL#!     RESET DOT
01FD 32F91122FB11               INSCNT<-A#BOTADD<-HL#!  AND OTHER THINGS
0203 32B601                     FIRSLN<-A
0206 226815325E0D32B501         CURADD<-HL#EMPTYF<-A#NULFLG<-A
020F 3A8701D2D007               A<-PASFLG#DEC A#BR NZ,REPT
0215 218C1577226A15             HL<-#SYMSRT#(HL)<-A#SYMNEX<-HL
021C
021C                    ;       THE BIT BELOW FINDS OUT WHAT IS GOING
021C                    ;       ON IN THE INPUT TEXT AND CALLS THE PRELIMINARY
021C                    ;       ROUTINES TO DEAL WITH IT.
021C
021C CD360108FE28CCB909 REPT:   CALL GETATM#EX AF#CMP#'(#CALL Z,LOAD#BR Z,.
0227 3A3501B728EF               A<-ATLENG#TSTA#BR Z,REPT
022D 3AAC15FE2ECC6602           A<-ATBLCK#CMP#'.#CALL Z,DOTPRO
0235                                    !PUTTING IT HERE PREVENTS
0235                                       !.BLKW (FOR EXAMPLE)
0235                                    !BEING TAKEN AS A ONE WORD OPCODE
0235
0235                            BR Z,REPT
0237 3A8811                     A<-TERMIT
023A FE5FCCB9092ADB             CMP#'<-#CALL Z,LOAD#BR Z,REPT#!  IT'S A LOAD INSTRUCTION
0241 FE3C                       CMP#'<
0243 2008                       BR NZ,P1.NX
0245 CD81DFCD8909181CF          CALL EVALUA#CALL LOAD#BR REPT
024D CD9A0EE6102A05     P1.NX:  CALL WOTSIT#AND#LABL#BR Z,P,1MOR
0254 CD1F0F18C3                 CALL LABPUT#BR REPT
0259 CD880ECC4100B8BB   P.1MOR: CALL SEMCR#CALL Z,ONEPRO#BR Z,REPT
0261 CDBD031886                 CALL OTHERS#BR REPT
0266
0266                    ;       DEAL WITH ANYTHING BEGINNING WITH .
0266
0266 F5C5D5E521EF10CDD3 DOTPRO: PUSH AF,BC,DE,HL#HL<-#DOTTAB#CALL SEARCH
0270 C4ED12C2A103               CALL NZ,SYNTER#J NZ,DOTRET#!    SEARCH PSEUDO-OP TABLE
0276 E9                         J (HL)#!        IF FOUND, PROCESS IT
0277 3A8811FE3D         NEWDOT: A<-TERMIT#CMP#'=
027C C4ED12C2A103               CALL NZ,SYNTER#J NZ,DOTRET

0282 CD1F0F2ABA15226A15         CALL LABPUT#HL<-OURDOT#CURADD<-HL#J DOTRET#!     SORT OUT .=
028E 21870134           END:    HL<-#PASFLG#INC (HL)
0292 CD3601                     CALL GETATM
0295 218415CDD30F               HL<-#SYMTAB#CALL SEARCH#!       IS START LABEL PRESENT?
029B C4ED12                     CALL NZ,SYNTER
029E 3A8701FE032803             A<-PASFLG#CMP#3#BR NZ,END.3
02A5 CDE00D                     CALL LINPRN#!   PRINT LAST LINE
02A8 FE04200D           END.3:  CMP#4#BR NZ,END.1
02AC 3AF911B7C46E12             A<-INSCNT#TSTA#CALL NZ,OUT,PT#! OUTPUT LAST CODE
02B3 22FB11                     BOTADD<-HL#!    TRANSFER ADDRESS
02B6 CD6E12                     CALL OUT,PT
02B9 E1D1C1F1D3BA01     END.1:  POP HL,DE,BC,AF#J PASS
02C0                    RLKZ:
02C0 3A8811CD880E               A<-TERMIT#CALL SEMCR
02C6 200F                       BR NZ,RLZ.DO
02C8 210100                     HL<-#1#!        DEFAULT VALUE
02CB AFCDFD112B7CB52AF7 RLZ.OU: CLA#CALL OUTINS#DEC HL#A<-H#OR L#BR NZ,.
02D4 C3A103                     J DOTRET
02D7 CDA603             RLZ.NO: CALL GIFNL#!    MORE ARGUMENTS?
02DA B5                         OR L
02DB 20EE                       BR NZ,RLZ.OU
02DD C3A103                     J DOTRET
02E0                    WORD:
02E0 CDA603                     CALL GIFNL#!    A CONTAINS TOP 8 BITS
02E3 2805                       BR Z,WOR.OK#!   IF NOT ZERO TRUNCATION ERROR
02E5 CDF512AF6F                 CALL TRERR#CLA#L<-A
02EA 7ACDFD11           WOR.OK: A<-L#CALL OUTINS
02EE 3A8811FE2CC2AE8            A<-TERMIT#CMP#',#BR Z,WORD
```

```
29  02F5 C3A103                     J DOTRET
30  02F8                    ADDR:
31  02F8 CDA603                     CALL GIFNL#I    16 BITS IN HL ON RETURN
32  02FB 7DCDFD117CCDFD11           A<-L#CALL OUTINS#A<-H#CALL OUTINS
33  0303 3A8811FE2C28EE             A<-TERMIT#CMP#',#BR Z,ADDR
34  030A C3A103                     J DOTRET
35  030D                    BLKA:
36  030D 3A8811CD880E               A<-TERMIT#CALL SEMCR
37  0313 2012                       BR NZ,BLA.00
38  0315 210100                     HL<-#1
39  0318 29ED5B6815190          BLA.00: ADD HL,HL#DE<-CURADD#ADD HL,DE
40  031E 22BA15226815C3A103         OURDOT<-HL#CURADD<-HL#J DOTRET
41  0327 CDA60318EC             BLA.D0: CALL GIFNL#BR BLA.00
42  032C                        ASCIZ:
43  032C CD3C03AFCDFD11             CALL ASCI#CLA#CALL OUTINS
44  0333 C3A103                     J DOTRET
45  0336                        ASCII:
46  0336 CD3C03C3A103               CALL ASCI#J DOTRET
47  033C                        ASCI:   ;COMMON TO BOTH .ASCIZ & .ASCII
48  033C 3A8811FE20                 A<-TERMIT#CMP#SPACE
49  0341 CC3601                     CALL Z,GETATM
50  0344 3A8811FE222025             A<-TERMIT#CMP#'"#BR NZ,ASC.NQ
51  034B 2A3301                     HL<-LBLOW
52  034E 7EFE22280A             ASC.11: A<-(HL)#CMP#'"#BR Z,ASC.4
53  0353 FE0D280C                   CMP#CR#BR Z,ASC.3
54  0357 CDFD112318F1           ASC.41: CALL OUTINS#INC HL#BR ASC.1
55  035D 237EFE2228F4           ASC.21: INC HL#A<-(HL)#CMP#'"#BR Z,ASC.4
56  0363 328811                 ASC.31: TERMIT<-A
57  0366 FE0D                       CMP#CR

 1  0368 2801                       BR Z,.+3
 2  036A 23                         INC HL
 3  036B 223301180A                 LBLOW<-HL#BR ASTER
 4  0370 CD810FC4F512           ASC.NQ: CALL EVALUA#CALL NZ,TRERR
 5  0376 7DCDFD11                   A<-L#CALL OUTINS
 6  037A CD8703CC3601288A       ASTER: CALL COMCHK#CALL Z,GETATM#RR Z,ASCI
 7  0382 C9                         RET
 8  0383                        BLKW:
 9  0383 3A8811CD880E2011           A<-TERMIT#CALL SEMCR#BR NZ,BLW.DO
10  038B 210100                     HL<-#1
11  038E ED5B6815190            BLW.00: DE<-CURADD#ADD HL,DE
12  0393 22BA15226815C3A103         OURDOT<-HL#CURADD<-HL#J DOTRET
13  039C CDA60318ED             BLW.D0: CALL GIFNL#BR BLW.00
14  03A1 E1D1C1F1C9             DOTRET: POP HL,DE,BC,AF#RET
15  03A6 3A8811FE20             GIFNL:  A<-TERMIT#CMP#SPACE
16  03AB CC3601                     CALL Z,GETATM#I SKIP LEADING SPACE
17  03AE FE2CCC3601                 CMP#',#CALL Z,GETATM
18  03B3 CD810FC9                   CALL EVALUA#RET#I       GET VALUE OF ARGUMENT
19  03B7 3A8811FE2CC9           COMCHK: A<-TERMIT#CMP#',#RET
20  03BD
21  03BD                    ;       MOST OPCODES DEALT WITH HERE
22  03BD
23  03BD                    OTHERS:
24  03BD F5C5D5E5                   PUSH AF,BC,DE,HL
25  03C1 21B808CDD30E               HL<-#OTHTAB#CALL SEARCH
26  03C7 200B                       BR NZ,OTH.BX
27  03C9 3A8811FE20                 A<-TERMIT#CMP#'
28  03CE CC3601                     CALL Z,GETATM#I         STEP OVER THE SPACE
29  03D1 0600                       B<-#0#I NEEDED FOR LOGICAL OP, SET AND SIMILAR
30  03D3 E9                         J (HL)#I        OFF TO IT, MY BOY
31  03D4 CDED12CD160F       OTH.BX: CALL SYNTER#CALL GETOUT
32  03DA CDFD11             OTH.0X: CALL OUTINS
33  03DD E1D1C1F1C9         OTH.GX: POP HL,DE,BC,AF#RET
34  03E2 CD810F20ED             RST:    CALL EVALUA#BR NZ,OTH.BX
35  03E7 7DFE08F2D403               A<-L#CMP#10#J P,OTH.BX#I        ONLY RST 0 TO 7
36  03ED 87                         ADD A#ADD A#ADD A#I     3 LEFT SHIFTS
37  03F0 F6C7                       OR #307#I       MAKES THE CODE
38  03F2 18E6                       BR OTH.0X
39  03F4 CD8703280D             CALL:   CALL COMCHK#BR Z,CALLCC#I       CONDITIONAL CALL?
40  03F9 CD810F3ECD                 CALL EVALUA#A<-#315
41  03FE CDFD11                     CALL OUTINS
42  0401 CDF809             CALL.1: CALL OUTHL#I    OUT WITH THE ADDRESS
43  0404 18D7                       BR OTH.GX
44  0406 217708CDD30E20C6       CALLCC: HL<-#CONTAB#CALL SEARCH#BR NZ,OTH.BX#I  CHECK FOR GOOD COND.CODE

45  040E 7CF6C4CDFD11               A<-H#OR#304#CALL OUTINS
46  0414 CDAE0F                     CALL GEVALU
47  0417 18E8                       BR CALL.1
48  0419 CD8703280C         RET:    CALL COMCHK#BR Z,RETCC
49  041E 36C918B8                   A<-#311#BR OTH.0X
50  0422 217708CDD30E20AA       RETCC:  HL<-#CONTAB#CALL SEARCH#BR NZ,OTH.BX
51  042A 7CF6C0                     A<-H#OR#300
52  042D 18AB                       BR OTH.0X
53  042F CD8703CAD403       DBNZ:   CALL COMCHK#J Z,OTH.BX
54  0435 3E10CDFD11                 A<-#20#CALL OUTINS
55  043A CD4004C3DD03               CALL BROUT#J OTH.GX
56  0440 E5D5F5             BROUT:  PUSH HL,DE,AF#I DOES HARD BIT OF BRANCH
57  0443 CD810FE5D58681513          CALL EVALUA#DE<-CURADD#INC DE

 1  044B B7ED52                     TSTA#SBC HL,DE
 2  044E 7C8728D5                   A<-H#TSTA#BR Z,BRO.1
 3  0452 2F87C4F512                 CMA#TSTA#CALL NZ,TRERR#I        IF OUT OF RANGE
 4  0457 7DCDFD11           BRO.1:  A<-L#CALL OUTINS
 5  045B F1D1E1C9           BRORET: POP AF,DE,HL#RET
 6  045F CD8703280B         RR:     CALL COMCHK#BR Z,BR.1#I LIKE CALL, REALLY
 7  0464 3E18CDFD11CD4004           A<-#30#CALL OUTINS#CALL BROUT
 8  046C C3DD03                     J OTH.GX
 9  046F 217708CDD30E       RR.1:   HL<-#CONTAB#CALL SEARCH
10  0475 C2D403                     J NZ,OTH.BX
11  0478 E604C2D403                 AND#4#J NZ,OTH.BX
12  047D 3E20B4CDFD11               A<-#40#OR H#CALL OUTINS
13  0483 CD3601CD4004               CALL GETATM#CALL BROUT
14  0489 C3DD03                     J OTH.GX
15  048C                        J:
16  048C 3A8811FE28201D             A<-TERMIT#CMP#'(#BR NZ,J.1
17  0493                                    ;IT'S J(
18  0493 CD360121CF0CCDD30E         CALL GETATM#HL<-#DRTAB#CALL SEARCH
19  049C FE02280B                   CMP#2#BR Z,J.2
20  04A0                                    ;SHOULD BE IX OR IY
21  04A0 E608CAD403                 AND#10#J Z,OTH.BX
22  04A5 3EDDB4CDFD11               A<-#335#OR H#CALL OUTINS
23  04AB 3EE9               J.2:    A<-#351
24  04AD C3DA03                     J OTH.0X
25  04B0 CD8703280E         J.1:    CALL COMCHK#BR Z,J.3
26  04B5 3EC3CDFD11                 A<-#303#CALL OUTINS#I   NOT CONDITIONAL
27  04BA CD810FCDF809               CALL EVALUA#CALL OUTHL
28  04C0 C3DD03                     J OTH.GX
29  04C3 217708CDD30FC2D403 J.3:    HL<-#CONTAB#CALL SEARCH#J NZ,OTH.BX
30  04CC 3EC2B4CDFD11               A<-#302#OR H#CALL OUTINS#I      IT IS CONDITIONAL
31  04D2 CDAE0FCDF809               CALL GEVALU#CALL OUTHL
32  04D8 C3DD03                     J OTH.GX
33  04DB CD810F7D           SETRAD: CALL EVALUA#A<-L
34  04DF FE02380A                   CMP#2#BR C,SETBAD#I     MUST BE>=2
35  04E3 FE113006                   CMP#21#BR NC,SETBAD#I   AND     <=16
36  04E7 323201                     CRADIX<-A
37  04EA C3DD03             RADRET: J OTH.GX
38  04ED CDF512             SETBAD: CALL TRERR
39  04F0 18F8                       BR RADRET
40  04F2 06C51802           PUSH:   B<-#305#BR PUPO
41  04F6 06C1               POP:    B<-#301
42  04F8 21CF0CCDD30FC2D403 PUPO:   HL<-#DRTAB#CALL SEARCH#J NZ,OTH.BX#I   ONLY REG. PAIRS
43  0501 E6082808                   AND#10#BR Z,PUPO.1
44  0505 3EDDB4CDFD11               A<-#335#OR H#CALL OUTINS#;IX OR IY
45  0508 2620                       H<-#40
46  050D 78B4CDFD11         PUPO.1: A<-B#OR H#CALL OUTINS
47  0512 CD8703CC360128DE           CALL COMCHK#CALL Z,GETATM#BR Z,PUPO#I   MORE?
48  051A C3DD03                     J OTH.GX
49  051D 04                 CMP:    INC B
50  051E 04                 OR:     INC B
51  051F 04                 XOR:    INC B
52  0520 04                 AND:    INC B
53  0521 04                 SBC:    INC B
54  0522 04                 SUBL:   INC B
55  0523 04                 ADC:    INC B
56  0524 CB28CB20CB20       ADDL:   SLA B#SLA B#SLA B
57  052A 21080DCDD30E2007           HL<-#BRTAB#CALL SEARCH#BR NZ,ADD.1
```

```
 1  0532 3E80#00B5          AC=#200#OR B#OR L
 2  0536 C3DAU3             J OTH,OX
 3  0539 3AB811FE232013     ADU:1:  AC=TERMIT#CMP#'#BR NZ,ADU,2
 4  0540 CDAE0FC4F512CD403          CALL GEVALU#CALL NZ,TRER#+J NZ,OTH,BX
 5  0549 3EC6B0CDFD11               AC=#306#OR B#CALL OUTINS
 6  054F 7DC3DAU3                   AC=L#J OTH,OX
 7  0553 21CF0CCDD30E2812  ADU:2:   HL<=#DRTA8#CALL SEARCH#BH Z,ADD,3
 8  0559 CD3108                     CALL HLIXI#I  MAKES HL, OFFSET(IX OR IY) LOOK LIKE (HL)
 9  055E 3E86B0CDFD11               AC=#206#OR B#CALL OUTINS
10  0564 7AB7CADDU3                 AC=D#TST#+J Z,OTH,GX
11  0569 79C3DAU3           ADU:3:  AC=C#J OTH,OX#I OUT WITH ANY OFFSET
12  056D CDB703C2D403               CALL COMCHK#J NZ,OTH,BX
13  0573 7DB620F2D403               AC=B#SUB#40#J P,OTH,BX
14  0579 EHCD3602                   EX DE,HL#CALL GETATM
15  057D 21CF0CD5CDD30ED1C2         HL<=#DRTA8#PUSH DE#CALL SEARCH#POP DE#J NZ,OTH,BX
16  0588 79FE10CAD403               AC=B#CMP#20#J Z,OTH,BX
17  058E B720E2             ADDADD  TSTA#BR Z,ADDADD
18  0591 7AFE20C2D403               IITS ADC OR SBC
19  0591 7AFE20C2D403               AC=D#CMP#40#J NZ,OTH,BX
20  0597 7DB604F2D403               AC=L#SUB#4#J P,OTH,BX
21  059D 3EEDCDFD11                 AC=#355#CALL OUTINS
22  05A2 79FE102806                 AC=B#CMP#30#BR Z,ADDSBC
23  05A7 3E4AB4                     AC=#112#OR H
24  05AA C3DAU3                     J OTH,OX
25  05AD 3E42B4C3UAD3       ADUSBC: AC=#104#OR H#J OTH,OX
26  05B3                    ADJADD: IDE CONTAINS FIRST REG PAIR CODE
27  05B3                            IHL THE SECOND
28  05B3 7BE6U828H                  AC=E#AND#10#BR Z,ADDA,1
29  05B8                            IIX OR IY
30  05B8 3EDDH2CUFD11               AC=#335#OR D#CALL OUTING
31  05BE 7UE6U828D7                 AC=L#AND#10#BR Z,ADD,U          ADD IY,IX E.G.
32  05C3 7UBBC2D403                 AC=L#CMP E#J NZ,OTH,BX#I        MAKE 'EM LOOK LIKE HL
33  05C8 2620               ADU:0:  AC=L#LC=#2#CMP L#J Z,OTH,DX
34  05CA 7UE2E02BUCAD403            ICAN'T ADD IX,HL#FOR EXAMPLE
35  05D1 1806                       BR .+1U#        ISAVES HAVING A LABEL
36  05D1 1806                       AC=E#CMP#2#J NZ,OTH,BX#I        MUST BE HL,SOMETHING
37  05D3 7DFE02C2U403       ADDA,1: AC=L#SUB#7#J P,OTH,BX#I CAN'T HAVE AF#IY#IX
38  05D9 7UU607F2U403               AC=#11#OR H
39  05DF 3E0984                     J OTH,JX
40  05E2 C3UAU3             TEST:   CALL COMCHK#J NZ,OTH,BX
41  05E5 CUB703C2D403               CALL EVALUA#AC=L#SUB#10#J P,OTH,HX
42  05EB CUB10F7UD6A0AF2D403        AC=L#AUD A#ADD A#ADD A
43  05F4 7UB787B7                   HC=A#CALL GETATM#HL<=#SRTA8#CALL SEARCH
44  05F8 47CD3601210A0UCDD3         BR NZ,TEST,1
45  0602 20UC                       AC=#313#CALL OUTINS#I   TEST X,R
46  0604 3EC8CDFD11                 AC=B#OR L#OR#100#J OTH,OX
47  0609 7B95F640C3UA03             AC=B#OR L#OR#100#J OTH,OX
48  0610 CD31U8             TEST,1: CALL HLIXI#I    NOT SO SIMPLE
49  0613 3EC8CDFD11                 AC=#313#CALL OUTINS
50  0618 7AB779                     AC=D#TST#+#AC=C
51  0618 C4FD11                     CALL NZ,OUTINS
52  061E 3E46H0C3UA03               AC=#106#OR B#J OTH,OX
53  0624                    SETF:   .BLK#1 TEMPORARY STORAGE
54  0625 06C01802           SET:    AC=#20U#BR SETRES
55  0629 0680               RESET:  BC=#20U
56  062B CUB703C2D403       SETRES: CALL COMCHK#J NZ,OTH,BX#I       QUITE LIKE TEST
57  0631 76U22406                   AC=B#SETFC-A
```

(To be continued)



SINTROM means business

From an established base in scientific, educational and personal computers using SWTPC 6800 and SI00 8080 equipment, Sintrom announces a range of small systems for the business user. Easy availability of Micropolis-compatible applications software plus CP/M, COBOL, APL, FORTRAN and BASIC allows a speedy implementation of the total system.

● Ledger ● Payroll ● Inventory ● Word Processing ● High Resolution Graphics ● Industrial Control

Single Disk 143K £453
Dual Disk 630K £1159
Include PSU, SI00 controller, Basic/MDOS.
Add-on 143K £399
Add-on 630K £859

Host S100 computer with 32K and 1/0 card
From £1200

Plus integral dual Disk
From £2200

ADM 3A VDU £620
Centronics μPrinter £350-£400
Centronics 701 £1400

Office Hours:
Monday - Saturday

Access/Barclaycard
Prices exclude VAT/CARR

Write for free catalogue

SINTROM GROUP

Sintrom Microshop
Arkwright Road,
Reading, Berks. RG2 0LS.
Tel: Reading (0734) 84322

BUZZWORDS

Peter Reynolds





Backplane – Effectively synonymous with *motherboard* though the slot connectors it holds for additional circuit or memory cards may be fixed to a chassis rather than a circuit board.

Back-up. A term with several connotations in a computer context; in each it is roughly synonymous with "reserve". Computer records can very easily be destroyed, and it is therefore common to keep important data in duplicate (or "back-up"). Similarly, computer manufacturers often arrange that every piece of apparatus used by a customer is matched (or "backed-up") by comparable apparatus (possibly that of another customer) which could, in emergency, take over the work load while repairs were effected. Such arrangements are particularly important when a new design of computer is on trial.

Backing Store. A computer store of larger capacity than the working store or memory but of slower access. Floppy discs and magnetic tape cassettes are popular media for backing storage in personal computers.

Bank (e.g. data bank) – A large supply held available for ad hoc use as the demand arises.

Base. (1) The base of a numbering system can be considered as the quantity one greater than can be represented by any digit in that system. Thus the base of the familiar decimal system, using the digits 0 to 9, is 10 and the binary system, whose digits are limited to 0 and 1, has the base of 2. Octal has base 8 and hexadecimal has base 16.

(2) One of the elements in a transistor, the others usually being *collector* and *emitter.*

Base Address. An address identification from which modified or relative addresses are subsequently calculated.

BASIC – Beginners' All-purpose Symbolic Instruction Code. A high-level programming language widely used with small computers. The original version was designed at Dartmouth College (USA) but many enhanced or extended versions have been introduced by manufacturers of particular machines. For further description see PCW Volume 1, Number 2, page 20.

Batch Working – In commercial computing, where large quantities of data have to be processed, this is a common method of organising the work. Data is collected over a convenient period, for example, a day, and then the whole group (or batch) is put on to the computer, with or without first being sorted into some logical order. The opposite of *on-line* or *interactive* working.

Baud. A measure of the capacity of a telegraph line or other signal channel to carry information; synonymous with *bits* per second. Physical characteristics of the conductor limit the speed at which data can be transmitted without suffering corruption. Named after Baudot, a pioneer in telegraphy. A transmission speed of 300 baud is approximately equivalent to 30 characters per second.

B.C.D. Binary Coded Decimal (abbreviation). A system of express decimal quantities in which each digit of the decimal number is translated into a binary equivalent, instead of taking the binary equivalent of the number as a whole. Thus the number 23 could be represented in B.C.D. as 0010 0011, whereas in binary notation 23 is 10111. See also *excess-3 code.*

BCS. British Computer Society (abbreviation).

Benchmark. A set of test problems designed to be worked upon by a computer so that the time taken can be used for comparing the performance of a number of different computers. See, for example, PCW Volume 1, Number 1, pages 57/8.

Billion. A large number; in English usage 10^{12}, that is 1 000 000 000 000, but in America 10^{9}, or 1 000 000 000. Because of this ambiguity "billion" is a word best avoided in computer practice.

Binary. A system of counting in a scale of two instead of the more familiar decimal scale of 10. Thus the decimal values 0, 1, 2, 3, 4, 5, 6 etc. are expressed in binary form as 0, 1, 10, 11, 100, 101, 110 etc. The main reason why binary notation is universally adopted in digital computer design is that the electronic circuits upon which computers are based are particularly efficient when functioning in the bi-stable state; that is, a record, be it a magnetic pulse, a hole punched in paper or a current passing along a wire, is either present or not present. It is much easier to achieve a reliable record which can only be in one of two states than one which can be in any of ten conditions. A secondary reason for adopting the binary mode is that the logical decisions used in programs can easily follow the 'yes/no' indication shown by two-state circuitry. (PCW Volume 1, Number 3, page 16).

Binary Point. The binary equivalent of the decimal point. In binary, the point distinguishes between units on the left and halves on the right. Thus 101.1 binary equals 5.5 decimal; and 10.01 equals 2.25.

Binary Search. A technique for locating an item by successively halving the area under search. The method is applicable when items are in order but not necessarily in a complete sequence. For example, consider a search for one of 640 customers identified by four-figure account numbers listed in ascending order. The mid-item would be looked at first. Suppose it held too large an account number. The next inspection would be the 160th item and that might be followed by the 240th, the 280th, the 260th, the 270th, the 265th, the 263rd; until finally the required account number is found in the 264th item. Synonymous with *dichotomising* search.

Biquinary Code. A four-digit code, based on the binary symbols 0 and 1, in which 0 in the first position represents 0, and 1 in that position represents 5. The remaining three positions denote the usual binary values from 0 to 4. A feature of biquinary notation is that each of the values 0 to 9 is represented by four binary digits.

Biquinary/Binary/Decimal Equivalents

Decimal	*Biquinary*	*Binary*
0	0 000	0
1	0 001	1
2	0 010	10
3	0 011	11
4	0 100	100
5	1 000	101
6	1 001	110
7	1 010	111
8	1 011	1000
9	1 100	1001

Bistable. Capable of assuming either of two stable states. Thus a bistable switch (the normal household electric light switch is an example) may be either on or not on. Similarly, a spot on a magnetic storage device may be either magnetised or not magnetised.

This is the physical principle upon which all digital computers are currently based.

Bit. Contraction of "binary digit". Each component of a binary number (that is, each 0 and each 1) is known as a bit, so 1001 is a 4-bit number.

Bit-Adder — Same as *half-adder,* a circuit for limited addition of binary numbers. See PCW Volume 1, Number 1, page 33.

Black Box. An approach to computing which accepts predictable results without seeking to understand how they are achieved.

Block. A set of associated computer words or characters handled as a unit, usually for transfer between central processor and some peripheral device. For example, the information required to print one complete line on a printer (which might be 80 or 132 characters) could well be transferred to the printer as a block.

Block Length. The number of characters in a *block.*

Block Marker. An indicator at the beginning or end of a *block* (usually in the form of a magnetic pulse).

Blocking. (1) Combining two or more records into a *block.*

(2) Preventing the passage, e.g. of direct current in an alternating current circuit, by a device (in this case a capacitor) which opposes that current only.

Board. A sheet of electrically insulating material on which electronic components are mounted, e.g. mother-board, memory board.

Boolean Algebra (or Arithmetic). A system of mathematical logic named after George Boole, (1815-64), who devised a series of theorems based on operators such as *AND, OR, NOT* and *IF . . . THEN,* as opposed to the more common "plus," "minus" and "equals". Boolean arithmetic is inherent in the logic under which a computer will follow different instructions according to the result of a comparison of data.

Bootstrap. A short key program which instructs a computer to feed in more detailed instructions from some convenient input device, such as a magnetic tape, so that the whole configuration is brought to the desired state of readiness.

Bound. A computer configuration is said to be tape bound if its overall speed of operation is determined or limited by the speed of operation of the tape peripherals. It is processor bound if the processor is not capable of working as fast as its high-speed peripheral devices.

Branch. An optional sequence of program instructions following a *conditional jump* instruction.

Breadboard. A temporary, experimental circuit assembly. Boards are available with large numbers of sockets in which components may be inserted and interconnected by jumper leads.

Break. The name of a key and of the non-printing ASCII character it generates which normally aborts any program running when it is pressed. Some software allows a conditional jump to follow a break signal given at any time during the running of part of a program.

Breakpoint. A point at which a computer program may be interrupted by a special instruction for checking purposes. This facility is much used in program testing and debugging. The break instruction may be given by a *monitor* or *trace routine* or by manual operation of a switch.

B-Register. An optional name for the electronic register, more commonly termed an *index* or *modifier register,* used to modify addresses.

Bubble Memory. A very compact solid-state medium for data storage in which bits are represented by small magnetic domains or bubbles (typically 3 microns in diameter but possibly much smaller) on a garnet chip. A square inch garnet can store three million bits. The best-known initial application is in lightweight intelligent printing terminals by Texas Instruments.

Buffer. Subsidiary storage normally associated with input or output devices and the channels to which they may be connected. Its purpose is to provide a flexible link between devices working at different speeds or to hold data available for modification until the originator is satisfied that it should be transmitted to the processor. The buffer for a printer will typically hold one line of data, the buffer for a magnetic tape cassette may hold one block and the buffer for a VDU will typically hold at least one page or screen full of display.

Bug. A mistake or malfunction, (1) in the design of a routine or in the coding of a program, or (2) in the operation of a computer.

Bus. A set of electrical connections in a computer which connect to many components (e.g. elements of memory or some display device). Coded signals passing along a bus find their appropriate destination by electronic switching. Synonymous with *highway.*

Byte. An 8-bit sequence of binary code normally manipulated as an entity in an 8-bit computer system.

The world's best-selling personal computer



APPLE II

***New powerful basic and new graphics on Rom. £75 card.**
***Talk to Apple with voice recognition, speaker trained, 32 word vocabulary. £165 card.**
***Colour Graphics. High resolution 280h x 192v, 6 colours, easy-to-use. Low resolution 40h x 48v, 16 colours, very powerful.**
***Apple's disks. Powerful DOS. 116K bytes capacity, multiple drives, fast access. £395.**
***Use Apple as a computer terminal 110 or 300 BAUD. Full or half duplex or use with a Dec-writer. £95 card.**
***Use any 8 bit parallel printer with Apple 11. Print up to 3,700 lines per minute. 255 character lines, upper and lower case. £100 card.**

Personal Computers Limited

DISTRIBUTOR
18-19 Fish Street Hill, London E.C.3.
Tel. 01-623 1434
New address from August 21st
194-200 Bishopsgate, London E.C.2.

THE BIG APPLE 48K RAM-18K ROM £1850 incl.VAT

MoI MINE OF INFORMATION LTD MoI
1 Francis Avenue, St Albans, Herts AL3 6BL
..... *Tel. 0727 52801*.....

MICROCOMPUTER BOOKS!

RODNAY ZAKS of SYBEX INC.
C201 From Chips to Systems £7.50
C207 Interfacing Techniques 7.50

ADAM OSBORNE & ASSOCIATES
Introduction to Microcomputers
Vol. Zero The Beginner's Book 5.90
Vol. One Basic Concepts * 5.90
Vol. Two Some Real Products 10.90
* *MoI corrections list free with each copy.*

Assembler Language Programming
6800 ..... 6.50
8080A/8085 ..... 6.50

SCELBI SOFTWARE GOURMET SERIES
6800 Guide and Cookbook 7.90
8080 Guide and Cookbook 7.90

Best of CREATIVE COMPUTING
Volume One 6.90
Volume Two 6.90

101 BASIC Computer Games .....5 .40

More soon! Get on to our mailing list *now!*

For delivery outside UK add £1 per order (any quantity). Pay by Access, cheque or PO.
Prices include postage and fast service.

TIMETABLING FOR SCHOOLS

Charles Sweeten

The difficulty of writing a timetable can be summed up by saying that it is only when you have written 95 per cent of the timetable that you discover that it is impossible to write the other 5 per cent.

This happens as a result of either of two situations. The first arises when the initial scheduling is wrong in terms of the teacher allocation or the subject allocation. For example Mr A may be down for 42 periods in a 40 period week. Or class 3D may have only 36 periods in the 40 period week. It sounds stupid, but it is remarkably easy to do. The second situation arises when incompatible groups of teachers interact so as to go beyond the boundary of the teaching week. For example if Mr A and Mr B are teaching together in 3 blocks of 5 periods, and if Mr B and Miss C are teaching together in 3 blocks of 5 periods, and if Miss C and Mr A are teaching together in 3 blocks of 5 periods, then it is clear that none of this teaching can take place simultaneously and it will require 45 periods to accomodate it, in spite of the fact that each teacher is only teaching 30 periods.

The latter situation arises very frequently in a timetable, and this "interference" between blocks becomes very complicated when a sizeable school is involved. Would that all the "interference" patterns were as simple as the one above. I shall describe a method of examining this "interference" by means of a simple program that I have used for two years, first on a NOVA 2/10 minicomputer and then on a SWTPC 6800.

People differ in their use of terminology, so here is a list of what I mean by certain words:

Jargon

CURRICULUM	what is on the timetable as being taught.
FORM	a number of ~~pe~~pils of roughly the same age and with the same curriculum. Sometimes called a year group.
BLOCK	a number of subjects or sets which are timetabled at the same time as each other. Other subjects outside the block are not timetabled within the block. e.g. FORM 4 A E H Sc Sc M A A A Sc Sc E H M E PEPE H A A E M ScPEPE There are two blocks: one for Maths, one for P.E.
SET	a group of pupils, usually from the same form, who are taught a part of the curriculum together.
OVERLAP	the number of periods in a week that blocks occur at the same time.
INTERFERENCE	between blocks: the teachers teaching in one block are not available to teach in another block which occurs at the same time. Also the implications of shared sets etc.
FLOAT	the number of spare periods available to a teacher within a structure such as a block.

SYMBOLS USED

A	Art	IT	Italian
B	Biology	L	Latin
C	Chemistry	M	Maths
D	Divinity	MM	Further Maths
E	English	MUS	Music
EC	Economics/Political Science	P	Physics
		PHI	Philosophy
EL	Electronics	PSY	Psychology
EN	Engineering	S	Spanish
F	French	SC	Science
G	Geography	TD	Technical Drawing
GER	German	WS	Workshops
GK	Greek	PE	Physical Education
H	History		

HARDWARE REQUIREMENTS

System with room for BASIC + 4K user space.
VDU (preferably fast).
Printing output (*teletype is adequate).
Disk.

*(*Teletype is a TM of Teletype Corp.)*

SOFTWARE REQUIREMENTS

BASIC with string arithmetic and sequential disk data files. String matrices would speed the process up. EDITOR with macro commands or a powerful target/occurence structure.

In the description that follows I will refer to the SWTPC version of disk BASIC, which is painfully slow on string handling, but at the time of writing the only other BASIC around with these features is the APPLE II BASIC and that only got here on Monday, so I haven't had time to use it yet. (Note to TSC: please hurry up with your version of disk BASIC - if it is as nice as your other software then it is beautiful and I need it last week). I shall also be refering to TSC EDITOR, which takes some beating, and to the powerful TSC FLEX disk operating system.

PURPOSE OF PROGRAM

To assist the timetabler in the construction of the timetable.

To foresee points of difficulty and list teachers involved.

To identify awkward blocks and awkward teacher allocations.

To put into place the structure of the complete timetable by reducing the timetable to larger building blocks.

INFORMATION REQUIRED

Returns from Heads of Faculty or Department stating or confirming the teaching and curriculum required. These returns should list the teachers available and the agreed

allocations for each. They should list the available teachers against the list of groups and the total allocated should be shown to agree with the available allocations.

An example is shown in figure 1 of an allocation sheet from one department in my school.

MANPOWER ALLOCATION 1978/9

department : History and Economics

	block	set	periods	teacher
2nd yr 6th	A	ec1	3	JMM
			3	PMH
		ec2	3	JMM
			3	DJS
	B	h1	5	AM
			3	AMM
		h2	5	DJW
			3	PMH
	C	h3	5	PMH
			3	GDW
		h4	5	DJS
			3	GDW
		ec3	6	JMM
1st yr 6th	A	ec1	3	JMM
			3	DJW
	B	h1	6	PMH
		h2	6	AM
	C	h3	6	DJW
		h4	6	GDW
		ec2	6	DJS
		ec3	3	JMM
			3	KMBM
	F	ec4	3	JMM
		ec5	3	DJS
	G		3	KMBM
			3	GDW
			3	AMM
			3	AM
5th	A	h1	4	GDW
	B	h2	4	AMM
		h3	4	DJW
	C	h4	4	AM
		h5	4	DJS
		h6	4	KMBM
etc		etc		etc

	Form							
	3rd yr	2nd yr	1st yr					
teacher	6th	6th	6th	5th	4th	3rd	2nd	1st
AM	7	5	9	4				
AMM	3	3	3	4	6			
JMM	4	12	9					
PMH	3	11	6		6			
DJW		5	9	4	3	4	2	
GDW	3	6	9	4	3	4		
DJS		8	9	4	3	6		
KMBM			6	4	3	2		
GDA							2	4
etc								

Figure 1

METHOD

The first job is to transfer the information from the department allocation sheets onto disk files. These are organised by blocks. Each file will contain a list of the teachers teaching in one particular block, listed against the set that they are teaching and with the number of periods that they are teaching. Each file will be named after the block that it represents, and there will be a first entry in the file that gives the number of periods that the block covers. The files are created using EDITOR and it is easiest to put in the set labels at an early stage on their own to facilitate checking the required curriculum. At a later stage you go through each file, appending the teacher allocated, as the department returns come in. The file names must start with a letter and in my school I have used C6 for the 2nd year 6th, L6 for the 1st year 6th, F5 for the 5th year, and etc.

I have shown a selection of the block files that I used this year in figure 2. The files have to be organised in a way that the BASIC program can read, which is the reason for commas rather than tabulation.

It is now necessary to run the program. This will compare the entries in each block with the entries in every other block and calculate the interference. In order to deal with special cases, it will print every case of

```
BLOCKS

2nd yr 6th block C
8
P1,GLJ,8
P2,JFF,7
P3,CT,7
C5,GH,8
C6,ASN,8
M6,HJM,4
M6,RIB,4
M7,DLE,4
M7,JST,4
H3,PMH,5
H3,GDW,3
H4,DJS,5
H4,GDW,3
EC,JMM,6
A,JB,3
A,B,3
F1,EGB,4
F1,RWV,4
F2,JNB,4

2nd yr 6th block E
3
E11,RGF,3
E12,RJF,3
etc
.
.
.
.
.
1st yr 6th block A
8
M1,DLE,4
M1,MAB,4
M2,HJM,4
M2,RGW,4
M3,PPC,4
M3,JM,4
M4,MAB,4
M4,JS,4
M5,JST,4
M5,DLE,4
E1,RGF,6
E2,MBC,6
EC1,JMM,3
EC1,DJW,3
B1,IFT,4
B1,NWO,4
B2,JFH,4
B2,JFB,4
B3,IFT,4
B3,JAB,4
GER,MJLP,7
L,DHF,4
L,JVN,3

1st yr 6th block B
7
MM,JCBS,4
P4,CT,7
P5,PEB,7
C1,MTT,6
C2,WJUW,6
C3,JWF,6
C4,GK,6
G,RFSC,3
G,IDFC,3
E3,CGJ,6
E4,JDS,6
H1,PMH,6
H2,AM,6
GK,PHB,4
GK,DJH,3

1st yr 6th block C
7
P1,RBO,7
P2,JSL,7
P3,JFF,7
C5,ASN,6
C6,GH,6
M6,RIB,4
M6,HJM,3
M7,JS,4
M7,JM,3
H3,DJW,6
H4,GDW,6
A,JB,3
A,B,3
F1,JNB,4
F1,MND,4
EC3,JMM,3
EC3,KMBM,3
EC2,DJS,6

1st yr 6th block E
3
E11,NGS,3
E12,JDS,3
E13,MDA,3
E14,NAB,3
E15,MBC,3
E16,RJF,3
M11,RGW,3
M12,PPC,3
M13,BVH,3
.
.
.
.
.
.
.
.

5th block A
5
GER1,MJLP,5
GER2,JSW,5
GK,JVN,5
B1,JFB,4
B2,NWO,4
B3,IFT,4
B4,JFH,4
H1,GDW,4
MUS,TCB,4
A,B,4

5th block B
4
L1,RMA,4
L2,DHF,4
L3,PHB,4
L4,ECS,4
H2,AMM,4
H3,DJW,4
G1,AJAL,4
G2,IDFC,4

5th block C
4
B5,JAB,4
B6,FO,4
H4,AM,4
H5,DJS,4
H6,KMBM,4
G3,RFSC,4

4th block A
5
GER1,MJLP,5
GER2,JSW,5
GK,DJH,5
G4,AJAL,3
A1,JB,3

4th block B
4
L1,DHF,4
L2,ECS,4
L3,RMA,4
L4,PHB,4
L5,DJH,4
L6,JVN,4
G1,AJAL,3
G2,IDFC,3
G3,RFSC,3

.
.
.
.
```

Figure 2

interference. Lines 40 and 50 contain the names of the files of block information that are going to be examined. The names of the files are kept to 3 characters so that this can be done automatically. The method of reading each line from the file separately is very inefficient, but with this BASIC it is still quicker than reading the whole file and disentangling the resulting long string. Yes, I know I should have written a machine level subroutine, but this problem did not arise on the NOVA, and by the time I got down to using the SWTPC system there was no time left to improve the program.

VARIABLES USED

A$	contains list of file names for block 1
B$	contains list of file names for block 2
F$	file name of block 1
G$	file name of block 2
A	counter for going through A$
B	counter for going through B$
L1	number of periods in block 1
L2	number of periods in block 2
S$	set name in block 1
T$	set name in block 2
M$	teacher name in block 1
N$	teacher name in block 2
P	period allocation for teacher in block 1
Q	period allocation for teacher in block 2

The first run of the program will produce lists of interferences as given in the first two columns of figure 3. I have not given the full listing for my school, but have selected some of the interferences which led to special problems. In the print out, the first line means that

teacher JS is teaching set M1 in the 2nd year 6th block A for 4 periods and set M4 in the 1st year 6th block A for 4 periods. Since the 2nd year 6th block is 8 periods long, and the 1st year 6th block is 9 periods long, the maximum overlap is

9 + 8 − 4 − 4 = 9 periods.

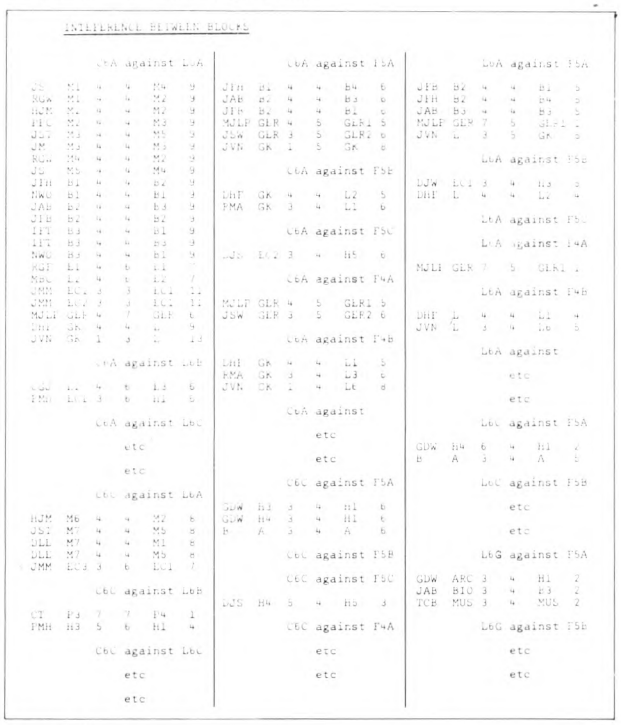


Figure 3

Further down we see that MJLP, who teaches GER in both blocks, appears to cause the greatest restriction.

6th GER ****−−−−−
L6th GER −*******

It would be perfectly simple to do this job by hand, but the process is tedious, and it is all too easy to miss just one interference. This is enough to make the final 5 per cent of the timetable impossible. Also, without some print out to look at, it is extremely difficult to see the more complicated interferences. Let us look at some of these now.

WORKED EXAMPLE

I will now apply these to the examples of interference that I have given in figure 3.

In C6A against L6A we see an example of Case 1 where JS teaches two sets in C6A and the maximum overlap is reduced to 5 periods. This gives us a list of teachers whose float would be reduced to zero if this overlap was used: JS, RGW, MAB, HJM, NWO, IFT, JFH, JAB, JFB.

It should be obvious that the smaller the float of a teacher is, the harder it is to fit in that teachers' timetable. So our object is to arrange the blocks in an overlapping pattern that maximises the float of the teachers in the school.

In C6A against F5A we see an example of Case 2 where MJLP and JSW share the GER set in C6A, and teach parallel sets in F5A, and the maximum overlap is reduced to 2. The Biology is also critical and the list of critical teachers is MJLP, JSW, JAB, JFB. The latter is an example of Case 3.

In C6A against L6A and C6A against F5A we see an example of Case 5 that would be particularly hard to spot without the interference printout. If C6A overlaps F5A by 1 period, which seems entirely reasonable, then C6A can only overlap F5A by 5 periods as far as the German is concerned.

It can be seen that cases of multiple interference can be extremely complicated. Consider again C6A against L6A and look more closely at the Biology.

Put NWO = 1, JFH = 2, JAB = 3, JFB = 4, IFT = 5

C6A	B1	1	1	1	1	2	2	2	2	−		
	B2	3	3	3	3	4	4	4	4	−		
	B3	5	5	5	5	1	1	1	1	−		
L6A	B1					5	5	5	5	1	1	1	1
	B2					·	·	·	·	4	4	4	4
	B3					3	3	3	3	5	5	5	5

But JFH has to go where the dots are. Thus the maximum overlap is really only 1.

For a case of multiple overlaps, consider PMH in combination with JMM.

```
          SPECIAL CASES OF INTERFERENCE
          -----------------------------


     Notation : 6/A9/T/S/4 represents teacher "T"
teaching subject "S" in Form 6, Block A for 4
periods, where Block A is 9 periods long.
          : T1,T2,T3 are teachers
          : S1,S2,S3 are sets
          : A,B,C,D,E,F,G,L,M are blocks
          : 6,L6,5,4,3,2,1 are forms


Case 1
        6/A9/T1/S1/4          L6/A8/T1/S2/4
                              L6/A8/T1/S3/4
The maximum overlap of 6/A and L6/A is
        9 + 8 - 4 - 4 = 5 periods.

        6/A  S1  ****------
       L6/A  S2      ****----
       L6/A  S2      ----****

Case 2
        6/A9/T1/S1/4          5/A5/T1/S2/5
        6/A9/T2/S1/4          5/A5/T2/S3/5
The maximum overlap of 6/A and 5/A is
        9 + 5 - 4 - 4 - 5 = 1 period

        6/A  S1  11112222-
        5/A  S2          11111
        5/A  S3          22222

Case 3
        6/B9/T1/S1/4          5/B5/T1/S2/4
        6/B9/T2/S1/4          5/B5/T2/S3/4
The maximum overlap of 6/B and 5/B is
        9 + 5 - 4 - 4 - 5 + (5-4) + (5-4) = 3 periods

        6/B  S1  11112222-
        5/B  S2     -   1111
        5/B  S3     2   -222

Case 4
        6/A9/T1/S1/8          L6/A8/T1/S3/5
        6/B9/T1/S2/7
The maximum overlap of 6/B and L6/A is
        9 + 8 - 7 - 5 - ((actual overlap of 6/A and L6/A)
                                              - 9 + 8)
The last term is omitted if it is negative.

                    AAAAAAAAA  BBBBBBBBB
        6th         *********-  --********
       L6th                 --******-
                            AAAAAAAAA

Case 5
        6/C8/T1/S1/5   L6/C7/T1/S2/6   5/C4/T1/S3/4
The maximum overlap of 6/C and L6/C          is 4
                       6/C and  5/C          is 3
                      L6/C and  5/C          is 1
                       6/C and L6/C and 5/C  is 1

        6/C  S1     *****---
       L6/C  S2           -*******-
        5/C  S3          **-     **

        Various combinations are possible.
```

COMPLETING THE TABLE

I now come to the next stage, which is to enter the maximum overlaps in a table. The method I use is to tabulate the C6 against the L6, as these are my most difficult forms. In each square I list the teachers who are critical to that overlap. In my case all the periods of C6 and L6 must overlap, and I enter a cross where an overlap occurs. The totals of the columns and the rows

must add up to the correct totals for that block. This is a process that I have always found easiest to do by hand, as I can then try to keep track of the multiple interferences, and try to ensure that the float of any one teacher remains as much as possible. I have written programs to do the job in the past, but they have only proved successful where there has been no interference. Figure 4 shows the table with the interferences entered, and figure 5 shows it after completion.

	C6A 9	C6B 9	C6C 8	C6E 3	C6F 2	C6L 2	C6W 3
L6A 7	1 NWO JFH JAB JFB IFT	5 MAB DLE	4 DLE JST MAB	2 RGF MBC			
L6B 7	6 CGJ PMH MBC JMM	2 JWF WJUW MTT	1 CT	1 JDS	1 WJOW		0 CT PEB
L6C 7	4 JS JCBS JM RGW JST	2 NAB RBO	1 GH ASN			1 PMH	
L6E 3	1 RGW	1 NAB MDA JDS NGS		0 NAB MDA JDS NGS RJF MBC RGF			0 RGW
L6F 3	3 JMM	1 AGR RWV	0 EGB		0 EGB		0 JAC
L6G 4	2 NWO	5	1 RWV JNB		1 JNB JAS RRA JAB		1 NWO
L6M 2		2 JAC					0 JAC
L6S 3		2 JAC	1 CT				0 JAC

Figure 4

	C6A 9	C6B 9	C6C 8	C6E 3	C6F 2	C6L 2	C6W 3
L6A 7	1	5 XXX	4 XXX	2			X
L6B 7	6 XXXX	2 X	1 X	1	1	X	0
L6C 7	4 XX	2 X	1	X	X	1	XX
L6E 3	1	1	XX	0	X		0
L6F 3	3 X	1	0	X	0	X	0
L6G 4	2	5 XXX	1	X	1		1
L6M 2		2	XX				0
L6S 3	XX	2 X	1				0

Figure 5

The next stage is to take account of the other forms and this is difficult in a 2 dimensional table. What I do is complete tables as in figure 6 and place these against the original table, as in figure 7. In order to do this I change the program as follows:

```
40   A$  = "L6AL6BL6CL6EL6FL6GL6ML6S"
50   B$  = "F5AF5BF5CF4AF4BF4MF3MF3B"
60   FOR A=1 TO 22 STEP 3
100  FOR B=1 TO 22 STEP 3
```

And then RUN again.

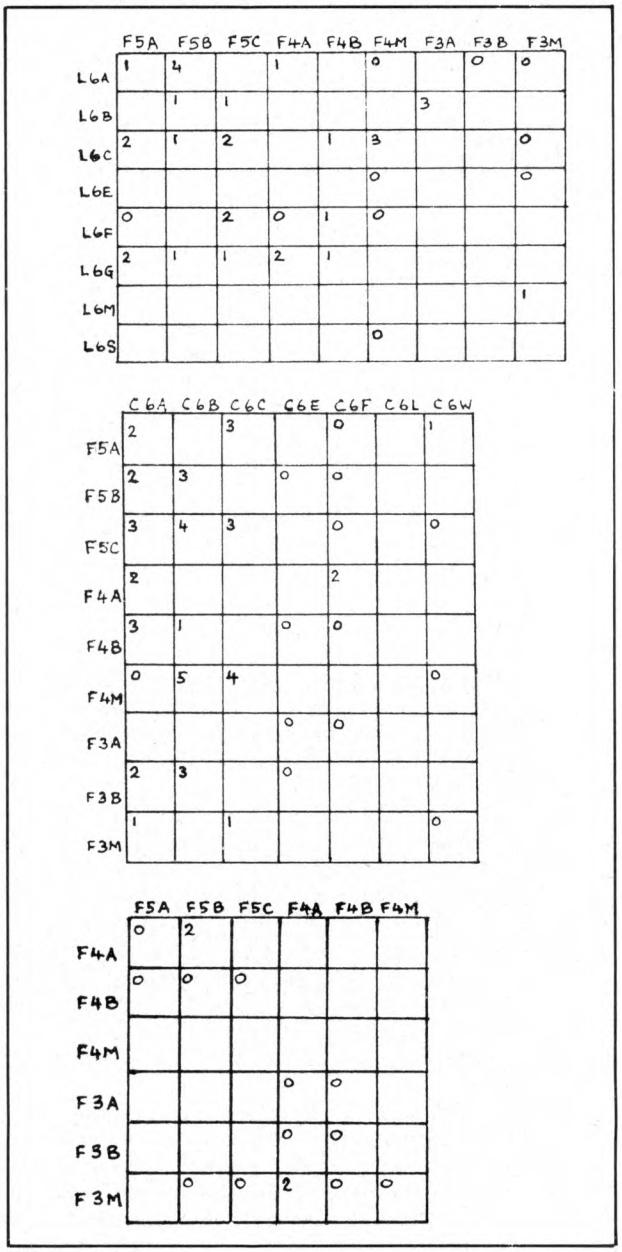


Figure 6

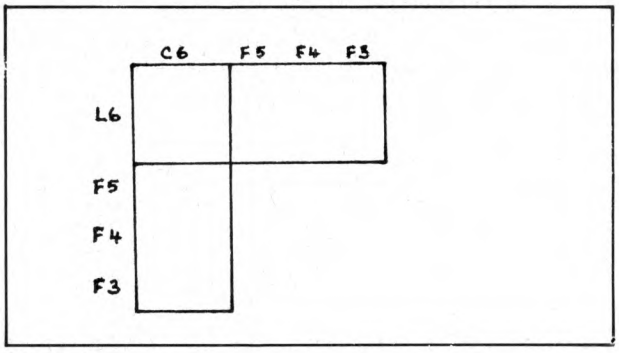


Figure 7

I then try to enter the possible overlaps for forms 5, 4 and 3 (which is where our main junior blocks occur), onto the crosses in the main table. This frequently necessitates some adjustment in the position of the crosses. The final table appears as in figure 8.

	C6A	C6B	C6C	C6E	C6F	C6L	C6W	
L6A		XXX AAA	XXX BBB				X B	5th 4th 3rd
L6B	XXXX B	X A A	X A M B			X A M B		5th 4th 3rd
L6C	XX C M	X A M		X C M	X		XX A	5th 4th 3rd
L6E			XX AA		X A			5th 4th 3rd
L6F	X C B			X C		X B		5th 4th 3rd
L6G		XXX MM ABB		X M				5th 4th 3rd
L6M			XX A M					5th 4th 3rd
L6S	XX BB	X A M						5th 4th 3rd

Figure 8

Once this has been done, I have merely to shuffle the 36 periods, with their overlaps fully specified, so as to get a reasonable balance to the week. I can then be sure that when I fill in the details into the blocks, everything will fit with no difficulty.

Well I didn't believe that either as I struggled with the entries this year. Then I found that I had put someone down for 4 periods instead of 7 and my interference table was up the spout. I had to start again, which is the time table's nightmare. But having got the data right second time around, it took only 7 working days to complete a 2000 entry timetable, including the computer work. I'm not claiming any records, but that was certainly quick going compared with my efforts of two years ago and back, which were of the order of 25 working days.

Now I am aware that some schools design their timetables in 20 minutes on the back of an envelope. This is done by dividing the teaching into the same number of faculties as there are year groups and then allocating equal time to each faculty. For this purpose they ignore the 6th form which is often rather small. It is then easy to draw up a timetable such as figure 9 and require the faculties to allocate the teaching themselves in such a way that it will fit. Every timetabler should ask himself whether such a method could possibly work in his school, for it would save many hours and ulcers.

2nd yr 6th	B1	B2	B3	B4	B5	
1st yr 6th	B1	B2	B3	B4	B5	
5th	F1	F2	F3	F4	F5	
4th	F2	F3	F4	F5	F1	
3rd	F3	F4	F5	F1	F2	
2nd	F4	F5	F1	F2	F3	
1st	F5	F1	F2	F3	F4	
periods	8	8	8	8	8	= 40

Figure 9

For those less fortunate who have to do the job by hand, I am suggesting that we make available to each other as many helpful programs as possible now that the micro computer is appearing in many schools. M.U.S.E. will publish and distribute material either by publishing the listings or by publishing a short description, and providing names and addresses of authors. Naturally it is useful if authors can provide notes on how to use their masterpieces.

For those interested in finding out about the current systems for writing timetables on large machines with large expensive programs (program £7000 plus £250 per run was one example), here is a list of addresses:

L.A.M.S.A.C. 3 Buckingham Gate, London SW1E 6JH
(they have conducted an evaluation of the NOR-DATA, the O.S.A., and the S.P.L. systems on behalf of L.E.A.'s)

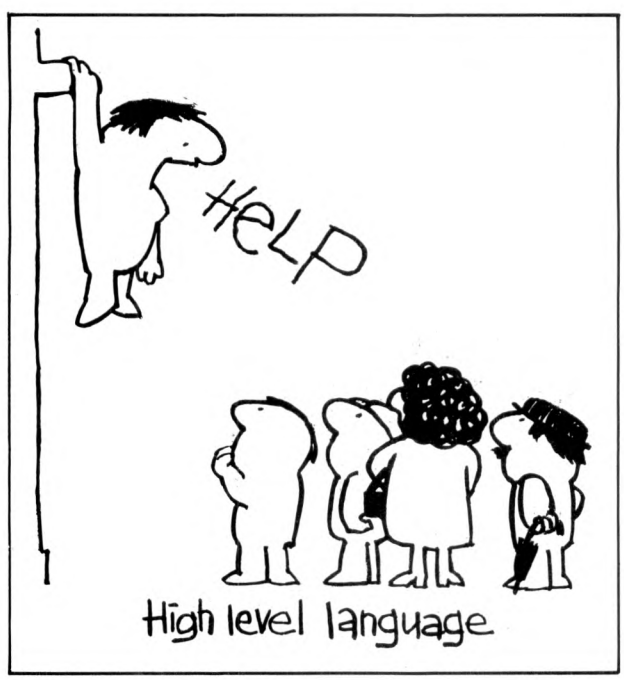
R.I.P.A. Hamilton House, Mabledon Place, London WC1H
(they market the version of the NOR-DATA system which was developed for U.K. schools by S.T.A.G.)

Oxford Systems Associates Ltd. Balliol College, Oxford
(they market the O.S.A. system)

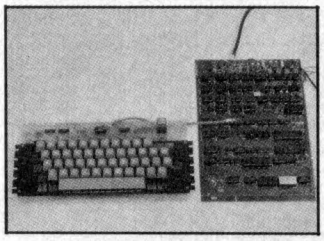
N.C.C. Ltd. Prudential Buildings, Colmore Row, Birmingham
(they market the S.P.L. system)

PROGRAM

```
0010 REM***BLOCK INTERFERENCE**********
0020 REM****CHARLES SWEETEN ***14/6/78*
0030 REM******SWTPC DISK BASIC*********
0040  A$="C6AC6BC6CC6EC6FC6LC6W"
0050  B$="L6AL6BL6CL6EL6FL6GL6ML6SF5AF5BF5CF4AF4BF4MF3MF3AF3B"
0060 FOR A=1 TO 19 STEP 3
0070  F$=MID$(A$,A,3)+".TXT"
0080 OPEN #1,F$
0090 READ #1,L1
0100 FOR B=1 TO 49 STEP 3
0110 PRINT
0120 PRINT " ",MID$(A$,A,3);" against ";MID$(B$,B,3)
0130  G$=MID$(B$,B,3)+".TXT"
0140 OPEN #2,G$
0150 READ #2,L2
0160 READ #1,S$,M$,P
0170 IF EOF(1)#1 THEN 250
0180 RESTORE #2:READ =2,L2
0190 READ #2,T$,N$,Q
0200 IF EOF(2)=1 THEN RESTORE =2:GOTO 160
0210 IF M$<>N$ THEN 240
0220 PRINT M$;TAB(6);S$;TAB(11);P;TAB(15);Q;TAB(19);T$;
0230 PRINT TAB(24);L1+L2-P-Q
0240 GOTO 190
0250 RESTORE #1
0260 READ #1,L1
0270 CLOSE #2
0280 NEXT B
0290 CLOSE #1
0300 NEXT A
0310 STOP
```







NASCOM 1
Z80 MICROCOMPUTER KIT

Includes interface for:
TV or Monitor – cassette, dump – Teletype – spare, PIO,
Expansion RAM-BOARD up to 32K.

Potential:
High Level Language, Mini Floppy Disk, Excellent Z80 evaluation kit.

Price £197.50 + VAT @ 8%
Access & Barclaycard welcome. Telephone Orders Accepted. Callers Welcome.

Write for details or send order to:
STRATHAND, 44 St. Andrews Square., Glasgow, G1 5PL. Tel: 041-552 6731/2

PET Software

Some of the sixty-plus titles in our current catalogue:—

Stock Portfolio,
Sales Analysis,
Tax 78/79
Stock Control,
Assembler/Editor,
Line Renumber,
Memory Diagnostic,
Peek & Poke,
File Handling,
PET Basic Tutorial,
Backgammon,
Super Startrek,
Plus many more, priced from £3.

Credit Cards accepted.

If you are a PET owner and would like to receive regular copies of our monthly catalogue, send an S.A.E. to:

Petsoft

PO Box 9, Newbury, Berks. RG13 1PB
Tel. 0635-201131 01-352 1100 Telex 8951672

MOTOROLA 6800 COMPUTERS

Single Board 6800.17 command Mikbug compat monitor, 1K crystal controlled VDU, CUTS, Buffered, V regs, room for expansion. £185.00

Mod. 2 SBC. Similar to above but with good quality QWERTY keyboard. £205.00

NEW. MINI 6800. Purpose designed for home computing. Has VDU, CUTS, RAM, IK MONITOR, QWERTY KEYBOARD.
£145.00. £145.00. £145.00.

VDU Kit. VDU section of above computers available separately. £60.00.

All prices are plus VAT and Post. Please send SAE for leaflets. Mail order only.

HEWART MICROELECTRONICS.
95, Blakelow Road, Macclesfield, Cheshire.



The Computer Magazine You Can Read!

You don't have to be a programmer or computer scientist to read Personal Computing. It's the magazine that tells you how to get started in computing. How to have fun. And how to use the computer for your own personal and business applications.

Each month Personal Computing is loaded with practical, fun articles designed to help you get the most out of your computer. We're the magazine with style, color and practicality. Topics covered in recent issues include Getting Into Computer Games, Using the Computer to Manage a Drug Store, Using Computers in Schools, Learning to Program in Three Easy Lessons, Ten Easy Steps to Become a Computer Hobbyist, How to Set Up a Word Processing System, The Future of Robots, A Computer That Speaks English, Intelligent Video Games, How to Profit from Your Computer Hobby, and much, much more.

Open your eyes, put on your thinking cap and subscribe to PERSONAL COMPUTING!

NAME

ADDRESS

CITY POST CODE

UK
☐ 1 year (12 issues) £16.00

Charge my
☐ Barclaycard
☐ Access
☐ American Express
☐ Diners Club
Account #
Card Expiry Date
Signature

OVERSEAS
☐ 1 year (12 issues) £17.00
(Please remit payment in UK funds on UK Bank)
☐ Cheque enclosed

BACK ISSUES £1.75/COPY
(Payment must accompany order)

Trade Enquiries Welcome

Mail to:
L. P. ENTERPRISES
313 Kingston Road,
Ilford, Essex, IG1 1PJ England

electronics today international AND TRANSAM

Jointly and proudly present the:

TRITON ONE-BOARD COMPUTER



Jointly sponsored by Transam and ETI Magazine, the Triton Computer will be available as a kit (costing about £300) from Transam while full constructional and software details will be published in Electronics Today International's November issue, on sale October 6th.

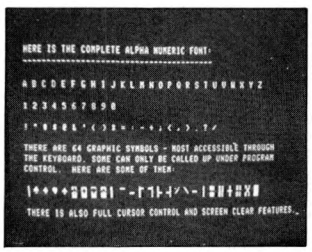
The Triton is designed by Mike Hughes and is genuinely believed to be a big step forward in the field of home computing. It is built on a single board and is based on the 8080 MPU.

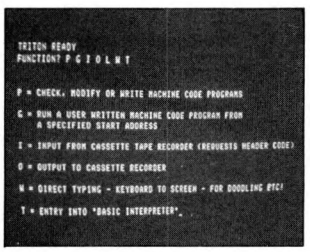
It features a very powerful 2K integer TINY BASIC as well as a versatile monitor to allow machine code programming. Used with a standard TV and cassette recorder it forms a powerful computing system, especially for domestic and educational applications. A unique VDU function, together with 64 graphics characters provide the ETI-Triton with excellent graphics handling capacity as well as full cursor control.

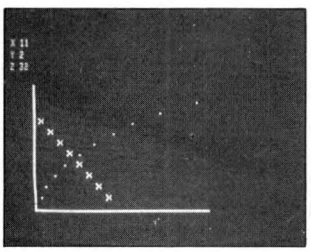
Other features include: BASIC command look-up table extension; named tape file search; memory-mapped VDU and full textual prompts throughout the software.

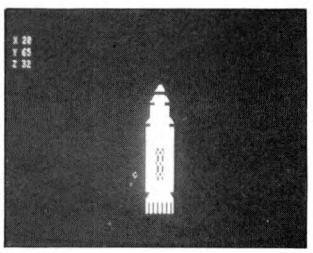
Practically everything is on one board which will hold up to 3K of RAM and from the start we've co-operated with Transam Components to have a kit available from square one.

ETI is taking a major step forward by giving all constructional details in one month — there'll be no hanging about for months to see the end of the article!









STARTING IN NOVEMBER

computing today

The ever increasing actlvity in the personal computing field has led to the situation where ETI magazine can no longer devote enough space to it in the magazine as well as keeping up with the world of electronics in general.

From next month therefore ETI will be carrying a new regular, 24 page supplement 'COMPUTING TODAY', within ETI itself. This will be over and above the regular size, in fact the November issue will be at least 140 pages.

COMPUTING TODAY will be devoted to personal computing and the first issue will contain part one of a series on BASIC programming techniques, an article on machine code programming with the Triton, a report on the US East Coast computer show, a review of the NASCOM 1, a CUTS encoder project plus news and Softspot, our software section.

TRANSAM

TRANSAM COMPONENTS LTD.
12 CHAPEL STREET
LONDON NW1 **TEL: 402 8137**

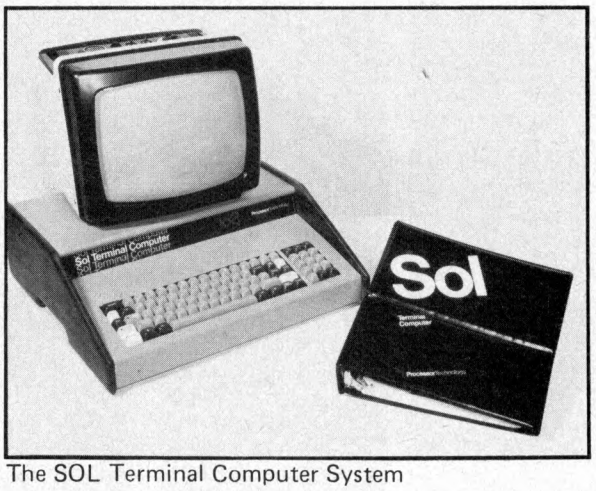
NEXT TO EDGWARE ROAD TUBE STATION MET LINE

'TRITON COMPUTER' IS THE TRADE MARK OF TRANSAM COMPONENTS LTD

* TRANSAM Components Limited is your new personal computing specialist in London.
* TRANSAM opens mid-September in Chapel Street.
* TRANSAM has a full range of Micros and support, plus software.
* TRANSAM brings you 'Memory Bank', a full memory service.
* Catalogue now available. Send 30p and SAE.

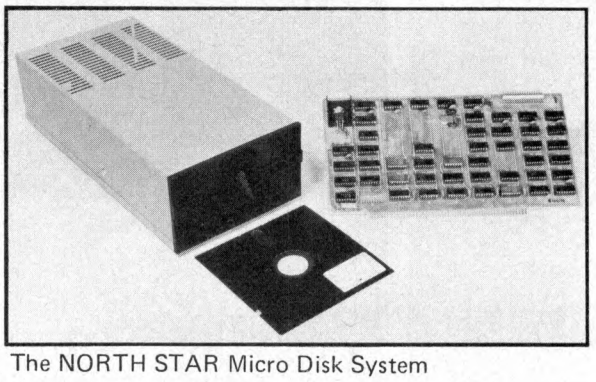
comart

SPECIALISTS IN MICROCOMPUTERS



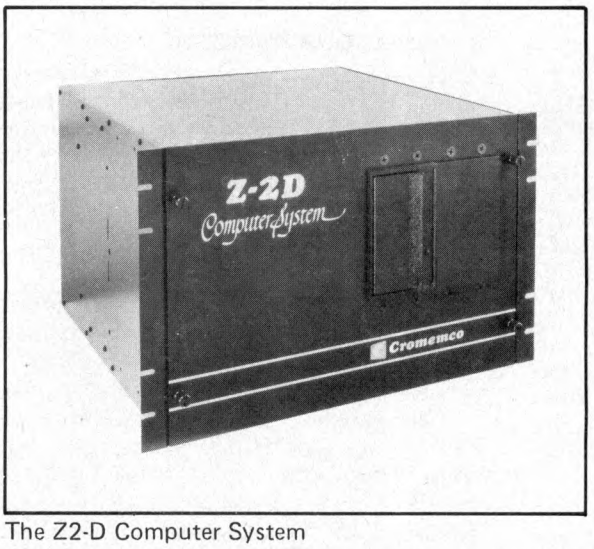
The SOL Terminal Computer System

Comart's range of S100 microcomputer systems features the SOL Terminal Computer System and the Cromenco Z2, Z2-D and System Three Computer Systems.



The NORTH STAR Micro Disk System

S100 sub-systems and modules include 16k and 32k byte memory, PROM programme and erasers, analogue and digital interfaces, serial interfaces and the Micro Disk System with its optional hardware floating point arithmetic board.



The Z2-D Computer System

Software, too is not forgotten!

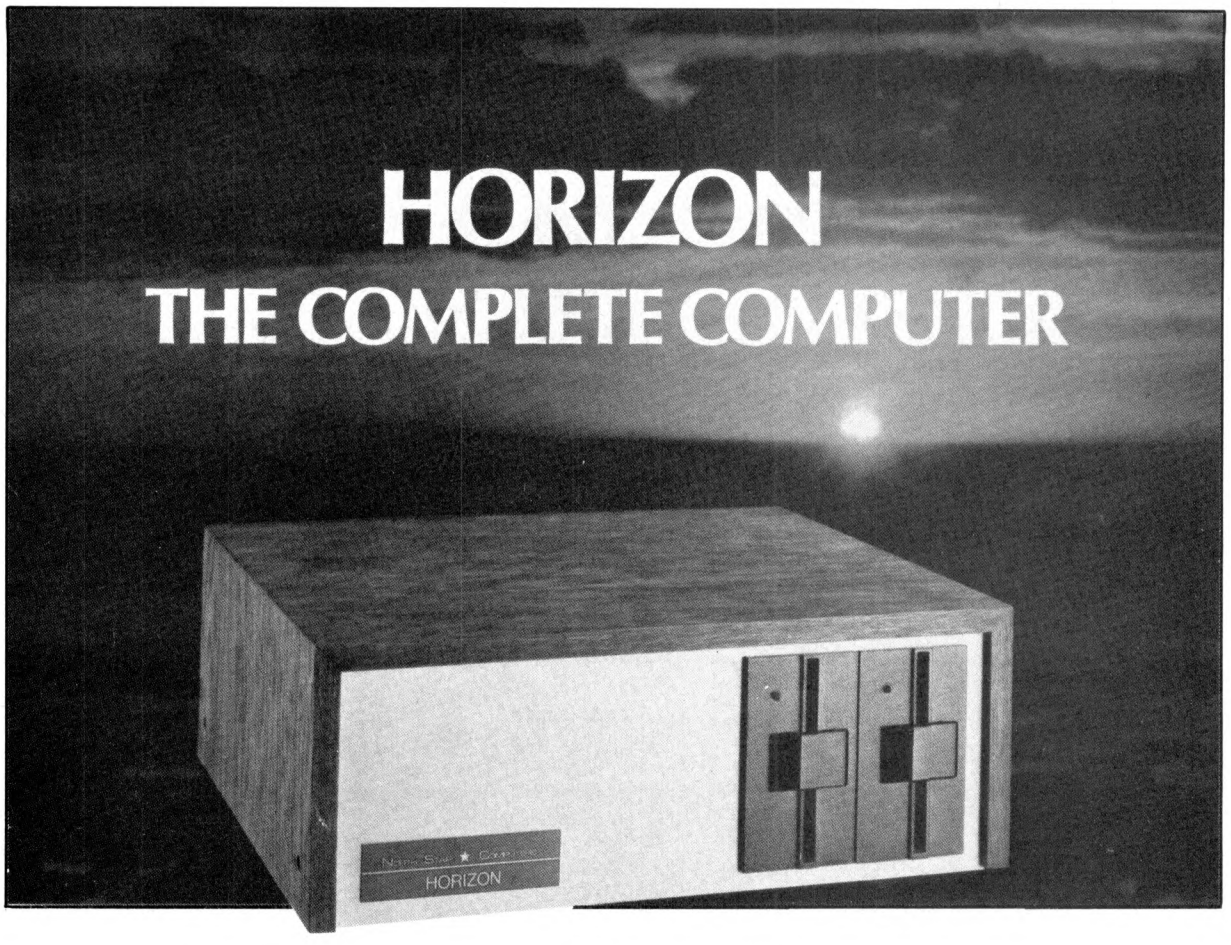
Assembler, BASIC and Extended BASIC are available for all our systems.

In addition FOCAL is a feature of the SOL and Macro Assembler, and Fortran IV options on Z2-D and System Three.

Contact us direct or contact your nearest Comart Dealer.

Computabits Ltd., 41 Vincent Street, Yeovil, Somerset. Tel: (0935) 26522
Computer Workshop (Manchester) Ltd., 29 Hanging Ditch, Manchester. Tel: 971 - 832 2269
NewBear Computing Store, 7 Bone Lane, Newbury, Berks. Tel: (0635) 46898
R.I.I.C. Business Systems Ltd., 110 Leagrave Road, Luton, Beds. Tel: (0582) 605535, Ext. 235.
The Byte Shop, 426/428 Cranbrook Road, Ilford, Essex. Tel: 01 - 554 2177
Xitan Systems, 31 Elphinstone Road, Highcliffe, Dorset. Tel: (4252) 77126

Comart Limited, PO Box 2, St. Neots, Cambs. PE19 4NY. Tel: 0480 215005



Look To The North Star HORIZON Computer.

HORIZON – a complete, high performance microprocessor system with integrated floppy disk memory. HORIZON is attractive, professionally engineered, and ideal for business, educational and personal applications.

To begin programming in extended BASIC, merely add a CRT, teletype or other hard-copy terminal. HORIZON-1 includes a Z80A processor, 16K RAM, minifloppy disk and 12-slot S-100 motherboard with serial terminal interface – all standard equipment.

WHAT ABOUT PERFORMANCE?

The Z80A processor operates at 4MHz – double the power of the 8080. The RAM memory board lets the Z80A execute at full speed. HORIZON can load or save a 10K byte disk program in less than 2 seconds. Each diskette can store 90K bytes.

AND SOFTWARE, TOO

HORIZON includes the North Star Disk Operating System and full extended BASIC on diskette ready at power-on. This BASIC, now in widespread use, has virtually everything desired in a BASIC, including sequential and random disk files, formatted output, a powerful line editor, strings, machine language CALL and more. Optional software (under CP/M) includes – CBASIC compiler/interpreter BASIC, Microsoft Disk Extended BASIC, MAC Macro Assembler, Microsoft COBOL-80 and FORTRAN-80, and more.

EXPAND YOUR HORIZON

Also available – Hardware floating point board (FPB); additional 8K and 16K memory boards. Add a second disk drive and you have HORIZON-2. Economical serial and parallel I/O ports may be installed on the motherboard. Many widely available S-100 bus peripheral boards can be added to HORIZON.

QUALITY AT THE RIGHT PRICE

HORIZON Z80A processor board, RAM, FPB and MICRO DISK SYSTEM can be bought separately for either Z80 or 8080 S-100 bus systems.

HORIZON-1: £1,275 assembled and tested.
HORIZON-2: £1,550 assembled and tested.

8K, 250ns static RAM: £145; 16K, 250ns static RAM: £295; FPB: £215; Z80A board: £185; 2nd serial I/O: £45; parallel I/O: £45. All prices are for assembled and tested units. Prices are exclusive of V.A.T. and carriage, and are subject to change.

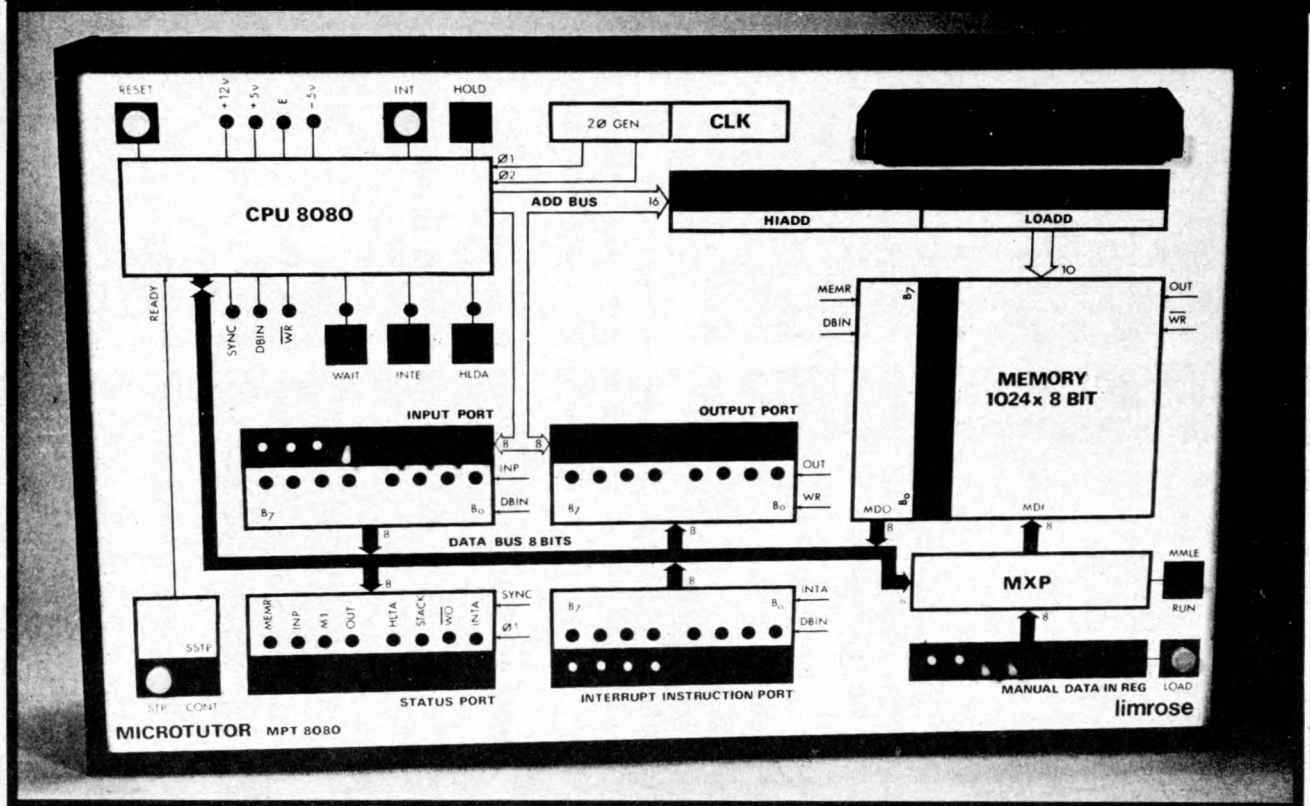
For full details contact:

INTERAM

INTERAM Computer Systems Ltd.
59 Moreton Street
Victoria, London SW1V 2NY
Telephone: 01-834 0261/2733

THE LIMROSE MICROTUTOR 8080

New, Fast and low cost method for learning the "how" of microprocessors



Now.. there is a new, fast, simple and inexpensive MICROTUTOR to give you truly "hands on" experience to master and apply microprocessors.

Limrose's MICROTUTOR MPT 8080 has been chosen by the British Post Office, and many other large companies in U.K. and overseas, to train their engineers in this new and exacting technology.

The MICROTUTOR MPT 8080 comes ready to use. Nothing else to buy or debug. It's front panel has all you need to understand how the industry-standard 8080 microprocessor works. And... you can expand it later to use as a 'Development System'.

It's inexpensive, it's versatile and comprehensive.

It has a full range of expansion modules, such as expansion motherboard, RAM, ROM, Teletype and VDU interfaces, Cassette Interface, EPROM and PROM Programmers, and even an inexpensive EPROM erasure.

And, of course, the MICROTUTOR MPT 8080 is supplied with a comprehensive Instruction Book. By following the Instruction Book, most persons with limited technical knowledge can rapidly learn how microprocessors work.

AND.... IT IS BRITISH - MADE !!

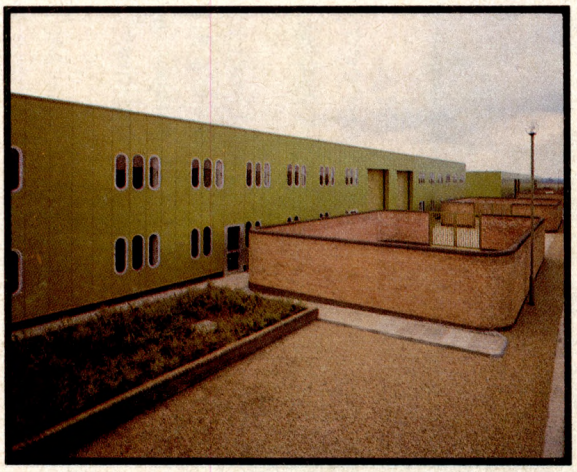
- * 8-bit Microcomputer with 1K RAM
- * 8-bit Input Port
- * 8-bit Output Port
- * 8-bit Status Port
- * Manual or Teletype Operation.
- * Full vectored- Interrupts
- * Buffered Data and Address Buses
- * Advanced "Real-time" monitor
- * Expandable with plug-in cards
- * Comprehensive Instruction Book

limrose electronics limited

241-243 Manchester Road, Northwich, Ches., CW9 7NE. Tel. 0606 41696/7

We're getting bigger to give you a better service

We've opened a Sales and Customer Service Centre in London. And our new factory is fully operational. Southwest Technical Products provide a range of superb computer systems with technical backing second to none.



Systems
To suit all types of user – OEM, process control, data handling, small business systems, and all accounting functions.

Software
Low cost packages for word processing, selective mailing, progress control and invoicing. Our Software Development Unit available to prepare programmes to customer specification.

Training
Inexpensive courses (at Dover Street): BASIC – programming for the businessman; microcomputers in EDUCATION; WORD PROCESSING made easy; SOFTWARE DEVELOPMENT – make your microprocessor work.

Maintenance
Comprehensive national service by Computer Field Maintenance Ltd.

Sales Office: 38 Dover Street, London W1.
Tel: 01-491 7507 Telex: 268913.
Factory: 12 Tresham Road, Orton Southgate, Peterborough.
Tel: 0733-234433 Telex: 32600.

SWTP

the symbol of reliability