DECEMBER 1985 VOL.6, NO.12



The Unofficial OSI Users Journal

P.O. Box 347 Owings Mills, Md. 21117 (301) 363-3268

INSIDE

HEXDOS UTILITIES	- 2
BEGINNER'S CORNER	3
LANGUAGE DEBATE REVISITED	6
SOFTWARE LISTINGS	9
OS-65U DATA FILES & OTHER	17
MYSTERIES	
DB-65E, OPERATING ENVIRONMENT	2Ø
PEEK(65) INDEX FOR 1985	22

Column One

Jingle bells! Jingle bells! Hey! Wait a minute! It's not even Thanksgiving time yet. Maybe we are just old fashioned, but it is hard to get into the spirit when the leaves are still falling.

But never mind. Santa and the elves have been busy putting together just the kind of issue that you can snuggle up by the fire with - providing that you don't take your thinking cap off. This issue is full of thought provoking topics, not to mention the continued listings of available software and the annual index.

Jim McConkey gives us some practical insight into HEXDOS with his DIR programs. Leo Jankowski dives headlong into the art of debugging and testing in his Beginner's Corner. Needless to say, his principles apply to all available languages and any programmer (beginner through Santa).

Now if you really want to think, curl up with Steve Gale's follow-up on the Great Language Debate, or Part I of Rick Trethewey's explanation of OS-65U. Steve Gale is the first to respond to Roy Agee's thought provoking series of articles. Two holly leaves and a snow flake for you Steve! We sure hope that there will be more responses of the same caliber. As for Rick, I can see him now, Ho! Ho! Ho! and a twinkle in his eye. Here's a died in the wool 65-D resident expert who found 65-U, rejoiced and is now sharing its wonders through those 65-D eyes. So you think that you have D and U under your belt. Well, here comes DB-65E. Wazzat? Thanks to Art Hughes, the wizzard of DBI, we delve into the world of operating environments as he begins, in part I, to explain this new yet compatible release. Stav tuned. There is more coming after your yule log has burned down.

÷

OSI has pulled the wraps off their Christmas present - the 712. What we hear about, what you cannot see in their ad, should make this new addition a very viable member of the OSI family. Watch "Manufacturer's Corner" next month.

Elf Earl (Morris) advises that we need more "feedback" on articles and letters, and he is right. So, just get out your WP, and let us hear from you. In that same vein, Earl points out that the response to your Want List of Articles has been very helpful, but needs up-dating and repeating. So, even if your WP is a little hung over, it should still be able to manage a short note telling us again what types of articles you would like to see or what areas you feel need clarification or explanation.

It's time to be making your New Year's resolution list. We have a special request for dealers and business users. Please use a little of your holiday time to write us about your business applications, since we regularly are asked about business installations and the unusual tasks these machines are capable of performing.

Just to show that it is not all take and no give, PEEK (thanks to Paul Chidley) has a present for C4/8P 48k users. Paul'S OSI-CALC is now available through PEEK. It is a full feature spreadsheet using 26 columns by 36 rows interactively on-line. The 30+ copies in circulation are all in BASIC (easy modification and comprehension), and we understand that someone has it running on a ClP. The best part is that it is only \$10 plus P&H for either 5" or 8". Next month, watch for the TOSIE Floppy Paddle Board!

There is nothing like sitting down and writing about Christmas (well, sort of) to get one in the spirit. And now that I am, on behalf of the rest of the PEEK staff (that's Karin and Ginny), have a Merry Christmas filled with computing delights and a productive program filled New Year.



By: Jim McConkey 7304 Centennial Road Rockville, MD 20855

Here are a couple of HEXDOS utilities which may come in handy. The program in Listing 1 is an adaptation of the utility supplied with HEXDOS which prints a directory and asks for a program to run. It is normally made to autorun on boot. This adaptation prints the directory in alphabetical order. It also uses an unusual technique to cut down on string manipulations so that the infamous garbage collector doesn't send the machine to never never land. Instead, on concatenating letters retriev-ed from the directory to form file names, the ASCII values are POKEd into a string de-fined in line \emptyset . This way, a single string can be reused. Note that since HEXDOS allows file names of up to 255 characters, the 60+ characters AS in line 0 will not handle all possible file names. I have never found this to be a problem since no one wants to type such long file names anyway. Line 10 sets up logical file 5 to read the directory file, while lines 20-60 get the file names. Lines 130-300 sort the directory alphabetically.

Listing 2 is a program that will list the directory of an OS65D disk. It is much the same as the previous utility, except the directory read is different. The directory read was adapted from the OS65D READER program presented by Steve Hendrix in PEEK(65) Apr. Line Ø is a 1983 page 2. machine code subroutine and should be entered first using the monitor. The hex dump the monitor. The hex dump published in '83 is duplicated here for your convenience. After entering the subroutine, do a warm start and enter the rest of the program. On

·		
Copyright • 1985 PEEK (65) Inc. All Rights	Reserv	ved
published monthly		1
Editor - Eddie Gieske		
Technical Editor - Brian Harston Circulation & Advertising Mor Karin Q. Gi	eske	
Production Dept A. Fusselbaugh, Ginny N	lavs	
Subscription Rates	Air	Surface
Canada & Mexico (1st class)		\$26
So. & Cen. America	\$38	\$30
Europe	\$38	\$30 *
Other Foreign	\$43	\$30
All subscriptions are for 1 year and are payal US Dollars.	ole in a	dvance in
For back issues, subscriptions, change of a , information, write to:	addres	s or other
PEEK (65) P.O. Box 347 Owings Mills, MD 21117 (301) 363-3268		
Mention of products by trade name in edit advertisements contained herein in no way dorsements of the product or products by t the publisher.	orial m consti his ma	aterial or itutes en- gazine or

0 A*=" 1 PRINTCHR*(3): PRINT"((((C DIRECTORY))))": PRINT 10 RESTORE: DATA0.16.0.24.1.2.0.0: FORI=574T0561.READT: POKEI.T: NEXTI 15 DIM N*(40):N=1 20 P=USR(5)+USR(5): IFP=.THENN=N-2: GOTO130 30 T=USR(5)+2564USR(5): S=2825 40 A=USR(5): IFA().THENPOKES.A: S=5+1: GOTO40 50 IFT(2THEN20 (0 M:(M)) IFT(2THEN20 50 IFT(2THEN20 60 N*(N)=LEFT*(A*,S-2825):N=N+1:GOTO20 130 L=(2*(LOG(N)/LOG(2)))-1 140 L=INT(L/2):IFL(ITHEN300 160 FGR2-ITOL:FORK=J+LTONSTEPL:I=K:T*=N*(I) 200 IFN*(I-L)(T*THEN230 200 IFN*(1-L)(T*THEN230 220 N*(1)=N*(1-L):I=1-L:IFI)LTHEN200 230 N*(1)=T*:NEXTX:NEXTJ:GOTO140 300 FORI=ITON:PRINTN*(I):NEXT 310 PRINT:INPUT*SELECTION*:S*:RVNS*

LISTING 2

```
0 REM (THIS WILL CONTAIN A MACHINE CODE SUBROUTINE)
1 P#=""""P=PEEK(123)+256#PEEK(124)+3
2 BP=PEEK(124)+1-(PEEK(123))128)
   REM
REM 0565D DIRECTORY BY JIM MCCONKEY ADAPTED FROM:
REM 0565D READER BY STEVE HENDRIX, FROM APR 1983 PEEK(65), PP2-5
ŝ
6 REM
9 PRINTCHR+(3): PRINT" OS65D DIRECTORY": PRINT"
   REM
                                                                                    -----
10 PRINT: PRINT
15 NL=LEN(FL*)
15 NL=LEN(FL)
20 FL%-LEFT%(FL%*" ",6)
30 POKE240,6:POKE241,11
40 PRINT:PRINT"INSERT 0565D DISK"
50 PRINT"AND PRESS RETURN"
60 IFUSR(0)()13THEN60
60 FIGSR(U)())JJHEROU
65 PRINT: PRINT
70 T=USR(-7)12,1,BP#256:IFT())ITHENPRINT"ERROR IN DIRECTORY":END
80 T=USR(-7)12,2,(BP+1)#256:IFT())ITHENPRINT"ERROR IN DIRECTORY":END
90 FORI-BP#256T0I+511STEP8
130 PRINT10#INT(LT/16)+LT-10#INT(LT/16)
140 NEXT
```

LISTING 3

0 A#=

0 A4=" 1 PRINTCHR*(3):PRINT"(<< DISK DIFFERENCE >>>":PRINT:PRINT 2 EL=PEEK(133):EM=PEEK(134):TL=EL:TM=EH=6:POKE134,TH 5 PRINT"Insert disk A & (CR)"::POKE240,0:POKE241.253:X=USR(-7):PRINT:PRINT 10 RESTORE:DATA1,2.0,0:PORI=578541:READT:POKE21.T:NEXTI 11 POKE574,TL:POKE575,EM:POKE576,TL:POKE577,TH 15 DIMN*(40,1),T(40,1):NA=0:NB=0 20 P=USR(5)+USR(5):IFPA:THENPOKE5,A:S=S+1:GOT060 70 IFT(2THEN20 00 RINT'INSTANTS B & (CR)";:POKE240,0:POKE241,253:X=USR(-7):PRINT:PRINT 110 RESTORE : DATA1, 2, 0, 0: FORI=578T0581: READT: POKEI, T: NEXT 111 POKES74, TL: POKES75, EH: POKES76, TL: POKES77, TH 120 P=USR(5)+USR(5): IFP=. THEN200 140 T=USR(5)+256#USR(5): S=2825 160 A=USR(5): IFA(). THENPOKES, A: 5=5+1: GOTO160 170 IFT(2THEN120 180 N9(NB,1)=LETT*(A*,S=2825):T(NB,1)=T:NB=NB+1:GOT0120 200 NA=NA-2:NB=NB-2 210 FORI=OTONN:T(1,0)=T(1+1,0)-T(1,0):NEXT 220 FORI=OTONB:T(1,1)=T(1+1,1)-T(1,1):NEXT 300 PRINT"ON disk A but not on B:":PRINT 310 FORI=OTONN:F=0:FORJ=OTONB 220 IFN*(I,0)=N*(J,1)ANDT(I,0)=T(J,1)THENF*-1:J=NB+1
330 NEXTJ:IFNOTFHENPRINTT(I,0);" ":N*(I,0)
340 NEXTI:PRINT Continued

2

400 PRINT"On disk B but not on A:":PRINT 410 FORI=0TONB:F=0:FORJ=0TONA 420 IFN(I,1)=%(J,0)ANDT(I,1)=T(J,0)THENF=-1:J=NA+1 430 NEXTJ:IFNOTFTHENPRINTT(I,1);" ":N*(I,1) 440 NEXTJ:PRINT 440 NEXTJ:PRINT 450 POKE 134,EH:POKE133,EL 500 PRINT:PRINT"DIFF COMPLETE":CLEAR 1000 REM DISK DIFFERENCE by Jim McConkey 1010 REM inspired by a CPU,Inc CP/M utility 1020 REM This utility is hereby released to the public domain.

listing, line Ø will look like
a REM followed by all sorts of
strange stuff.

Listing 3 is a disk compare utility inspired by a CPU, Inc. utility for CP/M. It compares the names and sizes of the files on two disks and lists the differences, if any. This is especially useful for checking backups. (What, you don't backup? Shame on you!) The file name reading is the same as for the first utility,

\star

BEGINNER'S CORNER

By: L. Z. Jankowski Otaio Rd l, Timaru New Zealand

DEBUGGING & TESTING OF PROGRAMS

Part l

Debugging can be the most painful part of programming. Much of the mental anguish is caused by attempts to debug a program that was written without a prior plan. The two tasks of solving a problem and writing code must be kept separate. Mix the two together and debugging becomes a nightmare.

HAVE A PLAN

If a program was written from a plan, and the program algorithm was formed away from the keyboard, then the Golden Rule of debugging can be usefully applied. When the Bug strikes ask the question: "What is this segment of code supposed to do?"

Ahal Look at the plan. If the plan is wrong, then change it, then change the program. Separating the two tasks and then completing them in the right order saves time!

Try this program:

10 Y=1 : A\$="Whodunnit" 20 PRINT Y A\$

No output! The syntax looks OK. What was the plan? It was, output the value of "Y" and then the value of "A\$". Actual output was nothing at but it is done twice.

Now I have an appeal. Does anyone know the current status of the HEXDOS user's library? I have been trying to get a copy of the public domain Tiny Compiler for HEXDOS, but my letters to the user's library go unanswered. If the library still exists, please let us know your whereabouts. In any case, if anyone has a copy of the Tiny Compiler, I would like to hear from you. Thanks!

×

all. Why? Well, BASIC ignores blanks and so the program printed the value of "YA\$": null. Correct the program by inserting a ";" or a "," between "Y" and "A\$".

DESK-CHECK

A most annoying bug is when a variable persists in holding the wrong value. For example, to function correctly a program requires that "Y" must equal "2". But "Y" stubbornly persists in holding some other value. A great deal of debugging is done and now "Y=3". Aaargh!

First question, is "Y" a constant or a variable? If it's a constant, it should be declared with all the other constants at the start of the program. If not, where does "Y" take its first non-zero value? If a variable has a definite starting point, e.g., "Y=2", then its value could be observed each time it was used by the program - a "deskcheck". A solid desk-check is always a worthwhile exercise if the bug is not found almost immediately. (Yup, it involves that simple word-processor pencil and paper!)

A desk-check can be made in two ways; either forwards, from the beginning of the program or block of code, or alternatively, it is possible to work backwards from a point in the program and try to reason out why that point was reached! In either case, the programmer should be clear about what the code should be doing and compare that with what the code actually does. What causes program output to be different from the programmer's expectation?

GLOBAL & LOCAL VARIABLES

Let's get back to that variable that should be equal to 2, but persists in equaling 3. If the program is of the long spaghetti variety and the variable is global (used all over the program) then a deskcheck of the program becomes a labyrinthian task. On the other hand, if the program is written in blocks, each block with only one entry and one exit, then there are fewer paths to trace between pieces of code and the bug is considerably easier to find.

The bug search becomes positively easy if the variable is local. A local variable or constant is one that is used in only one part of a program (block), and in no other. An additional advantage is that if an error does occur within a block, it will be prevented from spreading to code elsewhere.

Global variables are best limited to counting in FOR ... NEXT loops, in Input/Output routines and for other short and simple tasks.

SPEED

Extensive use of local variables could possibly affect program speed. A way round the problem is to first check if the program is running detrimentally slow ("trace" the program). If that is the case, discover the slow code and declare its variables before any others. As a last resort, when the program finally works, duplicate the use of variable names across several program blocks, thereby reducing the number of variables in the program and so speeding it up.

Program speed can often be gained with "clever" programming. Such code tends to be complex and it becomes inpenetrable with the passage of time. Keep it simple, keep it readable.

PROBLEM SOLVING

See Beginner's Corner April '85 issue.

Some debugging sessions end in frustration - the bug is intractable and no solution is forthcoming. At the very least though, a theory should have been tested and some information gained. OK, that dangerous stage has been reached where you're thinking of giving up programming. The first thing to do is to have a rest. Next, review the problem solving techniques being employed. Is the problem properly understood? Has this problem been struck before, or one like it, and how was it solved? Does sufficient information exist about the problem?

If analysis of the code does not reveal the bug, then more information is required. Striking a balance between analysis and information gathering is part of the programmer's art.

SPIES

If the value of a variable is in question, or if just more information is required about what the program is doing, then use a "spy". This is simply an extra line inserted at an appropriate point in the program, thus:

100 PRINT Y CHR\$(13);: REM Carriage Return but no Line-feed

The "CHR\$(13);" stops the screen from scrolling.

Another way to examine the

value of "Y" would be to stop the program with a CTRL-C (hold the "CTRL" key down and presss "C"), and then "spy" from Immediate Mode with a "PRINT Y". Follow this with "CONT" to continue the program.

DUMP

More than one variable value can be printed out of course. A dump of all variable values provides a complete description of the program at a particular point. Unfortunately, because the description is total, it tends to be very lengthy.

If you are running "HOOKS" (PEEK(65) Dec '83) then the command "VIEW" will dump the values of all variables, but not arrays. Dump array values from Immediate Mode with a FOR...NEXT loop, e.g.:

FOR C=1 TO 100: PRINT N\$(C) " ";: NEXT

A printout of the program is, of course, extremely useful. A dump to the printer of selected program output is also invaluable, as well as a printout of information provided by "spies". STOP THE PROGRAM

Using CTRL-C to stop a program is something of a hit and miss affair. It is at its most useful if the program wanders off and becomes stuck in an infinite loop, e.g.:

10 PRINT "Wombat" :GOTO 10

Complete precision is achieved with the BASIC word "STOP". The program halts precisely where required. For example, after line 90:

90 a line of BASIC

100 STOP

"STOP" can be made conditional by making it part of an "IF...THEN" statement. Variable values can now be examined in Immediate Mode by using "PRINT Y" etc. It is worth remembering that "VIEW" can also be used as part of a BASIC program, e.g.:

110 VIEW

TRAP

It is possible to force errors into a program. For example, this line is part of a program:

100 N(V,Y) = N(V,Y) / D

UNLOCK YOUR SYSTEM WITH BETA/65

MICROGRAM SYSTEMS IS NOW PROVIDING BETA/65 OBJECT CODE TO OSI USERS. BETA/65 CONVERTS YOUR CHALLENGER INTO A CUSTOM APPLICA-TIONS FACTORY. OVER 100 COMMANDS AND FUNCTIONS INCLUDING FULLY-DEVELOPED STRINGS.

A POWERFUL SYSTEM WITH NATURAL AND EASY NOTATION.

Speci	ify Y	our	System
-------	-------	-----	--------

- \Box CIP \Box C4P \Box C8P \Box Other _____
- \square 0S65D v. \square 3.1 \square 3.2 \square 3.3
- □ 5" Disk □ 8" Disk (40K or 48K Memory Only).
- □ Please find enclosed \$15 to cover mailing and reproduction costs. **MICROGRAM** will supply object disk plus BETA/65 system programming manual (120 pp.). Disk includes run-time package with editor and tutorial plus sample programs.
- \Box Please send more information.

Specify Return	Address
Name:	
Address:	
City/State:	
Zip/Telephone:	
The key to full-use	A
from your OSI Computer.	d
MICROGRAM	SYSTEMS

P.O. Box 252, La Honda, Calif. 94020 (415) 747-0811



The program is tested by forcing D to equal zero when the program reaches line 100. BASIC will then respond with the error message "/0 ERROR" -"division by zero error". The error has been "trapped". It is a simple matter to now debug the program by inserting the line:

90 IF D=0 THEN 110

The "trap" technique can be used to simply test code or to gather information on how the program behaves. То use "trap", start by making а prediction on how the program will respond to the introduction of the deliberate error condition. Next, amend the program and run it. Compare the outcome with the prediction. Often there is an indi-cation of what the program will NOT do as well as what it does do. The "trap" has another advantage. It is an excellent way of discovering how a programming language works, what it can and cannot do.

Next month, part two.

THE GREAT LANGUAGE DEBATE REVISITED

By: Steven M. Gale Box 481 Pittsburgh, PA 15213

INTRODUCTION

As a student of Computer Science and a Pascal Tutor at Carnegie-Mellon University and a Data Processing Manager for a small business, I read with interest Mr. Agee's, "Great Language Debate" (PEEK(65) September, 1985). Many people, especially people who learned about computers with BASIC or FORTRAN as the only languages available, don't understand why "structured programming" is good, what it entails, and why it's here to stay. After I make a crude attempt to explain these things, I'll show how we've added some structure to our 65U applications.

WHAT STRUCTURE IS ALL ABOUT

It has been found that the most expensive part of developing software is changing it to meet the changing requirements, and maintaining it for years after the original program was written. "Structured programming" is an attempt to make changes and maintenance easy. Some of the following techniques are used: logical algorithmic break-down, logical data break-down, data protection, and formatting for easy human consumption.

By logical algorithmic breakdown, I mean general problem solving by stepwise refinement. That is, you solve a problem by breaking it into separate problems and solving them similarly. To support this, a language should have some sort of sub-program structure (like subroutines). BASIC does.

Logical data break-down is similar to its algorithmic brother. Every program uses data, and this data should be organized into logical structures. This is usually done with "data types". (A "type" is the range of possible values a variable can have.) BASIC has some troubles here. It can group things of the same simple type together (arrays). It cannot group (arrays). It cannot group things of different types together (records). BASIC does not support the dynamic allocation of variables, so many data structures (like trees) are hard to implement without a lot of wasted space. Also, BASIC doesn't support enumerated types, i.e., TYPE MONTH = (Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec).

A common bug is when one subroutine accidentally modifies data it had no business modifying. This problem can be minimized by variable "scoping" (data protection). This means that a routine can only modify its own variables, or those explicitly "passed" from another routine (the var-(the iables passed are called "par-ameters"). This feature helps both data and algorithmic break-down. Break-down is the separation of things into parts, and data protection makes it possible to isolate (separate) data from data and code (that handles some data) from code (that handles other data). BASIC has no such fea-ture. The best you can do is use variable naming conventions, but this re readability, and is restricts error prone.

Formatting for easy human consumption involves many things. Identifiers (names of variables, subroutines, constants, types, etc.) should be meaningful. "GOSUB SORT" is better than "GOSUB 1024". The control structures: sequential execution, IF-ELSE, and WHILE neaten code by making it logical. It has been proven that you can write any program using these instead of GOTO. Since there are no GOTOs, no line numbers are needed. have a hard time thinking Т of things less human consumable than line numbers. Proper indenting and commenting are also necessary. BASIC is weak on all of these counts. It even discourages indenting and commenting for speed reasons. It's possible to try to simulate the control structures, but this is still pretty messy.

Pascal-like languages are here to stay for technological and economical reasons. It is more economical to develop software in a Pascal-like language because updating and maintaining is easier. If you can do it for less, why not? Pascal was less popular than BASIC in the early micro community because BASIC can run on a smaller machine. Now that technology has made more powerful ma-chines cheap, it's hard to find a new business computer that doesn't have a Pascal compiler available. It is true that there are a lot of machines running BASIC out there, but the numbers will become proportionately less and less as old machines are replaced by new ones, and new is developed software in The European scienti-Pascal. fic and business community has been using Pascal-like lan-guages for over 15 years. Limited application outside the classroom? They don't seem to think so.

UNUSABLE SOFTWARE AND A SOLUTION

We bought our OSI with the intention of writing our own

DISK DRIVE RECONDITIONING WINCHESTER DRIVES FLAT RATE CLEAN ROOM SERVICE. (parts & labor included) 23meg \$550.00 SA4008 Shugart Shugart SA1004 10meg \$390.00 Seagate ST412 10meg \$295.00 FLOPPY DRIVE FLAT RATES 8" Single Sided Shugart \$190.00 8" Double Sided Shugart \$250.00 8" Single Sided Siemens D&E Series \$150.00 8" Double Sided Siemens P Series \$170.00 Write or call for detailed brochure 90 Day warranty on Floppy & Large Winch. 1 Yr. Warranty on 5" & 8" Winchesters. Phone: (417) 485-2501 FESSENDEN COMPUTERS 116 N. 3RD STREET OZARK, MO 65721

SUPER MARD DISK Subsystems turns any floppy based computer into hard disk based, instantly. PLUGS INTO ANY OSI TYPE BUS ONE RIBBON CABLE CONNECTS TO DRIVE COMPUTERLY SELE CONTAINED

- COMPLETELY SELF CONTAINED
- 32 BIT ERROR DETECTION AND CORRECTION
- HAS REAL TIME CLOCK
 *CALENDAR W/BATTERY ON SCSI ADAPTER BOARD
- CAN BOOT DIRECTLY FROM OSI 505/510 CPUS OR DENVER BOARDS W/SCSI PROM
- IDEAL BACK-UP FOR ALL OSI HARD DISK COMPUTERS



FROM \$1,999.00

The SPACE-COM SUPER SUBSYSTEM Uses 51/4'' Industry Standard Hard Disk drives interfaced to the OSI bus by the DS-1 SCSI Host Adapter Board at the computer end and the state of the art OMTI 5000 series Intelligent Disk/Tape Controllers at the disk end. The Denver DS-1 Board not only provides the Bus Translation, but gives Real Time of Day, Day/Week, AM/PM, and Day/Mo. With on board battery, Date and Time are maintained w/o power.

Single 20 M/B drive (15.7 formatted) single case\$1,999.00 Single 26 M/B drive (21 formatted) single case\$2,199.00 Dual 20 M/B drives (31.4 formatted) dual case\$2,999.00 Dual 26 M/B drives (42 formatted) dual case\$3,299.00 Super Fast 85 M/B drive (70 formatted) single case\$3,999.00 Dual 85 M/B drives (140 formatted) dual case\$6,699.00



New PB-1 Piggy Back Serial Printer Board for DB-1 and OSI Bus.

- Can be plugged onto aux port of each DBI that wants availability of a local printer and each board's output can be routed to a printer or to any other PB-1 in system for a shared printer.
- Takes load off main bus for printing.
- Normal address FBFO, but jumperable to CFOO for standard port 8 address.
- If used w/female molex, plugs directly onto OSI bus.

Only \$ 69.95 ea.

SPACE-COM International 14661A Myford Road, Tustin, CA 92680 (714) 731-6502

software, since there were no packages on the market (for any machine) that would suit We bought about our needs. \$2000 of commercial software anyway to try to limit the amount of development needed. All of the software we bought for the OSI (except RESEQ) was so poorly written that it was unusable, so it's a good thing we weren't counting on outside help. The main problem with developing software was the fact that BASIC was our only choice if we wanted timesharing. The solution was to write a preprocessor for BASIC, which we called GASP (Gale's BI-RITE Structured Programming) and a set of text file utilities so text could be manipulated easily. Most in-house editor (JEDIT) and then compiled by GASP into compacted BASIC for fast execution. The main problem with this is the fact that the compiler is slow (compilers that are written in interpret-ed BASIC can't be expected to run fast).

Except for the lack of line numbers and the use of userdefined labels, any BASIC statement written using standard 65U BASIC syntax can be used in GASP. In addition, GASP recognizes structured

REM COMPARE UTILITY REM By Steven N. Gale REM Created: 8/12/85 REM Update: 8/12/85 5:00pm

REM This program will compare 2 text files and show the lines REM that are different. One file is considered to be the REM "control", and one the "test". The computer goes down line REM by line in each file until they don't match. If there is a REM number at the beginning of both lines, then the pointer to the REM file with the smaller line number is moved to the next line. REM If there are no line numbers, the "test" file pointer is moved REM to the next line. It is assumed that both files end with a REM MIN ONLY containing "GK".

REN	*** VARIAB	LE TABLE ***
REM	CCOUNT	control line count
REM	CFILE\$	control file name
REM	CLINES	control file: last line read
REN	CPASS\$	control file pass
REN	DV	output device number
REN	TCOUNT	test line count
REM	TFILE\$	test file name
REM	TLINEŞ	test file: last line read
REN	TPASS\$	test file pass

REM The main routine calls INIT, then loops through both files REM until one get an end-of-file marker. Then it outputs the number REK of lines and bytes in each file.

GOSUB AINIT

END

GOUDE TANNE (CLINES (> "OK" AND TLINES (> "OK") GDO GIP (CLINES = TLINES) GOSUE SMOVE-BOTH-FILES-1-LINE GELSE

GOSUB &MOVE-DOWN-1-FILE-1-LINE &END

PRINT #DV PRINT #DV, "Compare complete." PRINT #DV, "Control file:" CCOUNT "lines, " INDEX(1) "bytes." PRINT #DV, "Test file:" TCOUNT "lines, " INDEX(2) "bytes." CLOSE

programming keywords differentiated from BASIC statements by using an ampersand ("&") as the first character. It supports the IF, IF-ELSE, and WHILE constructs. All three can operate on a single statement or on a block of statements (grouped together with DO-END). Since it's compiled, it's okay to have lots of comments and blank lines. Putting one statement on a line also helps readability (the compiler puts as many statements on a line in the object file as it can to improve speed). It is now possible to achieve a degree of readability not possible in the best BASIC programs.

GASP doesn't add any scoping or data break-down facilities. To try to prevent as many bugs as possible, we have some (admittedly loosely enforced) naming conventions. Data is grouped into (imaginary) records. The fields of these records have the same names from program to program. There is a table of variables at the beginning of every program. Before using a new variable, you add it to the table. This helps eliminate variable naming conflicts. We try to pick meaningful variable names, but GASP doesn't process them so they must be compatible with BASIC. No keywords can be sub-strings of variable names. Changing an "O" (oh) to a "Ø" (zero) often helps. These tricks help, but the deficiencies of BASIC are not eliminated.

AN EXAMPLE

I have included the source and object code of a little program that compares files Listings 1 and 2). Rem (see Remember that WHILE, IF, and IF-ELSE operate on single instruc-tions. A DO-END pair groups many instructions together so instruc- : The they are treated as one. label MAIN is used for clarity. Since there are no calls to MAIN, the label could be removed. Note that all of BASIC's commands are still valid (I often use the BASIC IF).

It's fun to compare the source code with the object that the compiler produces. The object code is not as efficient as it could be, but it's pretty good. I thought about adding variable scoping, long variable names, and an optimizer, but I don't really have the time...

Continued on page 17.

REM This routine is called when the files are in synch. It moves REM both files to the next line. It doesn't read past an end-of-file REM marker.

REM This is called when the files are not in synch. It outputs REM line count and the last line read from both files. Then it: REM If the "test" line has a greater line number, then the "control" REM file is read. Otherwise, (is: "control" line number is bigger or REM there are no line numbers) the "test" file is read.

EMOVE-DOWN-1-FILE-1-LINE
PRINT #DV, "C" CCOUNT; CLINE\$
PRINT #DV, "C" CCOUNT; CLINE\$
PRINT #DV, "T" TCOUNT; TLINE\$
EIF0 (VAL(TLINE\$) VAL(CLINE\$))
ED0
INPUT \$1, [72,"A"] CLINE\$
CCOUNT + 1
EEND
ELSE
ED0
INPUT \$2, [72,"A"] TLINE\$
TCOUNT + 1
EEND
RETURN
RETURN

Listings continued on page 17.

SOFTWARE LISTING 1985

EXPLANATION OF LISTING CODES

BASIC Version No./ Minimum computer/ 1=SB, SBII, C1P, C2/4P 4 = C4P8=C8P O=C2/30EM D=C2/3-D 2=C200,C3A/B 3=C3ØØ Minimum Storage required/ C=Cassette 5=5 1/4" MF 8=8"FD 7=CD-7 2=CD-20/23/28/30/36/74/ digit following indicates number of devices required. Systems Supported/

S=Single User M=Multi-User H=Hard Disk R=Record Locking record lock assumes multiuser. Two may be specified.

Software Support by/ D=Dealer P=Phone M=Modem N=None O=Other

Sold by/ A=Author D=Dealer M=Mail order O=Other

Copies in Circulation/ No. multiplied by 10, i.e. l=Less than 11 l1=100-110

Price/ Dollars only, no cents, tax, shipping, etc.

Neither PEEK(65), Inc. nor ISOTRON, Inc. make any representation or warranties with respect to the contents of this software list and specifically disclaim any implied warranties of merchantability of fitness for any particular purpose. We have used the vendors' provided information wherever possible, though, we reserve the right to make alterations in the list without the obligation of either party to notify any person or persons of such revisions or changes. Many of the items listed have never been reviewed or utilized by ISOTRON, INC.

That said, we hope that this listing will prove useful to our readers, dispel the belief that "there isn't any software out there." APPLICATIONS BASIC 2.2/3/2/?/D/D/50/ \$395

Author: STACEY TAYLOR P.O. BOX 2718 NEWPORT BEACH, CA 92663

Seller: SOFT GOLD, INC. SAME

BUSINESS BASIC INTERPRETER, COMPATIBLE WITH MAI/BASIC FOUR LEVEL II. KEYED, RANDOM AND SERIAL FILE ACCESS BUILT IN.

DBASE II 2.2/3/82/S/O/A/?/ \$?

Author: ASHTON-TATE 10150 W. JEFFERSON BLVD. CULVER CITY, CA 90230

Seller: SAME

DBASE II IS A DATABASE MANAGE-MENT TOOL THAT ALLOWS EASY MANIPULATION OF SMALL AND MEDIUM SIZED DATABASES USING ENGLISH LIKE COMMANDS.

FRIDAY! 2.2/3/82/S/O/A/?/ \$?

Author: ASHTON-TATE 10150 W. JEFFERSON BLVD. CULVER CITY, CA 90230

Seller: SAME

FRIDAY! IS THE MICROCOMPUTER FILE HANDLING SYSTEM TO ENTER DATA IN A FILE, FIND INFORMA-TION IN A FILE, AND PRINT REPORTS USING SELECTED INFORMATION FOUND IN YOUR FILES.

TAX II ?/8/81/S/O/A/2/ \$50

Author: ROBERT BALDASSANO 4045 ASHBROOK CIRCLE SAN JOSE, CA 95124

Seller: RSB ENTERPRISES · SAME

TAX II COMPUTES 1040 SCHEDULES A,B,C,G, & FORM 6251. INCLUDES UTILITIES, TAX TABLES, & TAX FORM FILES. PRINTS ANNOTATED LISTING OF RESULTS. PRINT ON FORMS OPTIONAL. \$10. **PHONE COMMAND SYSTEM** ?/8/82/S/D/D/1/ \$1000

Author: DAMON CURRY 2 EAGLE DRIVE DAYTON, OH 45431

Seller: SOFTOUCH, INC. SAME

MULTI-USER, TERMINAL PLUS PHONE ACCESS TO COMPUTERIZED MESSAGE RELAY SYSTEM. BUFFERS INCOMING MESSAGES, RELAYS TO MANY PRINTERS (SERIAL). USES: HOSPITAL ADMISSION/DISCHARGE TRANSFER.

PROBIT ?/8/81/S/0/A/1/ \$30

Author: ROBERT T. KINTZ 104 COUNCIL ROCK AVE. ROCHESTER, NY 14610

Seller: SAME

PROBIT IS A PACKAGE TO COMPUTE PROBIT ANALYSIS, A STATISTICAL TECHNIQUE USED TO COMPUTE THRESHOLD IN INSECTICIDAL PHARMACOLOGICAL AND TAXI-COLOGICAL ASSAYS.

DOS/65 3.3/4/52/S/D/D/20/ \$125

Author: RICHARD A. LEARY 450 FOREST AVE. #D312 NORRISTOWN, PA 19401

Seller: MICRO SYSTEMS TECHNOLOGY SAME

DOS/65 IS A CP/M LIKE OPERAT-ING SYSTEM FOR OSI AND OTHER 6502 SYSTEMS. INCLUDED: ASSEMBLER, EDITOR, BASIC, DE-BUGGER AND OTHER UTILITIES. DOCUMENTATION IS APPROXIMATELY 200 PAGES.

◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇
OS65-D*UTILITY*SERIAL
◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇ ◇

BAP ?/8/81/S/O/A/3/ \$200

Author: J.M. TIRINO 16 MAPLE AVE. WEST NYACK, NY 10994



Seller: NORTH EAST FINANCIAL SAME

EAP MULTI PRECISION ARITH: ADD, SUBT, MULT, DIV; 13 DIGIT INTO 23 DIGIT. 'OUT' REPLACES 'NULL' WHEN ENABLED. REQUIRES 2K OF RAM. ROUNDS UP TO 9 PLACES AFTER DECIMAL POINT. 39 PAGE MANUAL WITH SAMPLE CODE.

DISK LIBRARIAN 3.3/4/51/S/D/D/1/ \$25

Author: L. Z. JANKOWSKI OTAIO RD 1, TIMARU NEW ZEALAND

Seller: THOMAS TECH SERVICES RD 1, BOX 135 LINDEN, PA 17744

DISK LIBRARIAN, POWERFUL, EASY TO USE. PRODUCES A SORTED LIST OF 1000 FILENAMES ACROSS 50 DISKS, OR MORE. LOAD, SAVE, EDIT, SORT & PRINT LIST. FIND UTILITY. SEARCH ON ANY NAME TO IDENTIFY THE DISK. MANUAL ON DISK. 'RJTEXT' PRGM PRINTS WP V1.2 FILES, RIGHT JUSTIFIED. 'NUZERO', ZERO-FILE UTILITY.

TEC 65/SMART TERMINAL 3.3/4/51/S/D/D/2/ \$50

Author: DAVID P. MARSH 2920 WEST MOANA LANE RENO, NV 89509

Seller: 6502 PROGRAM EXCHANGE SAME

COMBINES A FAST LINE ORIENTED EDITING PROGRAM BASED ON DECTECO WITH A TERMINAL PACKAGE THAT ALLOW UP AND DOWN LOADING FROM A REMOTE, FULL OR HALF DUPLEX, USER SELECTABLE PROMPT.

A/P W/CHECK WRITING ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PERFORMS A/P AND CASH REQUIRE-MENTS MANAGEMENT. REPORTS LIST UNPAID INVOICES FOR EACH VENDOR. AUTOMATICALLY PRINTS VENDOR CHECKS WITH DETAILED CHECK VOUCHERS. REQUIRES G/L TO BE ON-LINE.

APPAREL MFG. SYSTEM ?/3/2/MR/O/A/1/ \$?

Author: B.W. CLEMENT 121 PRESCOTT E. HWY. PRESCOTT, AZ 86301

Seller: SAME

APPAREL MFG. SYSTEM WITH PCWRK PR. PROD. CONTROL A/P, A/R, G/L, AND OCR BUNDLE TICKETS, CUTS, AND STYLES. SHIP SCHEDULE AND LOG. MEMOS AUTO VALUED AND POSTED TO A/R AND G/L. 12 WK. AVG. MAINTAINED PER. EMP.

A/R WITH INVOICING ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

SYSTEM PERFORMS RECEIVABLE ACCOUNTING & CREDIT ANALYSIS. CAN SELECT BALANCE FORWARD OR OPEN ITEM ACCOUNT ON A CUST-OMER BY CUSTOMER BASIS. SYSTEM ALSO PRODUCES MAIL LABELS AND ROLADEX CARDS.

CMOES MAIL ORDER 1.2/3/2/MH/D/D/1/ \$10K

Author: DAVID R. HENDRICKS 825 OSCEOLA DRIVE ROCKLEDGE, FL 32955

Seller: ASSOCIATED INFO. SYS. INC. SAME

CITRUS MAIL ORDER ENTRY SYS-TEM. MULTI-USER, NETWORK FOR LARGE DATA FILES TO 1,000,000 NAMES. EXPANDED VERSION OF 200 SERIES C.M.O.E.S. SOFT-WARE. DBMS COMPATIBLE FILES. TAPE BACKUP REQUIRED.

CONSUMER CREDIT AGENCY 1.4/3/2/MH/D/D/1/ \$6000

Author: JIM ISABELLA 3004 CENTER ROAD POLAND, OH 44514

Seller: COMPUTER APPLICATIONS SAME

STORES CLIENT AND VENDOR INFORMATION. INPUT CLIENT PAYMENTS, PRODUCES VENDOR CHECKS AND REGISTER. PRINTS INITIAL/DROP LETTERS AND OVER TEN OTHER REPORTS.

CONTRACTOR III ?/3/2/S/0/0/40/ \$12K

Author: HOLLY IRENE SMITH 1610 S. MAIN/SUITE G BOUNTIFUL, UT 84010

Seller: A-SYSTEMS CORP. SAME

CONSTRUCTION MANAGEMENT SOFT-WARE. JOB COSTING, PAYROLL, GENERAL LEDGER, ACCOUNTS PAYABLE, ACCOUNTS RECEIVABLE, ESTIMATING. MATURE. INSTALLED 7 YEARS. HUNDREDS OF USERS NATIONWIDE.

COUGER ?/3/2/MR/D/D/70/ \$695

Author: GLENDA TRUMBO 2609 KOOTENAI BOISE, ID 83705

Seller: COUGER MOUNTAIN SOFTWARE SAME

COUGER MOUNTAIN SOFTWARE PRO-GRAMS ARE A MODULAR ACCOUNTING SYSTEM, FEATURING MULTI-USER, MULTI-COMPANY CAPABILITIES. THE MODULES CAN BE USED AS STAND ALONES, SUBSETS, OR TOTALLY INTEGRATED.

EBS ACCOUNTING SOFTWARE 1.4/3/2/S/D/D/3/ \$?

Author: ELECTRONIC BUSINESS SYSTEMS 307 MICHIGAN N.E. GRAND RAPIDS, MI 49503

Seller: ISOTRON, INC. 140 SHERMAN ST. FAIRFIELD, CT 06430

ACCOUNTING MODULES WRITTEN IN KEYBASIC, INCLUDING: A/R, A/P, G/L, PAYROLL, AND INVENTORY. ABLE TO INTERACT WITH KEYBASE T (DBMS) PROGRAMS. MARKETED BY ISOTRON AND SUPPORTED BY EBS.

EMS-PAYROLL ?/3/2/MR/D/D/1/ \$1495

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

EDUCATION MANAGEMENT SYSTEM-



PAYROLL IS A COMPLETE SCHOOL DISTRICT PAYROLL SYSTEM WITH THE SCHOOL DISTRICT FUND ACCOUNTING SYSTEM, EMS-FUND. ALL PAYROLL FUNCTIONS ARE INCLUDED.

FIXED ASSET ACCOUNTING ?/3/2/MR/D/D/99/ \$395

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PROVIDES TRACKING AND DEPREC-IATION OF COMPANY'S ASSETS. 11 METHODS OF DEPRECIATION. ALLOWS MULTIPLE LOCATIONS, DEPARTMENTS, AND CLASSES OF ASSETS. ASSET INVENTORY MAY BE PRINTED BY LOCATION OR DEPARTMENT.

GENERAL ACCOUNTING ?/3/2/MR/D/D/10/ \$595

Author: CHRIS LOELKE 1839 1ST AVE. PRINCE GEORGE, CANADA

Seller: ARTIC DATA CORPORATION SAME

FULLY INTERACTIVE OR STAND ALONE A/R, G/L, A/P, ORDER/ ENTRY AND INVENTORY. JOB COSTING AVAILABLE WITH ALL MODULES AT EXTRA COST. CANADIAN PAYROLL SYSTEM AVAILABLE.

GENERAL LEDGER ?/3/82/S/D/D/50/ \$495

Author: KJELL ANDERSON 1926 HOLLYWOOD BLVD. HOLLYWOOD, FL 33020

Seller: MICRONETICS, INC. SAME

GL-MASTER SOPHISTICATED GENERAL LEDGER. 98 DIVISIONS, 90 SUPPORTING SCHEDULES. USER DEFINED COA AND FINAN-CIALS. MAINTAINS EARLY DETAIL. DISPLAYS 15 TRANSACTIONS PER SCREEN. NO MONTHLY CLOSINGS.

GENERAL LEDGER WRITE-UP ?/3/82/S/D/D/50/ \$1290

Author: KJELL ANDERSON 1926 HOLLYWOOD BLVD. HOLLYWOOD, FL 33020

Seller: MICRONETICS, INC. SAME CLIENT-MASTER CPA GL WRITE-UP. 98 DIVISIONS, 90 SUP-PORTING SCHEDULES. USER DEFINED COA AND FINANCIALS. MAINTAINS YEARLY DETAILS. DISPLAYS 15 TRANSACTIONS PER SCREEN. INTEGRATED PAYROLL, W2-1099-941.

G/L W/FINANCIAL WRITER ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PERFORMS GENERAL ACCOUNTING AND FINANCIAL STATEMENT PREPARATION. SUPPORTS SUB-SIDIARY LEDGERS FROM THE ACCOUNTING MODULES. PREPARES AND PRINTS BALANCE SHEETS AND P/L STATEMENTS FOR 12 ACCOUNT MONTHS. HAS FINANCIAL REPORT WRITER.

INVENTORY W/INVOICING ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

> PERMITS USER TO SELECT STAND-ARD OR AVERAGE COSTING METHOD. INVENTORY RECORDS CONTAIN PRODUCT INFORMATION, RECORD RECEIPTS OR ITEMS INTO INVEN-TORY, PRODUCE MANAGEMENT ANALYSIS. BUILT IN INVOICE MODULE INTEGRATES A/R.

JOB COST ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

TRACKS COST FOR PROJECT-ORIENTED BUSINESSES. COSTS CAN BE DEFINED IN UNITS OR IN DOLLARS. OVERHEAD CAN BE ALLOCATED AS A % OF COST OR AS FIXED AMOUNT. MONITOR COMPLE-TION STATUS FOR EACH PHASE OF JOB.

MULTI-COMPANY OPTION ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

ALLOWS MULTIPLE COMPANYS' SETS

OF BOOKS TO RESIDE ON THE SAME DATA DISK. A PARTICULAR COMPANY'S BOOKS ARE SELECTED FROM A FULL SCREEN MENU. YOU MAY ALSO CONSOLIDATE SEPARATE COMPANYS' TO CREATE A REPORT.

PAYROLL

?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PROCESSES PAYROLLS FOR SALARIED AND HOURLY EMPLOYEES FOR WEEKLY, BI-WEEKLY, AND MONTHLY PAY PERIODS. AUTO-MATICALLY COMPUTES FEDERAL, STATE AND LOCAL WITHHOLDING AS WELL AS FICA, FUTA, AND SUI.

P.O. REQ. FOR INVENTORY ?/3/2/MR/D/D/99/ \$395

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

OPTIONAL ADD-ON TO THE INVEN. CONTROL PACKAGE. PROVIDES FOR GENERATION AND PRINTING OF BOTH AUTOMATIC AND MANUAL PURCHASE ORDERS. INTERFACES WITH ACCOUNTS PAYABLE. HAS ECONOMIC ORDER QUANTITY ANALYSIS.

PROFESSIONAL TIME ACC'T ?/3/2/MR/D/D/99/ \$495

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PROVIDES RECORDING AND BILLING INFORMATION FOR OFFICES SUCH AS ATTORNEYS AND ACCOUNTANTS. TRACKS HOURS FOR EACH CLIENT BY PERSON AND SERVICE PER-FORMED. REPORTS MAY BE PRINT-ED FOR OPEN, BILLABLE, AND CLOSED FILES.

RMS-BEEF ?/3/2/S/D/D/1/ \$995

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

RANCH MANAGEMENT SYSTEM-BEEF WAS DESIGNED FOR COW/CALF MANAGEMENT TO KEEP TRACK OF



ALL INFO TO MAKE DECISIONS ON BREEDING, CALVING, CONDITION, CULLING, LOCATION, REGISTRA-TION, TREATMENT, WEANING, ETC.

RMS-SHEEP ?/3/2/S/D/D/1/ \$995

Author: HYRUM W. TATTON 2525 S. MAIN ST.,/#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

RANCH MANAGEMENT SYSTEM-SHEEP WAS DESIGNED FOR EWE/LAMB MANAGEMENT TO KEEP TRACK OF ALL INFO TO MAKE DECISIONS ON BREEDING, CONDITION, CULLING, LAMBING, LOCATION, SHEARING, TREATMENT, WEANING, ETC.

SALES ORDER ENTRY ?/3/2/MR/D/D/99/ \$595

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE.,S. BURNSVILLE, MN 55337

Seller: SAME

ALLOWS QUICK ACCESS TO ORDER INFORMATION. PRODUCES A LIST OF THE TASKS NECESSARY TO FILL A CUSTOMER'S ORDER. HAS ON-LINE ORDER INQUIRY. INTEGRATES WITH INVENTORY AND ACCOUNTS RECEIVABLE.

THE EXCALIBUR SERIES ?/3/2/MR/D/D/99/ \$595

Author: TRACY KENNEDY 324 N. ORLANDO AVE. MAITLAND, FL 32751

Seller: ARMOR SYSTEMS, INC. SAME

AN 11 MODULE, INTEGRATED ACCOUNTING PACKAGE. RUNS ON 50 MICROS AND 12 OPERATING SYSTEMS. THE MODULES ARE: G/L, A/R, A/P, PAYROLL, INVEN, WIP, COUNTER SALES, O/E, BILLING, CUSTOMER INFO, AND DEPRECIATION.

CLINIC MANAGEMENT SYSTEM ?/3/2/MR/D/D/1/ \$4200

Author: MICRO-SOTA SOFTWARE, INC. 5901 BROOKLYN BLVD. MINNEAPOLIS, MN 55429

Seller: SAME

5

CMS COMPUTERIZES ALL DATA BASES AND TREATMENT RECORD. INCLUDES W/P, G/L, A/R, REFERRAL SOURCE, INSURANCE, MED. HISTORY, AND MUCH MORE. DOCTOR DESIGNED AND DEVELOPED FOR MEDICAL AND DENTAL OFFICE. CUSTOMIZABLE.

DWP INTERFACES ?/3/2/MR/D/D/99/ \$100

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller: SAME

PROVIDES ACCESS TO THE MASTER AND DETAIL DATABASE FILES FROM GENERAL ACCOUNTING APPLICA-TIONS TO DEBASE II. SEPARATE MODULES ARE AVAILABLE FOR: G/L, A/R, A/P, PAYROLL, AND INVENTORY.

EMS-ATTEND ?/3/82/S/D/D/1/ \$495

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

EDUCATION MANAGEMENT SYSTEM-ATTEND IS A COMPLETE HIGH/ MIDDLE SCHOOL ATTENDANCE ACCOUNTING SYSTEM. IT WILL INTEGRATE WITH THE SOON TO BE RELEASED STUDENT RECORD SYS-TEMS EMS-STUDENT AND EMS-SCHEDULE.

EMS-FUND ?/3/2/MR/D/D/1/ \$2995

Author: HYRUM W. TATTON 2525 SOUTH MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

EDUCATION MANAGEMENT SYSTEM-FUND IS A COMPLETE SCHOOL DISTRICT FUND ACCOUNTING SYSTEM. IT IS DESIGNED TO COMPLETE ALL SCHOOL DISTRICT ACCOUNTING FUNCTIONS: A/P, BUDGET, G/L, PURCHASE ORDERS, ETC.

GMS-FUND ?/3/2/MR/D/D/1/ \$2995

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller:

COMPUTER SYSTEMS HOUSE SAME

GOVERNMENT MANAGEMENT SYSTEM-FUND IS A COMPLETE MUNICIPAL FUND ACCOUNTING SYSTEM. IT COMPLETES FUNCTIONS SUCH AS ACCOUNTS PAYABLE, BUDGET, GENERAL LEDGER, PURCHASE ORDERS, ETC.

GMS-UTILITY ?/3/2/MR/D/D/1/ \$1995

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

GOVERNMENT MANAGEMENT SYSTEM-UTILITY IS A COMPLETE GENERAL PURPOSE MUNICIPAL BILLING SYSTEM. IT INTERFACES WITH GMS-FUND. APPLICABLE TO ELECTRIC, GAS, GARBAGE, SEWER, TELEPHONE, WATER, ETC.

GSM-PAYROLL ?/3/2/MR/D/D/1/ \$1495

Author: HYRUM W. TATTON 2525 S. MAIN ST./#18 SALT LAKE CITY, UT 84115

Seller: COMPUTER SYSTEMS HOUSE SAME

GOVERNMENT MANAGEMENT SYSTEM-PAYROLL IS A COMPLETE MUNI-CIPAL PAYROLL SYSTEM WHICH INTEGRATES WITH MUNICIPAL FUND ACCOUNTING SYSTEM, GMS-FUND. ALL PAYROLL FUNCTIONS ARE INCLUDED.

MEDICAL OFFICE MANAGER 1.4/3/2/MR/D/D/1/ \$12K

Author: JIM ISABELLA 3004 CENTER ROAD POLAND, OH 44514

Seller: COMPUTER APPLICATIONS SAME

A PROFESSIONAL MEDICAL BILLING SYSTEM; AUTOMATIC PRINTING OF RECEIPTS, INSURANCE FORMS, AND STATEMENTS. PRACTICE BUILDING AND 25 OTHER REPORTS PRINT.

MENTAL HEALTH AGENCY 1.4/3/2/MR/D/D/1/ \$12K

Author: JIM ISABELLA 3004 CENTER ROAD POLAND, OH 44514 Seller: COMPUTER APPLICATIONS SAME

THIS SYSTEM MAINTAINS CLIENT DEMOGRAPHIC, FINANCIAL, AND M.I.S. INFO. A/R, INSURANCE FORMS AND STATEMENTS ARE PRO-VIDED ALONG WITH MANY M.I.S. AND PRODUCTIVITY REPORTS.

PHYSICIANS' OFFICE COMP. ?/3/2/MH/D/D/?/ \$5750

Author: JODIE BAKER 1240 KONA DRIVE COMPTON, CA 90220

Seller: PROFESSIONAL SYS. CORP. SAME

PHYSICIANS' OFFICE COMPUTER IS COMPLETE IN-OFFICE MEDICAL SOFTWARE SYSTEM FOR PATIENT FINANCIAL AND CLINICAL DATA MANAGEMENT AS WELL AS APPOINT-MENT SCHEDULING.

 $\begin{array}{c} \langle \rangle & \langle \rangle \\ \hline \textbf{TURBODOS^{*}UTILITY^{*}SERIAL} \\ \langle \rangle & \langle \rangle \\ \end{array}$

BACKREST ?/3/2/MH/D/D/99/ \$180

Author: GUY SICILIANO 17 WEST 17TH ST. NEW YORK, NY 10011

Seller: STOK SOFTWARE, INC. SAME

BACKREST IS A USER FRIENDLY HARD DISK BACKUP AND RESTORE PROGRAM. FEATURES: SPLITS LARGE FILES, TRAPS AND HANDLES ERRORS, FAULT TOLERANT RESTORATION, AND PRINTED REPORTS.

BI-289 ?/3/2/MR/D/D/99/ \$375

Author: DENNIS L. DRUMMOND 6441 S.W. CANYON COURT PORTLAND, OR 97221

Seller: CONTROL-C SOFTWARE, INC. SAME

BUSINESS BASIC INTERPRETER FOR MICROS. PROVIDES ALL OF THE FEATURES OF AN OPERATING SYS-TEM. FEATURES LIBRARIAN, SYSTEM UTILITY FOR FILE CON-VERSION. INCLUDES MASKING, ERROR HANDLING, & ISAM MGR.

COSORT ?/3/81/S/O/A/99/ \$150 Author: PAUL FRIEDLAND BOX W MANHASSET, NY 11030

Seller: INFORMATON RESOURCES SAME

A GENERAL PURPOSE SORT/MERGE FACILITY. MAY BE USED AS A UTILITY PROGRAM OR INTEGRATED INTO APPLICATIONS. MINIMAL TIME ALGORITHM SORTS OR MERGES FROM A FILE OR RECORDS WHICH ARE FIXED OR VARIABLE.

PHD INTEGRATED DATA MGR. ?/3/2/MR/D/D/99/ \$495

Author: MICRO BUSINESS APPLICATIONS 12281 NICOLLET AVE., S. BURNSVILLE, MN 55337

Seller:

RELATIONAL DBMS THAT INTE-GRATES MBA'S ACC'T MODULES OR FUNCTION AS STAND ALONE. USE MAY DESCRIBE VIEWS OF INFO BY DRAWING FORMS ON THE SCREEN AND RELATING INFO IN SEPARATE FILE BY NAME.

T-DUMP 1.4/3/81/S/D/D/1/ \$40

Author: JIM ISABELLA 3004 CENTER ROAD POLAND, OH 44514

Seller: COMPUTER APPLICATIONS SAME

A FILE DUMP UTILITY FOR TURBO-DOS. VERY SIMILAR TO THE OS-65U FDUMP IN OPERATION AND OUTPUT. VERY FAST OPERATION USING COMPILED PASCAL.

TURBO-PLUS 1.4/3/2/S/D/D/99/ \$500

Author: R. JUNCSOK 276 FIFTH AVE. NEW YORK, NY 10001

Seller: MICROSERVE, INC. SAME

AN EXTENDED SUPPORT PACKAGE FOR USERS OF THE TURBODOS OPERATING SYSTEM. ELECTRONIC MAIL, BATCH SCHEDULER, TWX, ARE JUST SOME OF THE UTILITIES THIS PACKAGE PROVIDES.

 AGRICULTURAL MGMT. SYSTEM 1.4/2/2/MH/D/D/1/ \$10K

Author: DR. DANIEL M. SWEGER 2515 C. EAST MARKET ST. HARRISONBURG, VA 22801

Seller: IHS COMPUTER SERVICES SAME

MGMT. SYSTEM FOR LIQUID FERTI-LIZER DEALERS. INCLUDES COMP-REHENSIVE CROP MGMT, FERTILIZ-ER APPLICATIONS, FIELD HISTORY AND EPA REPORTING. FULLY IN-TEGRATED WITH ACCOUNTING SYSTEM.

ALTERNATIVE DBMS ?/4/82/S/D/M/20/ \$150

Author: CHUCK CARLETTA 2324 E. 8TH ST. DAVIS, CA 95616

Seller: THOMAS TECH. SERVICE RD #1, BOX 135 LINDEN, PA 17744

COMPLETELY MENU DRIVEN EASY TO USE, HAS ADVANCED REPORT WRIT-ER, WILL SORT TO 10 LEVELS, VERY FAST. ULTIMATE IN DBM SYS FOR OSI COMPUTERS. ALLOWS YOU TO CREATE, MODIFY, ADD, DELETE INQUIRE & FORM CUSTOM REPORTS FROM YOUR DATA BASE, SORTS RE-CORDS ON ANY FIELD. EASY TO TO MODIFY. WRITTEN IN BASIC.

EIS INVENTORY SYS. V .30 1.2/D/7/MR/D/D/3/ \$1700

Author: RICKY R. PETERSON 206 PINE VALLEY DRIVE WARNER ROBINS, GA 31093

Seller: ELECTRONIC INFO. SYSTEMS SAME

HANDLES PERPETUAL INVENTORY, INVOICING, ACCOUNTS RECEIV-ABLES, BILL OF MATERIALS AND PURCHASE ORDERS. INTEGRATES WITH ORDER ENTRY AND ACCOUNT-ING SYSTEMS. MANUFACTURING ORIENTED.

BIS ORDER ENTRY V3.0 1.2/D/7/MR/D/D/1/ \$1500

Author: RICKY R. PETERSON 206 PINE VALLEY DRIVE WARNER ROBINS, GA 31093

Seller: ELECTRONIC INFO. SYSTEMS SAME

THIS PROVIDES CONTROL OF ORDERS, INVENTORY STOCK COM-MITMENT, AND ALL NECESSARY IN- FORMATION FOR MAXIMUM CUSTOMER SERVICE WITH MINIMAL INVENTORY LEVELS. INTERFACES WITH INVENTORY SYSTEM.

EQUIPMENT MANAGER 1.4/0/82/S/D/D/1/ \$400

Author: DAMON CURRY 2 EAGLE DRIVE DAYTON, OH 45431

Seller: SOFTOUCH, INC. SAME

DATA BASE MANAGER FOR RENTAL EQUIPMENT. TRACKS DESCRIPTION, CUSTOMER/LOCATION, MODEL #, SERIAL #, RENT, SHIP DATE AND RETURN DATE. MENU DRIVEN MULTIPLE SORT LEVELS.

FINANCE & INSURANCE QUOTATION 1.44/2/71/MH/D/A/1/ \$4000

Author: SOFTOUCH, INC. 3120 FAR HILLS DAYTON, OH 45429

Seller: SAME

INTEGRATED VEHICLE & CUSTOMER DATA BASE W/LOAN CALCULATION INCLUDING CREDIT LIFE AND ACCIDENT/HEALTH INSURANCE. ACTIVE (READY TO SELL) AND INACTIVE (SOLD W/CUSTOMER ID) VEHICLE DATA BASE. PRINTS LOAN CONTRACTS. MAILING LABELS. CUSTOM REPORTS. USES COMPUTER AS AID IN CLOSING SALES.

FORMS JOBBER

1.4/D/7/MH/D/D/1/ \$?

Author: JIM BOOTH 806 FREDERICK ST. HAGERSTOWN, MD 21740

Seller: CUSTOM COMPUTING, INC. SAME

ADDRESSES INVENTORY NEEDS OF BUSINESS FORMS JOBBERS. PRO-JECTED REORDER INFOMATION BASED ON PREVIOUS USE. TIES TO OUR "SUPERTRAK" ACCOUNTING PACKAGE. MANY REPORTS.

FULLY INTEG. ACCOUNTING SYS. 1.4/D/71/HR/D/D/1/ \$4000

Author: FRANK LACY 5302 LAKE WASHINGTON, N.E. KIRKLAND, WA 98033

Seller: ALPHA COMPUTER SYSTEM SAME

RATED THE BEST BY A LEADING

TRADE PUB. YOU CAN GET CASH FLOW AND P&L COMPARED TO PRIOR YEAR'S ACTUALS OR BUDGETS FOR THE CURRENT PERIOD. INCLUDES INV./INVOICE, G/L, CASH FLOW, P&L, A/R, A/P, & MORE.

GENERAL ACCOUNTING 1.2/D/71/MH/O/A/3/ \$2995 Author:

DALE VICTOR, CPA BOX 331 SAUK RAPIDS, MN 56379

Seller: SAME

COMPREHENSIVE INTEGRATED ACCOUNTING PACKAGE. USER DE-FINES OPTIONS USED TO CUSTOM-IZE PACKAGE FOR WHOLESALE, RETAIL, OR SERVICES FIRM.

INSURANCE SYSTEM 1.4/0/82/S/D/D/1/ \$?

Author: HENRY C. KAFTON, JR. 136 W. 500 S. BRIGHAM CITY, UT 84302

Seller: COMPUTER UTILITIES/UTAH SAME

VER. 1.4 UNDERGOING REVISION. VERSION 2.0 AVAILABLE JUN-JUL '85. BAL. FWD. REC. INV CAN BE ENTERED & PRINTED OR ENTERED AFTER FACT. CALC SERVICE CHG. PRINT ATB & STATEMENTS. GL, SOL, BROKER DATA ACCUMU.

INV. MGMT. SALVAGE YARDS 1.4/D/7/MH/D/D/1/ \$?

Author: JIM BOOTH 806 FREDERICK ST. HAGERSTOWN, MD 21740

Seller: CUSTOM COMPUTING, INC. SAME

PARTS AND CARS. PRINTS TEAR DOWN SHEETS FROM STORED RE-FERENCE NUMBERS. HANDLES HOLLANDER & MITCHELL TYPE REF. NUMBERS. COUNTER INVOICING, FULL A/R AND BILLING. TIES TO FULL G/L & P/R PROGRAMS.

INVENTORY MANAGER

1.4/0/82/S/D/D/1/ \$400

Author: DAMON CURRY 2 EAGLE DRIVE DAYTON, OH 45431

Seller: SOFTOUCH, INC. SAME

INVENTORY RECORDS OF MFG. PARTS AND RAW MATERIALS CON- SUMPTION. TRACKS # OF PARTS MFG., SHIP, RETURN, RAW MATER-IAL RECEIVED, CONSUMED, SCRAP-PED. CALCULATES FINISHED GOODS AND RAW MATERIAL INVEN. VALUE.

JOB COST ACCOUNTING 1.4/D/71/SH/0/0/1/ \$1500

Author: PAUL BLAKEMORE 228 SCHOOL ST. GENEVA, IL 60134

Seller: PROCESSOR SYSTEMS SAME

JOB COST ACCOUNTING.

JOB COST ACCOUNTING SYSTEM 1.4/D/2/MH/D/D/1/ \$8000

Author: DR. DANIEL M. SWEGER 2515 C. EAST MARKET ST. HARRISONBURG, VA 22801

Seller: IHS COMPUTER SERVICES SAME

CONTRACTOR'S ACCOUNT SYSTEM FULLY INTEGRATES WITH ACCOUNT-ING SYSTEM. INCLUDES SUBCON-TRACTOR, SCALE WAGES, RETAIN-AGE, CHANGE ORDERS. MAINTAINS ALL DETAIL & GIVES MONTHLY, YEARLY, & J.T.D. REPORTS.

LEVEL LOAN 1.4/0/82/S/0/0/1/ \$90

Author: J. M. TIRINO 16 MAPLE AVE. WEST NYACK, NY 10994

Seller: NORTH EAST FINANCIAL SAME

LEVEL LOAN WITH BALLOON. LOAN AMORT W/BALLOON PAY COMPUTES PAYMENT GIVEN INTRATE. ACCEPTS 13 DIGIT INPUT TO 23 DIGIT OUTPUT. REQUIRES EAP MULT PRECISION ARITHMETIC TO RUN, WHEN ACCURACY COUNTS.

PATIENT SERVICE SYSTEM 1.4/D/71/MH/D/D/4/ \$2995

Author: WILLIAM W. GIBBS 21 N. CENTRAL AVE. STAUNTON, VA 24401

Seller: SHENANDOAH MICROCOMPUTER SAME

A COMPLETE MEDICAL & DENTAL ACCOUNTING & BILLING SYSTEM. FEATURES INCLUDE MAILER STATE-MENTS, SUPERBILLS, AGED RE- CEIVABLES, RECALL NOTICES, & INSURANCE FOR AMA, ADA, BC/BS.

PAYROLL

1.2/D/71/MH/A/A/5/ \$795

S Author: DALE VICTOR, CPA BOX 294 SAUK RAPIDS, MN 56379

Seller: SAME

COMPLETE PAYROLL CHECKWRITING PACKAGE. FUNCTIONAL FOR LARGE ORGANIZATIONS WITH 3000 EMP-LOYEES OR MORE, OR A SMALL BUSINESS. UNLIMITED DEDUCTIONS PAY TYPES, ALLOCATED TIPS AND MULTIPLE STATES.

POS/INVENTORY CONTROL 1.4/0/7/MH/D/D/1/ \$3000

Author: THOMAS P. CROCKETT 4 SWIMMING RIVER RD. LINCROFT, NJ 07738

Seller: IDEAL BUSINESS SYSTEMS SAME

MAINTAINS ONLINE INVENTORY CONTROL AT TIME OF SALE. HANDLES ALL DAILY FUNCTIONS OF LIQUOR STORE INCLUDING DAILY SALES REPORTS, TREND ANALYSIS AND REORDER REPORTS. ACCOUNTING OPTIONS AVAILABLE.

PREMIUM LEVERAGING 1.2/0/82/S/0/A/1/ \$200

Author: WAYNE R. COLE, CLU 805 CHUMLEIGH ROAD BALTIMORE, MD 21212

Seller: COMPUTER WONDERS, LTD., SAME

PIGGY BACKING OR PREMIUM LEV-ERAGING IS A SALES AID OR FLANNING TOOL WHICH USES THE CASH VALUES AND/OR DIVIDENDS OF OLD AND NEW INSURANCE TO HELP PURCHASE MORE INSURANCE OR CHANGE PREMIUM OUTLAYS.

PROFESSIONAL INVESTMENT 1.2/D/71/SH/D/D/2/ \$1500

Author: RICKY R. PETERSON 206 PINE VALLEY DRIVE WARNER ROBINS, GA 31093

Seller: ELECTRONIC INFO. SYSTEMS SAME

SYSTEM EFFICIENTLY MAINTAINS PORTFOLIOS, STOCK/BOND INFORM-ATION, TRACKS TRANSACTIONS, ASSIGNS MARKET CLASSIFICATIONS AND PROVIDES REPORTS NECESSARY FOR MANAGERS AND CLIENTS.

SSS JOB COST ACCOUNTING 1.4/D/71/S/D/D/3/ \$3500

Author: BILL LAPLANTE 700 PAPWORTH AVE. #103 METAIRE, LA 70005

Seller: SOFTWARE SALES & SUPPORT SAME

USED MOSTLY IN THE CONSTRUC-TION INDUSTRY. COMPLETELY INTEGRATED. CONTAINS PAYROLL, GENERAL LEDGER, ACCOUNTS RE-CEIVABLE AND ACCOUNTS PAYABLE. JOB COSTING MODULES AVAILABLE SEPARATELY FOR \$700 EACH.

SSS TEMPORARY LABOR 1.4/D/71/SH/D/D/1/ \$4200

Author: BILL LAPLANTE 700 PAPWORTH AVE. #103 METAIRIE, LA 70005

Seller: SOFTWARE SALES & SUPPORT SAME

WRITTEN EXCLUSIVELY FOR TEMPO-RARY LABOR COMPANIES. COM-PLETELY INTEGRATED. GENERATES INVOICES FROM PAYROLL. INCLUD-ES PAYROLL, INVOICE/BILLING, GENERAL LEDGER, ACCOUNTS RE-CEIVABLE AND ACCOUNTS PAYABLE.

SUPERTRAK 1.4/D/71/MH/D/D/7/ \$?

Author: JIM BOOTH 806 FREDERICK ST. HAGERSTOWN, MD 21740

Seller: CUSTOM COMPUTING, INC. SAME

FULL GENERAL LEDGER INCLUDING FINANCIAL STATEMENTS. A/R, A/P, C/R, C/D, AJE JOURNALS. ORDER ENTRY OR POINT OF SALE INVOICING. PAYROLL OPT. OPEN ITEM OR BALANCE FWD A/R. INVENTORY, BILLING, DEPRECIATION.

TAX II ?/0/81/S/0/A/2/ \$50

Author: ROBERT BALDASSANO 4045 ASHBROOK CIRCLE SAN JOSE, CA 95124

Seller: RSB ENTERPRISES SAME

TAX II COMPUTES 1040 SCHEDULES

A,B,C,G, & FORM 625]. INCLUDES UTILITIES, TAX TABLES AND TAX FORM FILES. PRINTS ANNOTATED LISTING OF RESULTS. PRINT ON FORMS OPTIONAL. \$10.

THE CARPET SYSTEM

].4/D/7/MH/D/D/1/ \$?

Author: JIM BOOTH 806 FREDERICK ST. HAGERSTOWN, MD 21746

Seller: CUSTOM COMPUTING, INC. SAME

COMPLETE INVEN. FOR CARPET. RESERVATIONS AND ORDER PROCES-SING. USES FEET AND INCHES. REPORTS INCL: CUTTING, TAGS, SALES, RESERVED, AVAILABLE TO SELL, ON ORDER, AND MANY MORE. IN USE BY MAJOR DEPARTMENT STORE CHAIN.

SUPER PLANNER PLUS 1.42/8/82/MH/P/0/1/ \$150

Author: RICHARD E. REED 450 N. GREEN TEHACHAPI, CA 93561

Seller: SAME

PLANNER PLUS ENHANCEMENTS AUTO COPY LARGE MODELS INTERFACES TO DMS FILES, SENDING AND RECEIVING DATA, SENDS DATA TC WORD PROCESSING FILES. SOLD ONLY TO REGISTERED OWNERS OF 65-U PLANNER/KEYCALC OR IN CONJUNCTION WITH THAT OSI SOFTWARE.

⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>
 ⟨>

BROADT YMCA PACKAGE].4/D/71/MH/D/D/1/ \$3000

Author: DAVID BROADT 517 N. 4TH ST. LEWISBURG, PA 17837

Seller: BROADT COMPUTERS SAME

DESIGNED FOR THE SPECIFIC NEEDS OF YMCA ORGANIZATIONS. INCLUDES INTEGRATED A/P, G/L, AND PAYROLL, AS WELL AS, POINT OF SALE FOR MEMBERSHIPS AND PROGRAMS, MEMBERSHIP BILLING AND AGING WITH STATISTICS.

CONDITION MAILING LABELS 1.2/0/82/S/0/A/1/ \$75



Author: WAYNE R. COLE, CLU 805 CHUMLEIGH ROAD BALTIMORE, MD 21212

Seller: COMPUTERS WONDERS, LTD., SAME

CONDITIONAL MAILING LABELS. PREPARE MAILING LABELS OR DO SCREEN SEARCH FROM DATA FILE. LETS OPERATOR CHOOSE PERSONAL OR BUSINESS ADDRESS AND KEY OFF OF CERTAIN CONTENTS OF A SPECIFIC FIELD.

PHONE COMMAND SYSTEM ?/8/82/M/D/D/1/ \$1000

Author: DAMON CURRY 2 EAGLE DRIVE DAYTON, OH 45431

Seller: SOFTOUCH, INC. SAME

MULTI-USER, TERMINAL PLUS PHONE ACCESS TO COMPUTERIZED MESSAGE RELAY SYSTEM. BUFFERS INCOMING MESSAGES, RELAYS TO MANY PRINTERS (SERIAL). USES: HOSPITAL ADMISSION/ DISCHARGE TRANSFER.

PROBIT ?/8/81/S/O/A/1/ \$3Ø

Author: ROBERT T. KINTZ 104 COUNCIL ROCK AVE. ROCHESTER, NY 14610

Seller: SAME

PROBIT IS A PACKAGE TO COMPUTE PROBIT ANALYSIS, A STATISTICAL TECHNIQUE USED TO COMPUTE THRESHOLD IN INSECTICIDAL PHARMACOLOGICAL AND TOXI-COLOGICAL ASSAYS.

RX. DR. PACKAGE 1.4/D/71/MH/O/A/1/ \$4500

Author: MARK RATHGEBER R. PH. 1137 HOBBS MILL DR. ST. LOUIS, MO 63011

Seller: RATHGEBER & ASSOCIATES. SAME

AN EXTREMELY FAST & EFFICIENT PHARMACY SYSTEM. COMPLETE PRESCRIPTION PROCESSING. COMPREHENSIVE DRUG FILE WITH UPDATES. NUMEROUS DAILY MANAGEMENT REPORTS. OPTIONAL ACCOUNTING AND NURSING HOME SOFTWARE.

THE CLUB CHAMPION 1.4/D/71/MH/D/D/Ø/ \$? Author: JIM BOOTH 806 FREDERICK ST. HAGERSTOWN, MD 21740

Seller: CUSTOM COMPUTING, INC. SAME

COMPLETE GOLF COURSE AND CC ACCOUNTING AND RECORD KEEPING MEMBERSHIP, AUTOMATIC BILLING, USGA HANDICAPS, INVENTORY, GIFT CERTIFICATES, MAINTENANCE INVOICING. FULL A/R, G/L, AND SUBJOURNALS.

TIME AND TASK PLANNER 1.4/D/71/S/D/D/1/ \$300

Author: JOHN HUNTLEY 3223 BROSS ROAD HASTING, MI 49058

Seller: GANDER SOFTWARE, LTD. SAME

TIME PLANNING. 5 USERS CAN KEEP SORTED 'TO DO' AND 'FUTURE' LISTS AND THEN TRANS-FER INFORMATION TO ONE OF 5400 APPOINTMENT TIMES WITHIN 60 DAYS - DESIGN YOUR OWN DAY BOOK. RUN, PRINT CALENDARS FOR ANY MONTH/YEAR. A FEW MINUTES A DAY WILL KEEP YOU ON TRACK AND SAVE TIME.

TP-2 1.4/0/82/MH/0/0/1/ \$200

Author: ISOTRON, INC. 140 SHERMAN ST. FAIRFIELD, CT Ø6430

Seller:

WP PACKAGE CREATED FOR DAILY CORRESPONDENCE, MEMOS, LET-TERS, ETC. HAS FILE/MERGE WITH DMS DATA FILES, HIDDEN ATTRIBUTES, ON SCREEN FORMAT-TING, TYPE AHEAD BUFFER, & LINK FEATURES FOR LONG DOCUMENTS.

ALPHA/OMEGA DATA SYSTEM 1.4/0/81/S/D/D/2/ \$195

Author: DR. DANIEL M. SWEGER 2515 E. MARKET ST. HARRISONBURG, VA 22801

Seller: IHS COMPUTER SERVICES SAME THE DBMS FUNCTIONS: FILE CREATE, FILE SORT, FILE EDITOR /MAINT., REPORT WRITER, FILE MERGE/TRANSFER, MAILING LABEL PRINTING, DELETE/REPACK FILE RECORDS, AND A PROGRAM GENERATOR. CAN DOWNLOAD DATA.

COMPARE V1.0 1.2/8/82/S/D/D/0/ \$50

Author: RICKY R. PETERSON 206 PINE VALLEY DR. WARNER ROBINS, GA 31093

Seller: ELECTRONIC INFO. SYSTEMS SAME

PROVIDES AN AUDIT TRAIL FOR SOFTWARE CONFIGURATION MANAGEMENT BY COMPARING PROG-RAM LISTING AND GIVING AN OUT-PUT OF LINES THAT HAVE BEEN ADDED, EDITED, OR DELETED, AND LABELING AS SUCH.

DATA TRANSFER 1.4/D/71/S/D/D/1/ \$175

Author: RUSSEL D. DAUGHERTY P.O. BOX 719 PARKERSBURG, WV 26101

Seller: KPS BUSINESS SYSTEMS SAME

TRANSFER DATA FROM COMPUTER TO ANOTHER COMPUTER AT HIGH SPEED.

BIS BASXR V1.0 1.2/D/71/SH/D/D/4/ \$50

Author: RICKY R. PETERSON 206 PINE VALLEY DRIVE WARNER ROBINS, GA 31039

Seller: ELECTRONIC INFO. SYSTEMS SAME

BASIC CROSS REFERENCE IS A PROGRAMMING TOOL FOR DEBUGGING AND MODIFICATION. IT WILL LIST ALL VARIABLES, LIST ALL DISK RELATED OPERATIONS, AND LIST ANY RESERVED WORD/ COMMANDS.

HF COPY V2.0 1.2/D/71/S/D/D/6/ \$75

Author: RICKY R. PETERSON 206 PINE VALLEY DRIVE WARNER ROBINS, GA 31093

Seller: ELECTRONIC INFO. SYSTEMS SAME

Continued on page 17

REM This routine gets the file names, passwords, and the output REM device number from the user. Then it opens the files and REM initializes variables. REM

PRINT

PRINT "File comparer: see source code for doc."

PRINT INPUT "Control file, password"; CFILE\$, CPASS\$ IF CPILE\$="" OR CFILE\$="ABORT" THEN STOP IF CPASS\$="" OR CPASS\$="ABORT" THEN STOP

INPUT "Test file, password"; TFILE\$, TPASS\$ IF TFILE\$="" OR TFILE\$="ABORT" THEN STOP IF TPASS\$="" OR TPASS\$="ABORT" THEN STOP

INPUT "Output device number"; DV IF DV = 0 THEN DV = 1

CLOSE OPEN CFILES, CPASSS, 1 OPEN TFILES, TPASSS, 2 CCOUNT = 0 TCOUNT = 0 CLINE\$ = "" TLINES = "" RETURN

AN OFFER

The savings in debugging, modification, and maintenance has more than paid for the devel-opment of GASP. We have about 50,000 lines of GASP code. Written in BASIC, these appli-

LISTINGS CONT. FROM PAGE 16

THE HARD DISK/FLOPPY COPIER BACKS UP FILES FROM A HARD DISK ON TO 1 OR MORE FLOPPIES (LARGE FILES), OR THE REVERSE, FROM FLOPPIES TO HARD DISK. ESSENTIAL PROTECTION IN MENU-DRIVEN FORMAT.

SORT/MERGE

1.4/8/81/SH/D/D/10/ \$89

Author: PEEK(65), INC. P.O. BOX 347 OWINGS MILLS, MD 21117

Seller: SAME

SANDER'S MACHINE LANGUAGE SORT/MERGE IS A FAST DISK BASED SORT/MERGE. MANUAL TELLS HOW TO CALL AND RETURN TO ANY BASIC PROGRAM ON ANY DRIVE. ALMOST UNLIMITED FILE SIZE. AN OSI STANDARD, ONLY FROM PEEK(65), INC.

TERMINAL EXTENSIONS 1.4/8/81/S/D/D/2/ \$50

Author: SOFTWARE CONSULTANTS 6435 SUMMER AVE. MEMPHIS, TN 38134

Seller: PEEK(65), INC. P.O. BOX 347 OWINGS MILLS, MD 21117

TERMINAL EXTENSIONS GIVES DIRECT CURSOR POSITIONING, MNEMONICS, & NUMBER FORMATTING MORE POWERFUL THAN PRINT

cations would be unmanageable. GASP is far from a perfect development environment, but it is a definite improvement. If you want more information about GASP and/or our text file utilities, please write me at:

USING. READY FOR ACT 5, TELEVIDEO, HAZEL 1420. OTHERS MAY BE ADDED TO THE TABLE.

THE DATA SYSTEM 1.4/D/71/MH/D/D/11/ \$650

Author: JOHN HUNTLEY 3223 BROSS ROAD HASTINGS, MI 49058

Seller: GANDER SOFTWARE, LTD., SAME

TOTAL DBM FOR 65U. FAST SORTS, PACKS, & MERGES. CONDITIONS & STORE REPORTS, CALCS, EDITORS, MERGES. USE 7 KEYS. DEFINED EDITORS. POST TO FILES. EDIT/FORMAT GLOBALLY. WRITE FOR INFORMATION.

Т-МПМ 1.2/D/71/HR/D/D/70/ \$1200

Author: ED COOPER 1430 MINER ST. DES PLAINES, IL 60016 Seller: ED COOPER & ASSOCIATES SAME

TIMES SHARING MULTI-USER MANAGER FOR CD-23, CD-36, CD-74. LOG-ON, LOG-OFF, FEATURES ALLOWS MICRO COMPUTER TO ACT LIKE LARGE TIME SHARING COMPUTERS WITH COMPLETE BACK-UP CAPABILITIES. ACCOMMODATES SIX USERS. ×

10 REM COMPARE UTILITY 8/12/85 5:30PM GASP VER. 2.31 20 GOSVB210

- 30 IFNOT(CLINE\$<>"OK"ANDTLINE\$<>"OK")THEN80
- 40 IFNOT(CLINE\$=TLINE\$)THEN60 50 GOSUB110:GOTO70
- 60 GOSUB160 70 GOTO30
- 70 GOTO30 80 PRINT#DV:PRINT#DV,"Compare complete." 90 PRINT#DV,"Control file:"CCCUNT"lines, "INDEX(1)"bytes." 100 PRINT#DV,"Test file:"TCOUNT"lines, "INDEX(2)"bytes.":CLOSE:END 110 IFNOT(CLINE\$<>"OK")THEN130 120 INPUT%1,[72,"A"]CLINE\$:CCCUNT=CCOUNT+1 130 IFNOT(TLINE\$<>"OK")THEN150
- 140 INPUT%2, [72, "A"]TLINES: TCOUNT=TCOUNT+1
- 150 RETURN
- 160 PRINT#DV:PRINT#DV, "C"CCOUNT;CLINE\$:PRINT#DV, "T"TCOUNT;TLINE\$
- 100 IFNOT (VAL(TLINE\$))VAL(CLINE\$))THEN190 180 INPUT\$1,[72,"A"]CLINE\$:CCOUNT=CCOUNT=1:GOTO200 190 INPUT\$2,[72,"A"]TLINE\$:TCOUNT=TCOUNT+1
- 200 RETURN 210 PRINT:PRINT"File comparer: see source code for doc.":PRINT
- 220 IMPUT"Control file, password";CFILE\$,CPASS\$
 230 IFCFILE\$=""ORCFILE\$="ABORT"THENSTOP
- 240 IFCPLASS=""ORCPASS="ABORT"THENSTOP 250 INPUT"Test file, password";TFILE\$,TPASS\$ 260 IFTFILE\$=""ORTFILE\$="ABORT"THENSTOP

- 260 ITTTILES=""ORTFILES="ABORT"THENSTOP 270 IFTPASSS="#ORTFILENSTOP 280 INPUT"OUEput device number";DV:IFDV=OTHENDV=1 290 CLOSE:OPENCFILES,CPASS\$,1:OPENTFILE\$,TPASS\$,2:CCOUNT=0:TCOUNT=0 300 CLINES=":TLINES=":RETURN

Gale's BI-RITE Supermarkets 5360 Mayfield Rd. Lyndhurst, OH 44124

or

SGOQ@tc.cc.cmu.edu (on many networks).

\star

OS-65U DATA FILES AND OTHER **MYSTERIES:** FEAR AND LOATHING GUIDE

By: Rick Trethewey 8 Duran Court Pacifica, CA 94044

I've seen a lot of video system owners beginning to explore OS-65U in order to take advantage of the extra abilities in that operating system handling for handling data files. After the initial shock wears should find themselves pretty much at born much at home with OS-65U's BASIC. After all, most of the language is identical to its 65D counterpart. Still, the differences are significant. In this article, I hope to describe OS-65U's data file handling and how to get the most out of it.

To begin, we need to look at how OS-650 deals with the disk, no matter what the purpose is for using it. Under 650, every byte available is referred to by an "address". If you think of these addresses like memory locations, you'll be on the right track (pardon the pun). All requests to 650 to access the disk are done by passing information to 65U as to which drive, and disk address is involved and if data is to be read from or written to the disk. In addition, a memory address is specified, telling the operating system where to put information read or, con-

17 PEEK [65] December, 1985

versely, where the data to be written is located. Finally, the number of bytes to be transferred is also given to the operating system. I'll be discussing how this information is passed to 65U a bit later on. When 65U needs to deal with any byte/address on the disk, it first reads the track containing the byte/ address into an internal buffer and then it goes on to transfer the information needed to the desired memory location outside of the buffer. My main point here is that because the operating system reads a chunk of data before giving it to the user, 65U can effectively read or write as little as a single byte at a time.

There is a price to be paid for this ability, though. First, the operating system has to include space for this internal buffer. That's 3584 bytes out of a normal 48K workspace. Next, because there are intermediate steps involved between all reads and writes, there is some time lost as compared to 65D which has no such buffering. Finally, there is another price to be paid that is also a fea-ture. Because the operating system is controlling this buffer, it can avoid unnecessary disk accesses because the information involved in many operations will already reside in memory and can be immediately given to the user. The price incurred by this feature is that in a multi-user environment, the contents of any disk drive may not be the "latest" version of the information because one user may not have had his changes stored on disk yet because of the buffering. buffering. Over the years, this problem has had several solutions thrown at it by OSI. Even single-user systems can run into trouble because of this if they are careless careless about switching diskettes. We will return to this level of the operating system a bit later. For now, just keep this information in the back of your mind.

BASIC under OS-65U supports up to 8 data files simultaneously, and all without the need of adding a separate disk buffer for each one in front of your programs. The single internal buffer takes care of this for us. However, instead of referring to each open data file by device number, as we do under OS-65D, OS-65U refers to each file by a "channel number". We still use the BASIC commands INPUT and PRINT, but instead of "#6" or "#7", we use "%x" where "x" is the channel number involved in the operation, from 1 to 8. But I'm jumping the gun a bit here.

Before you can access a data file, you have to OPEN it. The command to do this under OS-65U is;

OPEN "FNAME", "PASS", CN

where "FNAME" is the name of the file, "PASS" is its pass-word, and "CN" is the channel number to use to support that file. NOTE: The file name and password are going to be different for any file you use... well, at least the file name should be. Since passwords are limited to four characters and most people use "PASS" for personal applications, you can almost always try "PASS" as the password and be successful. The password is optional to a certain extent. If you to a certain extent. If you try to access a file with the wrong password or by omitting the password from the command entirely, you will be limited to the kinds of access rights the file's creator defined when it was created. Ergo, Ergo, you may not get an error message immediately if you use the wrong password. Many times, you won't see an error message at all until you ex-ceed your "authority".

Conversely, when you've finished with a data file, you have to close it. The command to do this is simply;

CLOSE CN

where "CN" is the channel number. If no channel number is specified, all open channels will be closed. "CLOSE" has another effect that we need to pay attention to here, and that is that CLOSE forces OS-65U to write the current contents of its internal buffer back to the disk drive if that information has been altered since it was originally read from the disk.

OS-65U'S BASIC gives you explicit control over where in the data file the reads or writes begin. It does this with the INDEX command. The INDEX command syntax is;

INDEX<CN>=xxxx

where "CN" is the channel number and "xxxx" is the numeric index to the file in question. In real terms, the INDEX is the value added to the file's disk address to determine where the information is to reside on the disk, and 65U's pointer for that data channel is set to that value. This pointer is used for both input and output operations for the channel involved and as each byte is retrieved or sent, the pointer is automatically incremented. If you remember that the first byte of the file is located at INDEX Ø, then logically, setting the INDEX to any other value moves the beginning of the operation further along in the file. For sequential data files, INDEX is most often used to set the pointer to the beginning or the end of the data file. However, it is random access files that make the INDEX command really valuable.

Under our old friend OS-65D, unless you go to a significant amount of extra effort, random access files can only be used with record lengths that must be 2, 4, 8, 16, 32, 64, 128, or 256 bytes. The limited amount of space allocated for OS-65D is at the root of this limitation, and it is fine for personal applications, but it is extremely limiting, cumbersome, and inefficient in its use of disk space. It doesn't take a complex example to il-lustrate this inefficiency. If your applicaton's record length exceeds one of these values by even a single byte, you are forced to go to the next higher value and there is no built-in support past the 256 byte limit. Further, fields within these records are stored sequentially. This means that in order to access any field within a record, you must read in the entire record first. Similarly, when you need to update the record, you Continued on page 20



THE DATA SYSTEM

- Stored Report Formats
- Stored Jobs, Formats, Calcs.
- Multiple Condition Reports
- Multiple File Reports
- Calc. Rules Massage Data
- Up to 100 Fields Per Record

HARDWARE REQUIREMENTS: 48K OSI, Hard Disk, serial system, OS-65U 1.42 or Later; Space required: 1.3 megabytes for programs and data.

- User Designed Entry/Edit Screens
- Powerful Editor
- Merges Append, Overlay, Match
- Posting Batch Input
- Nested Sorts 6 Deep
- Abundant Utilities

PRICE: \$650.00 (User Manual \$35.00, credited towards TDS purchase). Michigan residents add 4% sales tax. 30 day free trial, if not satisfied, full refund upon return.

TIME & TASK PLANNER

30 DAY FREE TRIAL - IF NOT SATISFIED, FULL REFUND UPON RETURN

- "Daily Appointment Schedule"
- "Future Planning List" sorted
- "To Do List" by rank or date
 - A SIMPLE BUT POWERFUL TOOL FOR SUCCESS

HARDWARE: 48K OSI, 8" floppy or hard disk, serial terminal system, OS-65U v. 1.3 or later.

- Work Sheets for all Aspects
- Year & Month Printed Calendar
- Transfers to Daily Schedule

PRICE: \$300.00 (User Manual, \$25.00, credited toward TTP purchase). Michigan residents add 4% sales tax.

FINANCIAL PLANNER

- Loan/Annuity Analysis
- Annuity 'Due' Analysis
- Present/Future Value Analysis

HARDWARE REQUIREMENTS: 48K OSI, 8" floppy or hard

 Sinking Fund Analysis Amortization Schedules

Interest Conversions

PRICE: \$300.00 (User Manual, \$25.00, credited toward Planner purchase). Michigan residents add 4% sales tax.

DEALERS: Your Inquiries Most Welcome

disk, serial terminal system, OS-65U v. 1.2 or later.

GANDER SOFTWARE. Ltd.

3223 Bross Road The Ponds Hastings. MI 49058 (616) 945-2821

FROM THE FOLKS WHO BROUGHT YOU: All This THERE IS MORE COMING SOON: Program Generator for TDS Proposal Planner Time and Billing A/R

[&]quot;It Flies

must re-write the entire record to the data file, wasting both time and space.

Next month, the INDEX (X) command.

MANUFACTURER'S CORNER

DB-65E

An Operating Environment

What is the DB-65E Operating Environment? What do we mean by an Operating Environment? These may be some of the questions that you have in the back of your mind. In this column, and subsequent columns, we hope to answer those questions and possibly give you a different approach to efficient programming.

Historically, the communication between man and the machine has taken two distinct paths. One path has been to provide a Machine Command Set that allows basic machine functions independent of any high - order language (e.g. UNIX, CP/M, etc.). The other path provides the execution of machine functions dependent upon a high - order language being resident (e.g. OS-65U, APPLESOFT, etc.).

The concept behind an Operating Environment is that an interface to a set of machine language commands has been defined and that this low-order language is resident at all times. However, there is no direct man-machine communication via this interface. This set of machine language commands must provide all of the basic Input and Output functions to any external device and additionally coordinate all inter-processor communication and semaphore handling. This method, we believe, maximizes the flexibility of programming options and still retains a level of discipline required for program portability.

This set of machine language commands within the DB-65E Operating Environment is called the IOCS, (Input and Output Control System.) A complete description of the IOCS is beyond the scope of this particular column but will be further explained in future columns. Before we leave IOCS, please be aware of the fact that the IOCS kernal is highly structured and conceptionally machine-independent.

The second part of the DB-65E

Operating Environment is a high - order language system. Initially, the DB-65E Operating Environment will be released with the DB-65E BASIC Language System. In the planning stages for future releases will be a "C" Language System, a FORTH Language System, a PASCAL Language System and a Relational Data Base Language System. Additionally, a complete Machine Language Development System containing a multi-level Macro Assembler and Linking Loader is planned for future release.

DB-65E BASIC LANGUAGE SYSTEM

THE DB-65E BASIC Language System is a BASIC Interpreter based on MicroSoft's M6502 BASIC Interpreter, but there have been many changes and extensions. However, even with these changes, the majority of programs developed for use with OSI'S OS-65U BASIC Language will execute properly under the DB-65E Operating Environment.

Some of the more significant changes and extensions of the DB-65E BASIC Language System are:

CREATE, REMOVE (Delete), RE-NAME, and SYSTEM

These functions are now "Reserved Words" within the BASIC Interpreter and may be used within the context of a program.

INPUT/OUTPUT FUNCTIONS

The Line EDITOR is functional and resident at all times without the loss of any other functions.

Two new commands have been added, "CRT" and "{}". The "{}", allow direct cursor positioning within a "PRINT", "INPUT", or "CRT" command line. The "CRT" command allows up to 16 different screen functions to be performed. Any or all of the 16 commands may be user-defined prior to the execution of a program.

A "USING" function has been added for the formatting of printed outputs, however, this "USING" function also allows the formatting of string variables for storage or later output.

Two more new commands, "STRIP" and "[]", have been added. The "[]" command allows a string variable to be left, right or center justified within a field of predefined characters. The "STRIP" command is complementary to the "[]" command in that the predefined characters are _re-moved from the left, right or all of the string variable.

Two additional new commands are "GET" and "INKEY". The "GET" command returns a predetermined number of characters, up to 132, from the Console device and assigns them to a string variable. The "INKEY" command polls the Console device for a character input and returns either the character or a "null" (no character present) in the string variable.

Another new command is the "DISK" function. This command allows the transfer of a block of data to, or from, the Disk and a memory location. The length of the block, disk address, memory address and direction of transfer are user-defined within the command syntax.

The new command "FILE" has been added to directly obtain such information as: Disk address of a <filename>, Length of a <filename>, Size of a Disk device, Size of the System where a <filename> is located, System Number where a <filename> is located, etc.

* Another new command "INSTR" has been added to determine the starting position of sub-string within a string variable.

The "FIND" command has been modified to allow "less than" or "greater than" searches of data and to compensate for field or record lengths as opposed to strictly linear searching of data matches. Additionally, a new command "DCARE" has been added to allow the user to define the "don't care" character to be ignored during the search function.

PROGRAM EXECUTION

A new command "SUBMIT" has been added to allow the use of a command file to control the sequence of programs to be executed.

Yet another new command is "ENCODE" which allows data to be encrypted as it is stored on the Disk device.

The "USR" command has been modified to allow the use of specific memory addressing as to the location of machine language routines.

The math functions, including

MAKE IT FAST

68010 CPU with 10 MHz clock...2–4 Mbyte no-wait-state RAM...fast DMA and Hard Disks...Virtual Memory with demand paging... an OS with Real Time response...

MAKE IT "CRASH PROOF"

Unmatched fault tolerance...RAM with memory Error Correction...Automatic Recovery after power failure...Safe Bit-Map disk allocation...

MAKE IT COMMUNICATE

Easily! With Real Time response ... communications boards with their own CPUs... Ethernet, "Cheapernet" and most other commonly used protocols.

MAKE IT FRIENDLY

Bundled software provides comprehensive menus and screen handlers...a fully integrated WP, spreadsheet, RDBM, and electronic mail system...

MAKE IT ADAPTABLE

4 free VME slots for expansion...extendable renus...options like graphics, extra RAM, specialized I/O and external disks...designed for systems integrators...



Loaded with new possibilities for handling office-oriented, industrial or scientific applications requiring Real Time response in a Unix environment.







Manufactured by ISOTRON, Inc., 140 Sherman Street, Fairfield, Connecticut 06430 (203) 255-7443 Telex 756436

all transcendental functions, have been increased to 16digit accuracy (19-digit internally) and the normal "binary round-off" error has been greatly minimized. Also, ALL of the math functions are resident at ALL times.

An added feature is that through the use of a "FLAG" command, Real-Time Interrupts may be serviced during the execution of a BASIC program.

IN SUMMARY

The DB-65E Operating Environment software, as with DBI's hardware, provides a natural evolutionary route to maximize performance and efficiency for their customers. The DB-65E Operating Environment with its BASIC Language System, containing a set of NEW and ENHANCED commands and the IOCS Machine Language interface, provides the basis for enhancing existing programs and new program development.

DBI's philosophy has been and will be to provide products that allow upgradeability and portability without necessitating the sacrificing or replacement of installed equipment and systems. The software described in this article is no exception. Even though this software presently executes only on a "6502" type of microprocessor, it will operate with the future???

By: Art Hughes



NEW COLOUR-PLUS: TOSIE has just finished its hardware compatible version of the original Color-Plus board. Our board is made with price, not quality in mind. It is not silk screened, it is not solder-resisted, it is not a top quality PCB, but it does work! If you are an experienced Hardware-Hacker and you have a working OSI 16-pin I/O bus on your machine, then this board is for you. TOSIE is a non-profit user's group, we are making this board available at cost as a service to the OSI user's community. Cost of the bare board is \$20 Cdn (\$20 US in the U.S.A., the dollar difference helps cover the added postage). If you require a top qulity product with easy installation we suggest you contact Generic Computer Products for an assembled version of the Color-Plus.

To order your OSI 16-pin Colour-Plus bare board or for more info write, - T.O.S.I.E., P. O. Box 29, Streetsville, Ontario, Canada L5M 2B7

* * * * *

FOR SALE: OSI C3 Computer System - Including 144K RAM, Dual Floppy Diskette, 36 MBYTE Hard Disk, 16 Slot Backplane, OS-65U Transaction Processing Basic, CA-10 Board, Configured for three (3) users; Two (2) Adam-3A+ Terminals; One (1) NEC 5515T Spinwriter Letter Quality Printer; Software. Everything in good condition \$4,500. Will consider selling components. Please call (412) 653-6400 between 8:00 AM and 5:00 PM Eastern Standard Time. Ask for Ray.

* * * * *

FOR SALE: OSI 250 Computer System - Including 104K RAM, 36 MBYTE Hard Disk, One (1) Floppy Disk Drive, OS-65U Transaction Processing Basic, CA-10 Board Configured for three (3) users; Three (3) Adam-5 Terminals; Software. Everything in excellent condition \$4,800. Please call (412) 653-6400 between 8:00 am and 5:00 PM Eastern Standard Time. Ask for Ray.

* * * * *

New CMOS static RAM: 26-Chips TOSHIBA 6116-3 150ns. \$1.50 each, S&H add \$1.00. David Adams, 407 Rollcrest, Midland, MI 48640.

* * * * *

FOR SALE - USED OSI (ISOTRON) EQUIPMENT: 11 - #520 16k RAM boards, 2 - #524 48k RAM boards - Revision A, 4 - #550 Multi-user boards, 8 - #510 CPU boards - Revision C, 2 -#522 Memory board - Revision / B, 2 - #527 Memory board, 1 -#505 CPU board 1 MHz, 1 - #535 Memory board 1 MHz. Other hardware available. Call Red River Computers, Inc. of Fargo, ND for "hot" prices. 1-701-281-0780. Ask for Fritz Wensel.

* * * * *

FOR SALE: OKIDATA 300 LPM dot matrix printer with 10 spare ribbons. High speed unit produces 4800 4 up E/W labels per hour, Centronics parallel input. Super rugged unit for heavy production runs. Factory service available. Checked at factory. Original price, over \$5,000. Selling price, \$1,200 plus COD truck delivery. Mr. Dell, (603) 924-9464.

* * * * *

FOR SALE: C3C-36 MB 3 User with 2 8" floppy drives, DMS Nucleus and Sort. All gold contacts. Looking for quick sale. Call Paul (301) 636-1400.

* * * * *

Send for free catalog, Aurora Software, 37 South Mitchell, Arlington Heights, IL 60005. Phone (312) 259-4071.

PEEK(65) INDEX FOR 1985

2MHZMENDANJAN22200SERIES UPDATEAUG13360SERIES UPDATEAUG1365816IN OSI, A PLANJAN1065D3.2FIND SEQUENTIALAPR2165D3.2VARIABLE LISTERFEB1665D3.2VARIABLE LISTERFEB1665D3.3DUG, FIXOCT865D3.3DURY COMTROLFEB865DEOCT EXPLAINEDOCT865DEDCT EXPLAINEDOCT865DEACTENDED INPUT 3.2/3.3FEB1465DFILE PACKINGJAN465DINFUT CONTROLMAR1465DEACTA FILES EXPLAINEDJUL265UERCRE TRAPPINGJUN1165UFILE HANDLINGJUL1265UFILES EXPLAINEDJUL1265UFROGRAMMER TIPSJUL265UREGRAMMER TIPSJUL265UREGRAMMER AIDSJUL265UREGRAMMER AIDSJUL265UVIDEOMAY1465UVIDEOMAY1265UVIDEOMAY1465UVIDEOMAY1265USERIES OSI - REVIEWAUG9ADET-100MAY122700SERIES OSI - REVIEWAUG9ADESSING, INDIRECTJUN3 <tr< th=""><th>TTEM</th><th>MO</th><th>PG</th></tr<>	TTEM	MO	PG
200SERIES UPDATEAUG 12300SERIES UPDATEAUG 1365816IN OSI, A PLANJAN 1065D 3.2FIND SEQUENTIALAPR 2165D 3.2VARIABLE LISTERFEB 1665D 3.2JAN 120COT 865D 3.3BUG, FIXOCT 865D 3.3DISK CONTROLFEB 865D 5.3JDISK CONTROLFEB 865D EOT EXPLAINEDOCT 865D EOT EXPLAINEDOCT 865D ENTENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D INFUT CONTROLMAR 1465D THE PACKINGJUN 2165U DATA FILES EXPLAINEDJUN 1265U ERROR TRAPPINGJUN 1265U FILE HANDLINGJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 1265U FROGRAMMENT FILESJUL 265U ROGRAMMENT FILESJUL 265U ROGRAMMENT AIDSJUL 265U ROGRAMMENT AIDSJUL 265U ROGRAMMENT AIDSJUL 365U VIDEOMAY 2165U VIDEOMAY 2165U VIDEOMAY 2165U VIDEOMAY 22700 SERIES OSI - REVIEWAUG 9ADDERSSING, INDIRECTJUN 3ADS-100MY 14ADS-200JUN 3ASSEMELY LANGUAGE CLASS VIIIFEB 13ASSEMELY LANGUAGE CLASS VIIIFEB 13ASSEMELY LANGUAGE CLASS VIIIFEB 13ASSEMELY LANGUAGE CLASS VIIIFEB 13ASSEMELY LANGUAGE CLA	2 MHZ MEM CONVERSION	JAN	22
360SERIES UPDATEAUG 1365816IN CSI, A PLANJAN 1065D 3.2FIND SEQUENTIALAPR 2165D 3.2VARIABLE LISTERFEB 1665D 3.3BUG, FIXOCT 865D 3.3DISK COMTROLFEB 865D 3.3DISK COMTROLFEB 865D 5.3SUGS COMTROLFEB 865D 5.3DISK COMTROLFEB 865D COT EXPLAINEDOCT 865D EXTENDED INPUT 3.2/3.3FEB 1465D THE PACKINGJAN 465D INPUT CONTROLMAR 1465D THE PACKINGJUN 2165U DEBUGGING TOCLSJUL 265U DEBUGGING TOCLSJUL 265U FRICE EXPLAINEDJUL 1265U FRIES EXPLAINEDJUL 1265U FRIES EXPLAINEDJUL 2065U FRIES EXPLAINEDJUL 2065U FRIES EXPLAINEDJUL 2065U FRIES EXPLAINEDJUL 2065U REGRAMMING AIDSJUN 1265U REGRAMMING AIDSJUN 1365U SELECTIVE SEARCH/FRINTMAY 2165U ROGRAMMING AIDSJUN 1365U ACIA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 22700 SERIES OSI - REVIEWAUG 22700 SERIES OSI - REVIEWAUG 22700 SERIES CSI - REVIEWAUG 22700 SERIES CSI - REVIEWAUG 22700 SERIES CSI - REVIEWAUG 23710 FROGMY 14ADS-100 FROGMY 14ADS-200JUN 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 1	200 SERIES UPDATE	AUG	12
65816 IN CSI, A PLANJAN 1065D 3.2 FIND SEQUENTIALAFR 2165D 3.2 VARIABLE LISTERFEB 1665D 3.3 BUG, FIXOCT 865D 3.3 BUG, FIXOCT 865D 3.3 DISK CONTROLFEB 865D 3.3 DISK CONTROLFEB 865D CATENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D FILE PACKINGJAN 465D FILE PACKINGJAN 465D CATENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJUN 2165U DENT CONTROLMAR 1465D THE PACKINGJUN 2165U DEBUGGING TOOLSJUL 2165U DEBUGGING TOOLSJUL 2165U FILE HANDLINGJUL 1265U FILE HANDLINGJUL 2265U FILE BANDENTJUL 2265U RCGRAMMENT FILESJUL 265U RCGRAMMING AIDSJUL 265U RCGRAMMING AIDSJUL 265U RCGRAMMING AIDSJUN 1265U RCGRAMMING AIDSJUN 1265U VIDEOMAY 2165U VIDEO TRIVERSJUN 1168000 CSI SYSTEMJUN 3ADS-100MAY 22700 SERIES OSI - REVIENMAR 22700 SERIES OSI - REVIENAUG 9ADDESSING, INDIRECTJUN 8ADS-100MY 14ADS-200MY 14ADS-100MY 14ADS-200MY 14ADS-200MY 14ASSEMELY LANG CLASS VIIIFEB 10ASSEMELY LANG CLASS VIIIFEB 10ASSEMELY LANG CLASS VIIIFEB 14BACKPLANNE PINOUTMAR 2 </td <td>300 SERIES UPDATE</td> <td>AUG</td> <td>13</td>	300 SERIES UPDATE	AUG	13
GLD 3.2 VARIABLE LISTERFEB 1665D 3.2 VARIABLE LISTERFEB 1665D 3.3 BUG, FIXOCT 865D 3.3 CURSCR CONTROLFEB 865D 3.3 DISK CONTROLFEB 865D 3.3 DISK CONTROLFEB 865D BOOT EXPLAINEDOCT 865D EXTENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D V3.3 SECOND DRIVE TRAPMAR 1465D V3.3 SECOND DRIVE TRAPMAR 1465D V3.3 SECOND DRIVE TRAPMAR 1465U DERGR TRAPPINGJUN 1265U ERGR TRAPPINGJUN 1265U FILE HANDLINGJUL 1265U FILE HANDLINGJUL 1265U FILE HANDLINGJUL 1265U FILE HANDLINGJUL 2065U FROGRAMMING AIDSJUL 2065U PROGRAMMING AIDSJUL 2065U RECETIVE SEARCH/PRINTMAY 1465U VIDEOMAY 2165U VIDEO DRIVERSJUN 1168000 OSI SYSTEMJUN 3ADS-100MAR 22700 SERIES OSI - REVIEWAUG 9ADDERSSING, INDIRECTJUN 3ADS-100MY. 14ADS-100MY. 14ADS-100MY. 14ADS-100FIX CASSETTEJUN 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 1	65D 2 2 FIND CROUENWIN	JAN	10
GED 3.2/3.3 DEV 1 FIXJAN 19GED 3.3 CURSOR ONTROLFEB 8GED 3.3 CURSOR CONTROLFEB 8GED 3.3 DISK CONTROLFEB 8GED 3.3 DISK CONTROLFEB 14GED FILE PACKINGJAN 4GED EXTENDED INPUT 3.2/3.3FEB 14GED FILE PACKINGJUN 12GED EXTENDED INPUT 3.2/3.3FEB 14GED TILE PACKINGJUN 21GED EXERCA TRAPPINGJUN 21GEU ERROR TRAPPINGJUL 12GEU FILE HANILINGJUL 12GEU FILES EXPLAINEDJUL 12GEU FROGRAMMER TIPSJUL 20GEU PROGRAMMER AIDSJUL 20GEU PROGRAMMING AIDSJUL 2GEU RESERVED WORDS LISTJUL 3GEU RESERVED WORDS LISTJUL 3GEU RESERVED WORDS LISTJUN 3GESU ACLA EXPLAINEDMAR 22700 SERIES OSI ~ REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100MY. 14ADS-200JUN 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BEGINNER'S CORNERMAY 1	65D 3.2 VARIABLE LISTER	APR FEB	16
65D 3.3 BUG, FIXOCT 865D 3.3 CURSOR CONTROLFEB 865D 3.3 DISK CONTROLFEB 865D BOOT EXPLAINEDOCT 865D EXTENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D FILE PACKINGJAN 465D VATA FILES EXPLAINEDMAR 1265U DATA FILES EXPLAINEDJUL 265U DATA FILES EXPLAINEDJUL 1265U DATA FILES EXPLAINEDJUL 1265U ERROR TRAPPINGJUL 1265U FILE HANLINGJUL 1265U FILE HANLINGJUL 1265U FILE HANLINGJUL 1265U FROGRAMMERT TIPSJUL 2065U PROGRAMMERT AIDSJUL 2065U PROGRAMMERT AIDSJUL 2065U PROGRAMMERT AIDSJUL 365U SELECTIVE SEARCH/PRINTMAY 1465U VIDEO IRIVERSJUL 365U SELECTIVE SEARCH/PRINTMAY 2165U VIDEO IRIVERSJUL 36800 OSI SYSTEMJUL 36800 OSI SYSTEMJUL 36800 OSI SYSTEMJUN 3ADS-100JUN 3ADS-100JUN 3ADS-100MY 14ADS-200JUN 3ASSEMBLY LANG CLASS IXMAR 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERCG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 10BEGINNER'S CORNER<	65D 3.2/3.3 DEV 1 FIX	JAN	19
65D 3.3 CURSOR CONTROLFEB 865D 3.3 DISK CONTROLFEB 865D BOOT EXPLAINEDOCT 865D EXTENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D FILE PACKINGJAN 465D INPUT CONTROLMAR 1465D V3.3 SECOND DRIVE TRAPMAR 2265U DATA FILES EXPLAINED - P1 DEC 1765U DEBUGGING TOOLSJUL 265U ERROR TRAPPINGJUN 2165U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 2065U FILES EXPLAINEDJUL 2065U FILES EXPLAINEDJUL 2065U PROGRAMMER TIPSJUL 2065U PROGRAMMER TIPSJUL 2065U PROGRAMMING AIDSJUL 2065U PROGRAMMING AIDSJUL 365U SEERVED WORDS LISTJUL 365U VIDEOMAY 2165U VIDEOMAY 21700 SERIES OSI - REVIEWAUG 32700 SERIES OSI - REVIEWMAR 22700 SERIES COSI - REVIEWJUN 3ASSEMBL	65D 3.3 BUG, FIX	OCT	8
65D 3.3 DISK CONTROLFEB 865D BOOT EXPLAINEDOCT 865D EXTENDED INPUT 3.2/3.3FEB 1465D FILE PACKINGJAN 465D INFUT CONTROLMAR 1465D V3.3 SECOND DRIVE TRAPMAR 2265U DATA FILES EXPLAINED - P1 DEC 1765U DEBUGGING TOOLSJUL 265U ERROR TRAPPINGJUN 1265U FILE HANDLINGJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 2065U FILES EXPLAINEDJUL 2065U FROGRAMMER TIPSJUL 2065U PROGRAMMER TIPSJUL 2065U PROGRAMMER TIPSJUL 2065U RESERVED WORDS LISTJUL 365U RESERVED WORDS LISTJUL 365U VIDEOMAY 2165U VIDEO IRIVERSJUN 1168000 OSI SYSTEMJUN 3ADS-100MAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100 PROGMY. 14ADS-100 PROGMY. 14ASSEMBLY LANG CLASS IXMAR 2ASSEMBLY LANG CLASS VIIIJAN 2ASSEMBLY LANG CLASS VIIIJAN 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC LINE # PRINT FROM M.L.NOV 23BASIC LINE # SORNERMAY 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNER CMLJAN 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC CARS VORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2	65D 3.3 CURSOR CONTROL	FEB	8
6 3D BOOT EXPLAINEDOCT 86 5D EXTENDED INPUT 3.2/3.3FEB 146 5D FILE PACKINGJAN 46 5D INPUT CONTROLMAR 146 5D V3.3 SECOND DRIVE TRAPMAR 226 5U DATA FILES EXPLAINED - P1 DEC 176 5U DEBUGGING TOOLSJUL 26 5U ERROR TRAPPINGJUN 126 5U FILE HANDLINGJUL 126 5U FILE SEXPLAINEDJUL 126 5U FILE HANDLINGJUL 126 5U FROGRAMMING AIDSJUL 26 5U PROGRAMMING AIDSJUL 26 5U PROGRAMMING AIDSJUL 26 5U PROGRAMMING AIDSJUL 36 5U PROGRAMMING AIDSJUL 36 5U PROGRAMMING AIDSJUL 36 5U VIDEO DRIVERSJUN 116 5U VIDEO DRIVERSJUN 116 5U VIDEO DRIVERSJUN 116 5U VIDEO DRIVERSJUN 36 50 ACLA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9700 SERIES CONTRERJUN 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10 </td <td>65D 3.3 DISK CONTROL</td> <td>FEB</td> <td>8</td>	65D 3.3 DISK CONTROL	FEB	8
GED FILE PACKINGJANGED FILE PACKINGJANGED INFUT CONTROLMARGED V3.3 SECOND DRIVE TRAPMARGED V3.3 SECOND DRIVE TRAPMARGED COLSJULGED RROR TRAPPINGJUNGED FILE HANDLINGJUNGED FILE HANDLINGJULGEU FRORS EXPLAINEDJULGEU FRORS EXPLAINEDJULGEU FILE HANDLINGJULGEU FILE HANDLINGJULGEU FRORAMMER TIPSJULGEU FRORAMMER TIPSJULGEU PROGRAMMING AIDSJULGEU PROGRAMMING AIDSJULGEU PROGRAMMING AIDSJULGEU PROGRAMMING AIDSJUNGEU VIDEO DRIVERSJUNGEU VIDEO DRIVERSJUNGEN VIDEO DRIVERSJUNGEN VIDEO DRIVERSJUNADS-100JUNADS-100JUNADS-100JUNADS-100JUNADS-100JUNASSEMBLY LABELS & ADDRESSESJUNASSEMBLY LABELS & ADDRESSESJUNASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIFEBBASIC FROG AT ANY ADDRESSMAYBASIC FROG AT ANY ADDRESSMAYBEGINNER'S CORNERMARBEGINNER'S CORNER CMLJANBEGINNER'S CORNER CMLJANBEGINNER'S CORNER CMLFEBBETA/65 DATA RECORDER, PLOTSEPBEGINNER'S CORNER CMLFEBBETA/65 DATA RECORNERMAYBEGINNER'S CORNER CMLFEB<	65D EVERNORD THORE 2 2/2 2	OCT	8
65DINPUT CONTROLMAR1465DV3.3SECONDDRIVE TRAPMAR2265UDATA FILESEXPLAINEDPIDEC1765UDEBUGGINGTOOLSJUL265UERRORTRAPPINGJUN1265UFRORSEXPLAINEDJUL1265UFILEHANDLINGJUL1265UFRORSAMMERJUSJUL265UPROGRAMMER TIPSJUL265UPROGRAMMING AIDSJUL265UPROGRAMMING AIDSJUL265UPROGRAMMING AIDSJUL365USEERVED WORDS LISTJUL365USEERVED WORDSJUN1365UVIDEODRIVERSJUN1465UVIDEODRIVERSJUN3650ACLAEXPLAINEDMAR22700SERIESOSI - REVIEWAUG9ADDRESSING, INDIRECTJUN3ADS-100JUN3ASS-100JUN3ASS-100JUN3ASSEMBLY LABELS & ADDRESSESJUNASSEMBLY LABELS & ADDRESSESJUNASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIFEBBASIC FROG AT ANY ADDRESSMAYBASIC FROG AT ANY ADDRESSMAYBEGI	65D FILE PACKING	TAN	14
65D V3.3 SECOND DRIVE TRAPMAR 2265U DATA FILES EXPLAINED - P1 DEC 1765U DEBUGGING TOOLSJUL 265U ERROR TRAPPINGJUN 2165U ERROR TRAPPINGJUN 1265U FILE HANDLINGJUL 1265U FILE HANDLINGJUL 1265U FILE SERLAINEDJUL 265U FROGRAMMER TIPSJUL 265U PROGRAMMING AIDSJUL 265U PROGRAMMING AIDSJUL 265U PROGRAMMING AIDSJUL 265U RESERVED WORDS LISTJUL 365U SELECTIVE SEARCH/PRINTMAY 1465U VIDEO IRIVERSJUL 365U SELECTIVE SEARCH/PRINTMAY 2165U VIDEO IRIVERSJUL 3685Ø ACLA EXPLAINEDMAR 2270Ø SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-10ØJUN 3ADS-10Ø FROGMY 14ADS-200JUN 3ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10 <t< td=""><td>65D INPUT CONTROL</td><td>MAR</td><td>14</td></t<>	65D INPUT CONTROL	MAR	14
65U DATA FILES EXPLAINED - P1 DEC 1765U DERGERGE TOOLSJUL 265U ERROR TRAPPINGJUN 2165U ERRORS EXPLAINEDJUN 1265U FILE HANDLINGJUL 1265U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 2065U FROGRAMMER TIPSJUL 2065U PROGRAMMER TIPSJUL 2065U PROGRAMMER ALDSJUL 265U PROGRAMMING ALDSJUL 265U PROGRAMMING ALDSJUL 265U RESERVED WORDS LISTJUL 365U SELECTIVE SEARCH/PRINTMAY 2165U VIDEO DRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACLA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100 PROGMY 14ADS-100 PROGJUN 3ASSIMELY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 11BASIC LINE # PRINT FROM M.L.NOV 23BASIC PROG AT ANY ADDRESSMAY 10BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BINARY CHAR. ENCODINGMAY 2BINARY CHAR. ENCODINGMAY 2CANON DRIVESJUL 21BIO-COMPATIBILITY P	65D V3.3 SECOND DRIVE TRAP	MAR	22
65UDEBUGGING TOOLSJUL65UERROR TRAPPINGJUN 2165UFIRCRS EXPLAINEDJUL 1265UFILE HANDLINGJUL 1265UFILES EXPLAINEDJUL 2065UPROGRAMMER TIPSJUL 2065UPROGRAMMER ADSJUL 2065UPROGRAMMING ADSJUL 265UPROGRAMMING ADSJUL 265UPROGRAMMING ADSJUL 265UPROGRAMMING ADSJUL 265UPROGRAMMING ADSJUL 265UVIDEO DRIVERSJUL 365U ACIA EXPLAINEDMAY 2165U VIDEO DRIVERSJUN 3685Ø ACIA EXPLAINEDMAR 2270Ø SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-10ØJUN 3ADS-10Ø PROGMY 14ADS-20ØJUN 3ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC CLINE # PRINT FROM M.L.NOV 23BASIC PROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 11BASIC LINE # SORNERMAY 12BEGINNER'S CORNERMAY 12BEGINNER'S CORNER MEDUG/TESTDEC 3BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 22BINARY CHAR. ENCODINGMAY 22	650 DATA FILES EXPLAINED - Pl	DEC	17
0.50 ERGER INAPPINGJUN 2165U ERGRS EXPLAINEDJUL 1265U FILE HANLLINGJUL 1265U FILE HANLLINGJUL 1265U FILES EXPLAINEDJUL 2065U PROGRAMMER TIPSJUL 2065U PROGRAMMING AIDSJUL 265U PROGRAMMING AIDSJUL 265U RESERVED WORDS LISTJUL 365U SELECTIVE SEARCH/PRINTMAY 1465U VIDEOMAY 2165U VIDEO DRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACIA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100JUN 3ADS-100 PROGMY 14ADS-100 PROGMY 14ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 11BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 11BEGINNER'S CORNERMAY 12BIGINNER'S CORNERMAY 12BIGINNER'S CORNER MEDJUN 22BIGINNER'S CORNER MEDJUN 23BEGINNER'S CORNER MEDJUN 23BEGINNER'S CORNER MEDJUN 21BIGINNER'S CORNER MEDJUN 21BINARY CHAR. ENCODINGMAY 22BINARY CHAR. ENCODINGMAY 22 <td>65U DEBUGGING TOOLS</td> <td>JUL</td> <td>2</td>	65U DEBUGGING TOOLS	JUL	2
GSU FILE HANDLINGJUL 12GSU FILE HANDLINGJUL 12GSU FILES EXPLAINEDJUL 12GSU FILES EXPLAINEDJUL 20GSU FROGRAMMER TIPSJUL 20GSU PROGRAMMER ADSJUN 12GSU PROGRAMMING ADSJUN 12GSU RESERVED WORDS LISTJUL 3GSU SELECTIVE SEARCH/PRINTMAY 21GSU VIDEOMAY 21GSU ACTA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100 FROGMY 14ADS-100 FROGMY 14ADS-200JUN 3ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERCG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAY 11BEGINNER'S CORNERMAY 12BASIC LINE # PRINT FROM M.L.NOV 23BASIC SORNER CMLJAN 2BEGINNER'S CORNERMAY 12BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2CA	6511 FRRORS EXPLAINED	JUN	12
65U FILES EXPLAINEDJUL 1265U FILES EXPLAINEDJUL 265U INDIRECT FILESJUL 265U PROGRAMMER TIPSJUL 265U PROGRAMMING AIDSJUL 265U RESERVED WORDS LISTJUL 365U RESERVED WORDS LISTJUL 365U VIDEOMAY 2165U VIDEO IRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACIA EXPLAINEDMAR 22700 SERLES OSI - REVIEWACG 9ADDRESSING, INDIRECTJUN 3ADS-100 PROGMY 14ADS-200JUN 3ADS-100 PROGMY 14ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LABELS & ADDRESSESFEB 16ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIMAR 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 11BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINERMAY 22CANON DRIVESJUN 23CANON DRIVESJUN 22CANON DRIVESJUN 22CANON DRIVESJUN 22CANON DRIVESJUN 22CANON DRIVESJUL 21CANON DRIVESJUN 22CANON DRIVESJUN 22CANON DRIVESJUN 22 <tr< td=""><td>650 FILE HANDLING</td><td>JUL</td><td>12</td></tr<>	650 FILE HANDLING	JUL	12
65UINDIRECT FILESJUL265UPROGRAMMER TIPSJUL2065UPROGRAMMER TIPSJUL2065UPROGRAMMING AIDSJUL265UPROGRAMMING AIDSJUL265USESERVED WORDS LISTJUL365USELECTIVE SEARCH/PRINTMAY 2165UVIDEOMAY 2165UVIDEOMAY 2165UVIDEOMAY 2165UVIDEOMAY 2165UVIDEOMAY 2165UVIDEOMAY 2165UACLA EXPLAINEDMAR 22700SERLES CSI - REVIEWAUG 9ADS-100POGJUN 3ADS-100POGJUN 3ADS-100POGJUN 3ADS-100POGJUN 3ASS-100POGJUN 3ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERCG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 11BEGINNER'S CORNER CMLJAN 2BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBLLITY PROG.MAY 19BOCT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CANON DRIVES<	65U FILES EXPLAINED	JUL	12
65U PROGRAMMER TIPSJUL 2065U PROGRAMMER ALDSJUN 1265U PROGRAMMING ALDSJUL 265U RESERVED WORDS LISTJUL 365U SELECTIVE SEARCH/PRINTMAY 2165U VIDEOMAY 2165U VIDEODRIVERSJUN 116800 OSI SYSTEMJUL 36850 ACLA EXPLAINEDMAR 22700 SERIES OSI - REVIENADG 9ADDRESSING, INDIRECTJUN 3ADS-100 PROGMY 14ADS-200JUN 3ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC ERG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAY 10BEGINNER'S CORNERMAY 10BEGINNER'S CORNER CMLJAN 2BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BUNARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOCT 65D EXPLAINERMAY 19BOCT 65D EXPLAINERMAY 23CIP - ADD IBM FLOPPIESAFR 4CIP - ADD IBM FLOPPIESAFR 4CIP - ADD IBM FLOPPIESAFR 4CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 23CANON DRIVESJUL 23CANON DRIVESJUL 23CANON DRIVESJUL 23CANON DRIVESJUL 23CANON DRIVES	650 INDIRECT FILES	JUL	2
650PROGRAMMING AIDSJUN 12650PROGRAMMING AIDSJUL 2650RESERVED WORDS LISTJUL 3650SELECTIVE SEARCH/PRINTMAY 14650VIDEOMAY 21650VIDEODRIVERSJUN 1168000CSI SYSTEMJUL 36850ACIA EXPLAINEDMAR 22700SERIES OSI - REVIEMADG700SERIES OSI - REVIEMJUN 3700SERIES OSI - REVIEMJUN 3700SERIES & ADDRESSESFEB 13700ASSEMBLY LANG CLASS IXMAR 3700ASSEMBLY LANG CLASS VIIIFEB 10700ASSEMBLY LANG CLASS VIIIFEB 10700RAR10JUN 23701 <t< td=""><td>65U PROGRAMMER TIPS</td><td>JUL</td><td>2Ø</td></t<>	65U PROGRAMMER TIPS	JUL	2Ø
COUNTPROGRAMMING AIDSJUL65URESERVED WORDS LISTJUL65URESERVED WORDS LISTJUL65UVIDEOMAY 2165UVIDEODRIVERS65UVIDEODRIVERS65UVIDEODRIVERS700SERIES OSI - REVIEWADG700SERIES & ADDRESSJUN700SERIES & ADDRESSESFEB700SEMBLY LABELS & ADDRESSESFEB700SEMBLY LANG CLASS VIIIFEB700ASSEMBLY LANG CLASSMAR700SCORNERMAR700RINER'S CORNERMAR700RINER'S C	65U PROGRAMMING AIDS	JUN	12
GSU RESERVED WORDS LISTJUL65U SELECTIVE SEARCH/PRINTMAY 1465U VIDEOMAY 2165U VIDEO DRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACLA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100JUN 3ADS-100 PROGMY 14ADS-200JUN 3ASM - SHARED POINTERSJUL 1ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASED FROG AT ANY ADDRESSMAY 10BGINNER'S CORNERMAR 11BEGINNER'S CORNERMAY 10BGINNER'S CORNERMAY 10BGINNER'S CORNER CMLJAN 2BGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOT 65D EXTLAINEDOCT 8BUBBLE SORT/MERE, MODJUN 16BUSI-CALCMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVESJUL 21CANON DRIVESJUL 22 </td <td>65U PROGRAMMING ALLS</td> <td>JUL</td> <td>2</td>	65U PROGRAMMING ALLS	JUL	2
GSU SIZE ANDSERVENT FAILMAY 216SU VIDEODRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACLA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 3ADS-100JUN 3ADS-100 PROGMY 14ADS-200JUN 3ASM - SHARED POINTERSJUN 8ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC FROG AT ANY ADDRESSMAY 10BGINNER'S CORNERMAR 11BGINNER'S CORNERMAY 6BGINNER'S CORNERMAY 6BGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOT 65D EXTLAINEDOCT 8BUBBLE SORT/MERGE, MODJUL 12BIO-COMPATIBILITY PROG.MAY 22CLP - ADD IBM FLOPPIESAFR 4CLP AT 2 MHZ FIXJAN 22CLP BOOT FROBLEMAUG 20CLP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVES <t< td=""><td>6511 SELEVED WORLS LIST</td><td>JUL</td><td>3</td></t<>	6511 SELEVED WORLS LIST	JUL	3
65U VIDEO DRIVERSJUN 1168000 OSI SYSTEMJUL 36850 ACIA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 8ADS-100JUN 3ADS-100 PROGMY 14ADS-200JUN 3ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10BASIC LINE # PRINT FROM M.L.NUV 23BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAY 16BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CANON DRIVESJUL 21CANON D	650 VIDEO	MAY	21
68000 OSI SYSTEMJUL 36850 ACIA EXPLAINEDMAR 22700 SERIES OSI - REVIEWAUG 9ADDRESSING, INDIRECTJUN 8ADS-100JUN 3ADS-100JUN 3ASS-100 PROGMY 14ADS-200JUN 3ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 16ASSEMBLY LANG CLASS VIIIFEB 16ASSEMBLY LANG CLASS VIIIPAR 2BASIC FROG AT ANY ADDRESSMAY 10BASIC FROG AT ANY ADDRESSMAY 10BGINNER'S CORNERMAR 11BGINNER'S CORNERMAY 16BGINNER'S CORNERMAY 16BGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUL 21BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUL 12BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUL 21CIP - ADD IBM FLOPPIESAFR 4CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESJUL 21<	65U VIDEO DRIVERS	JUN	īī
685Ø ACIA EXPLAINEDMAR 2270Ø SERIES OSI - REVIENADGADDRESSING, INDIRECTJUN 8ADS-10ØJUN 3ADS-10Ø PROGMY. 14ADS-20ØJUN 3ASM - SHARED POINTERSJUN 8ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIJAN 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BEG INNER'S CORNERMAY 16BEG INNER'S CORNERMAY 16BEG INNER'S CORNERMAY 16BEG INNER'S CORNER CMLJAN 2BEG INNER'S CORNER CMLJAN 2BEG INNER'S CORNER CMLJAN 2BEG INNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 19BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21 <tr< td=""><td>68000 OSI SYSTEM</td><td>JUL</td><td>3</td></tr<>	68000 OSI SYSTEM	JUL	3
700 SERIES OSI - REVIENADGADDRESSING, INDIRECTJUNADS-100JUNADS-100JUNADS-200JUNASM - SHARED POINTERSJUNASSEMBLY LANC CLASSJUNASSEMBLY LANG CLASS IXMARASSEMBLY LANG CLASS VIIIJEEASSEMBLY LANG CLASS VIIIJANBASSEMBLY CLASS CLASSMARBASSEMBLY CLASSMARBASSEMBLY CLASSMARBASSEMBLY CLASSMARBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBASSEMBLY CLASSMAYBEGINNER'S CORNERMAYBEGINNER'S CORNER CMLJANBEGINNER'S CORNER CMLJANBESTAL65DATA RECORDER, PLOTBESTAL65DATA RECORDER, MODBUNARY SEARCH - FILEJULBUNARY SEARCH - FILEJULBUSI-CALCMAYCLP AT 2 MHZ FIXJANCLP AT 2 MHZ FIXJANCANON DRIVESJULCANON DRIVESJULCANON DRIVES <t< td=""><td>6850 ACLA EXPLAINED</td><td>MAR</td><td>22</td></t<>	6850 ACLA EXPLAINED	MAR	22
ADLRESSING, INDIRECT JUN 8 ADS-100 JUN 3 ADS-100 JUN 3 ADS-200 JUN 3 ASM-SHARED POINTERS JUN 8 ASM/ED BUG & FIX CASSETTE JUL 12 ASSEMBLY LANG CLASS IX MAR 3 ASSEMBLY LANG CLASS VIII FEB 13 ASSEMBLY LANG CLASS VIII JAN 2 BASSEMBLY LANG CLASS VIII JAN 2 BASIC LINE # PRINT FROM M.L. NOV 23 BASIC PROG AT ANY ADDRESS MAR 11 BGINNER'S CORNER MAR 11 BGINNER'S CORNER MAR 11 BGINNER'S CORNER MAY 6 BEGINNER'S CORNER OML FEB 8 BETA/65 DATA RECORDER, PLOT SEP 14 BINARY SEARCH - FILE JUL 21 BIO-COMPATIBILITY PROG. MAY 19 BOOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 22 CIP ADD IBM FLOPPIES APR 4 CIP ADD IBM FLOPPIES	700 SERIES OSI - REVIEW	AUG	9
ADS-100DONJADS-100PROGMY14ADS-200JUN3ASM - SHARED POINTERSJUNASSEMBLY LABELS & ADDRESSESFEBASSEMBLY LANG CLASS IXMARASSEMBLY LANG CLASS VIIIFEBASSEMBLY LANG CLASS VIIIJANASSEMBLY LANG CLASS VIIIJANASSEMBLY LANG CLASS VIIIJANASSEMBLY LANG CLASS VIIIJANBASSEMBLY LANG CLASS VIIIJANBASSEMBLY LANG CLASS VIIIJANBASSEMBLY LANG CLASS VIIIJANBASIC FIRG AT ANY ADDRESSMAYBASIC FROG AT ANY ADDRESSMAYBASIC PROG AT ANY ADDRESSMAYBEGINNER'S CORNERMARBEGINNER'S CORNERMAYBEGINNER'S CORNERMAYBEGINNER'S CORNER OMLFEBBEGINNER'S CORNER OMLJANBEGINNER'S CORNER OMLFEBBETA/65 DATA RECORDER, PLOTSEPBINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAYBUSI-CALCMAYCIP - ADD IBM FLOPPIESAPRCIP - ADD IBM FLOPPIESAPRCLP AT 2 MHZ FIXJANCLP BOOT FROBLEMADGCONN DRIVESJUL 21CANON DRIVES <td< td=""><td>ADDRESSING, INDIRECT</td><td>JUN</td><td>2</td></td<>	ADDRESSING, INDIRECT	JUN	2
ADS-200JUN 3ASM - SHARED POINTERSJUN 8ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS IXMAR 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIJAN 2BACKPLANE PINCUTMAR 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BGINNER'S CORNERMAR 11BGINNER'S CORNERMAR 11BGINNER'S CORNERMAY 6BGINNER'S CORNERMAY 6BGINNER'S CORNER OMLFEB 8BEGINNER'S CORNER OMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CIP - ADD IBM FLOPPIESAPR 4CIP AT 2 MHZ FIXJAN 22CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21	ADS-100 PROG	MY.	14
ASM - SHARED POINTERSJUN 8ASM - SHARED POINTERSJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS IXMAR 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIJAN 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAR 11BGINNER'S CORNERMAR 11BGINNER'S CORNERMAY 10BGINNER'S CORNERMAY 16BGINNER'S CORNERMAY 6BGINNER'S CORNER OMLJAN 2BGINNER'S CORNER OMLJAN 2BGINNER'S CORNER OMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CIP - ADD IBM FLOPPIESAPR 4CIP AT 2 MHZ FIXJAN 22CANON DRIVESJUL 21CANON DRIVESJUC 22 <td>ADS-200</td> <td>JUN</td> <td>3</td>	ADS-200	JUN	3
ASM/ED BUG & FIX CASSETTEJUL 12ASSEMBLY LABELS & ADDRESSESFEB 13ASSEMBLY LANG CLASS IXMAR 3ASSEMBLY LANG CLASS VIIIFEB 10ASSEMBLY LANG CLASS VIIIJAN 2BASDEMBLY LANG CLASS VIIIJAN 2BASDEMBLY LANG CLASS VIIIJAN 2BASDEMBLY LANG CLASS VIIIJAN 2BASDEMBLY LANGUAGE CLASS VIIIJAN 2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAY 6BEGINNER'S CORNER OMLJAN 2BEGINNER'S CORNER OMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CIP - ADD IBM FLOPPIESAPR 4CIP AT 2 MHZ FIXJAN 22CIP BOOT PROBLEMADG 20CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21 </td <td>ASM - SHARED POINTERS</td> <td>JUN</td> <td>8</td>	ASM - SHARED POINTERS	JUN	8
ASSEMBLY LABELS & ADDRESSES FBB 13 ASSEMBLY LANG CLASS IX MAR 3 ASSEMBLY LANG CLASS VIII FBB 10 BASIC LINE # PRINT FROM M.L. NOV 23 BASIC PROG AT ANY ADDRESS MAR 11 BEGINNER'S CORNER MAR 11 BEGINNER'S CORNER MAR 12 BEGINNER'S CORNER MAY 6 BEGINNER'S CORNER MEDUG/TEST DEC 3 BEGINNER'S CORNER OML FEB 8 BETA/65 DATA RECORDER, PLOT SEP 14 BINARY CHAR. ENCODING MAY 19 BOOT 65D EXPLAINED OCT 8 BUBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 22 CIP - ADD IBM FLOPPIES APR 4 CIP AT 2 MHZ FIX JAN 22 CIP BOOT FROBLEM AUG 20 CIP SPEAD SHEET MAY 22 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FIN	ASM/ED BUG & FIX CASSETTE	JUL	12
ASSEMBLY LANG CLASS VIII FEB 10 ASSEMBLY LANGUAGE CLASS VIII FEB 10 BACKPLANE PINCUT MAR 2 BASIC LINE # PRINT FROM M.L. NOV 23 BASIC FROG AT ANY ADDRESS MAY 10 BEGINNER'S CORNER MAR 11 BEGINNER'S CORNER MAR 11 BEGINNER'S CORNER MAR 11 BEGINNER'S CORNER MAY 10 BEGINNER'S CORNER MAY 10 BETA/65 DATA RECORDER, PLOT SEP 14 BINARY CHAR. ENCODING MAY 2 BINARY SEARCH - FILE JUL 21 BIO-COMPATIBILITY PROG. MAY 19 BOOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 CLP - ADD IBM FLOPPIES APR 4 CLP AT 2 MHZ FIX JAN 22 CLP BOOT FROBLEM ADG 20 CLP SPREAD SHEET MAY 22 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES AUG 22 C	ASSEMBLY LABELS & ADDRESSES	FEB MAD	73
ASSEMBLY LANGUAGE CLASS VIIJAN2BACKPLANE PINCUTMAR2BASIC LINE # PRINT FROM M.L.NOV23BASIC FROG AT ANY ADDRESSMAY10BEGINNER'S CORNERMAR11BEGINNER'S CORNERMAR11BEGINNER'S CORNERMAR11BEGINNER'S CORNERMAR11BEGINNER'S CORNERMAR16BEGINNER'S CORNERMAR16BEGINNER'S CORNER OMLJAN2BEGINNER'S CORNER OMLFEBBETA/65 DATA RECORDER, PLOTSEPBINARY CHAR. ENCODINGMAYBINARY SEARCH - FILEJULBUOT 65D EXPLAINEDOCTBUSI-CALCMAYCIP - ADD IBM FLOPPIESAPRCIP AT 2 MHZ FIXJANCLP BOOT FROBLEMAUGCONN DRIVESJULCANON DRIVESJULCANON DRIVESJULCANON DRIVESJULCANON DRIVESAUGCHANON DRIVESAUGCANON DRIVESAUGCANON DRIVESAUGCOMON/65D LINKEROCTCUADACTER CENEMATORWITCANON DRIVESAUGCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCANON DRIVESAUGCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCHANCY 65D LINKEROCTCHANCY 65D LINKEROCT <td>ASSEMBLY LANG CLASS VIII</td> <td>FFR</td> <td>10</td>	ASSEMBLY LANG CLASS VIII	FFR	10
BACKPLANE PINCUTMAR2BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 16BEGINNER'S CORNERAFR 16BEGINNER'S CORNERMAY 10BEGINNER'S CORNERMAR 16BEGINNER'S CORNERMAR 16BEGINNER'S CORNERMAY 10BEGINNER'S CORNER MELDUG/TESTDEC 3BEGINNER'S CORNER OMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CIP - ADD IBM FLOPPIESAPR 4CIP AT 2 MHZ FIXJAN 22CIP SPEAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUC 22CANON DRIVESJUC 21CHANCYFER CENERATIONTUN 11 <td>ASSEMBLY LANGUAGE CLASS VII</td> <td>JAN</td> <td>2</td>	ASSEMBLY LANGUAGE CLASS VII	JAN	2
BASIC LINE # PRINT FROM M.L.NOV 23BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BEGINNER'S CORNERMAR 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNER OMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJEN 2BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXFLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 22CIP - ADD IBM FLOPPIESAPR 4CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21	BACKPLANE PINOUT	MAR	2
BASIC FROG AT ANY ADDRESSMAY 10BEGINNER'S CORNERMAR 11BCINNER'S CORNERAPR 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNERMAY 16BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODDINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXFLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 23CIP - ADD IBM FLOPPIESAPR 4CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVES <td>BASIC LINE # PRINT FROM M.L.</td> <td>NOV</td> <td>23</td>	BASIC LINE # PRINT FROM M.L.	NOV	23
DEG INNER'S CORNERMAR 11BEG INNER'S CORNERAPR 16BEG INNER'S CORNERMAY 6BEG INNER'S CORNER DEBUG/TESTDEC 3BEG INNER'S CORNER CMLJAN 2BEG INNER'S CORNER CMLJAN 2BEGINNER'S CORNER CMLFEB 8BETA/65 DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 23CIP - ADD IBM FLOPPIESAPR 4CIP AT 2 MHZ FIXJAN 22CIP BOOT FROBLEMADG 20CIP SPREAD SHEETMAY 22CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESAUG 22CANON DRIVESJUL 21CANON DRIVESAUG 22CANON DRIVESAUG 22CANON DRIVESAUG 22CANON DRIVESAUG 22CANON DRIVESAUG 22CANON DRIVESAUG 21CANON DRIVESAUG 22CANON DRIVESAUG 22CHARAVER CORDERATORAUG 23	BASIC PROG AT ANY ADDRESS	MAY	10
DEG INNER'S CORNER MAY 10 BEG INNER'S CORNER DEBC INNER'S CORNER DEBUG/TEST DEC 3 BEG INNER'S CORNER CML JAN 2 BEG INNER'S CORNER CML JAN 2 BEG INNER'S CORNER CML FEB 8 BETA/65 DATA RECORDER, PLOT SEP 14 BINARY CHAR. ENCODING MAY 19 BOT 65D EXFLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 CIP - ADD IBM FLOPPIES AFR 4 CIP SPREAD SHEET MAY 20 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES JUL 22 CANON DRIVES AUG 21 CANON DRIVES AUG 22 CANON DRIVES AUG 22 <td>BEGINNER'S CORNER BEGINNER'S CORNER</td> <td>MAR</td> <td>16</td>	BEGINNER'S CORNER BEGINNER'S CORNER	MAR	16
BBGINNER'S CORNER DEBUG/TEST DEC 3 BEGINNER'S CORNER OML JAN 2 BEGINNER'S CORNER OML FEB 8 BETA/65 DATA RECORDER, PLOT SEP 14 BETA/65 DATA RECORDER, PLOT SEP 14 BINARY CHAR. ENCODING MAY 2 BINARY SEARCH - FILE JUL 21 BIO-COMPATIBILITY PROG. MAY 19 BOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 CLP - ADD IBM FLOPPIES AFR 4 CLP AT 2 MHZ FIX JAN 22 20 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FINAL NOV 22 CHABACY65D LINKE	BEGINNER'S CORNER	MAY	6
BEG INNER'S CORNER OMLJAN2BEG INNER'S CORNER OMLFEB8BETA/65 DATA RECORDER, PLOTSEPBINARY CHAR. ENCODINGMAYBINARY SEARCH - FILEJULBIO-COMPATIBILITY PROG.MAYBUBBLE SORT/MERGE, MODJUNBUSI-CALCMAYCLP - ADD IBM FLOPPIESAFRCLP SPREAD SHEETMAYCANON DRIVESJULCANON DRIVESJULCHADACTER CENERATORNOVCANON DRIVESJULCANON DRIVESJULCANON DRIVESJULCANON DRIVESJULCHADACTER CENERATORNOVCANON DRIVESJULCHADACTER CENERATORNOV	BEGINNER'S CORNER DEBUG/TEST	DEC	Ĵ.
BEG INNER'S CORNER CMLFEB8BETA/65 DATA RECORDER, PLOTSEPBINARY CHAR. ENCODINGMAYBINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAYBODE 65D EXPLAINEDOCTBUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAYCLP - ADD IBM FLOPPIESAFRCLP BOOT FROBLEMAUGCIP SPREAD SHEETMAYCANON DRIVESJUL 21CANON DRIVESJUL 22CANON DRIVESJUL 23CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVESAUGCERMON/65D LINKEROCTOUADACTER CENEMATORVIN	BEGINNER'S CORNER OML	JAN	2
BETAY OS DATA RECORDER, PLOTSEP 14BINARY CHAR. ENCODINGMAY 2BINARY SEARCH - FILEJUL 21BIO-COMPATIBILITY PROG.MAY 19BOOT 65D EXPLAINEDOCT 8BUBBLE SORT/MERGE, MODJUN 16BUSI-CALCMAY 23ClP - ADD IBM FLOPPIESAFR 4ClP AT 2 MHZ FIXJAN 22ClP BOOT FROBLEMAUG 20CLP STREAD SHEETMAY 23CANON DRIVESJUL 21CANON DRIVESJUL 21CANON DRIVES FINALNOV 22CANON DRIVES FINALNOV 22CBGMON/65D LINKEROCT 19CHARACTER CEMERATORNT 11	BEGINNER'S CORNER OML	FEB	.8
BINARI CHAR, ENCODING PAY BINARY SEARCH - FILE JUL 21 BIO-COMPATIBILITY PROG. MAY 19 BOOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 ClP - ADD IBM FLOPPIES AFR 4 ClP AT 2 MHZ FIX JAN 22 ClP BOOT FROBLEM AUG 20 CLP SPREAD SHEET MAY 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 21 CBGMON/65D LINKER OCT 19	BETA/ 65 DATA RECORDER, PLOT	SEP	14
BIO-COMPATIBILITY PROG. MAY 19 BOOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 ClP - ADD IBM FLOPPIES AFR 4 ClP AT 2 MHZ FIX JAN 22 ClP BOOT FROBLEM AUG 20 CLP SPREAD SHEET MAY 23 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES AUG 22 CANON DRIVES TUL 21 CANON DRIVES MOV 22 CANON DRIVES FINAL NOV 21 CBRON/GED LINKER OCT 19	BINARI CHAR. ENCODING BINARY SEARCH - FILE		21
BOOT 65D EXPLAINED OCT 8 BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 ClP - ADD IBM FLOPPIES AFR 4 ClP AT 2 MHZ FIX JAN 22 ClP BOOT FROBLEM AUG 20 CLP STREAD SHEET MAY 23 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 21 CBRON/65D LINKER OCT 19 CHABAC/65D LINKER OTT 19	BIO-COMPATIBILITY PROG.	MAY	19
BUBBLE SORT/MERGE, MOD JUN 16 BUSI-CALC MAY 23 ClP - ADD IBM FLOPPIES AFR 4 ClP AT 2 MHZ FIX JAN 22 ClP BOOT PROBLEM AUG 20 CLP SPREAD SHEET MAY 23 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 21 CANON CRIVES FINAL NOV 21 CBGMON/65D LINKER OCI 19 CHARACTER CORDATION UN 11	BOOT 65D EXPLAINED	OCT	8
BUSI-CALC MAY 23 ClP - ADD IBM FLOPPIES AFR 4 ClP AT 2 MHZ FIX JAN 22 ClP BOOT FROBLEM AUG 20 CLP SFREAD SHEET MAY 23 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 22 CANON DRIVES FINAL NOV 21 CBGMON/65D LINKER OCI 19	BUBBLE SORT/MERGE, MOD	JUN	16
CLP - ADD IBM FLOPPIES AFR 4 CLP AT 2 MHZ FIX JAN 22 CLP BOOT FROBLEM AUG 20 CLP SFREAD SHEET MAY 22 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CBMON/65D LINKER OCI 19 CHAPACYER CENERATOR NOV 11	BUSI-CALC	MAY	23
CIP AI 2 MIA FIX JAN 22 CIP BOOT PROBLEM AUG 20 CIP SPREAD SHEET MAY 22 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CEGMON/65D LINKER OCI 19 CHAPACYER CEMERATOR NOV 11	CIP - ADD IBM FLOPPIES	APR	4
CIP SPREAD SHEET MAY 22 CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CBGMON/65D LINKER OCT 19 CHARACYGED CENERATOD JUN 11	CIP BOOT PROBLEM	JAN	24
CANON DRIVES JUN 23 CANON DRIVES JUL 21 CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CBGMON/65D LINKER OCT 19 CHARACTER CENERATOR JUN 11	Clp spread sheet	MAY	22
CANON DRIVES JUL 21 CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CBGMON/65D LINKER OCI 19 CHARACTER CENERATION NUN 11	CANON DRIVES	JUN	23
CANON DRIVES AUG 22 CANON DRIVES FINAL NOV 22 CBGMON/65D LINKER OCT 19 CHARACTER CENTERATOR UT 11	CANON DRIVES	JUL	21
CANON DRIVES FINAL NOV 22 CEGMON/65D LINKER OCT 19 CHADACTER CENEDATOD TITL	CANON DRIVES	AUG	22
	CANON DRIVES FINAL	NOV	22
	CHARACTER GENERATOR	JUN	11





тm 83181



-	-	

	COLOR PLUS REVISITED	MAR	13
	COMPUSERVE OSI-SIG - REVIEW	FEB	15
	COPYFI BUG & FIX	JUL	20
	CURSOR CONTROL 65D 3.3	FEB	23 8
	DABUG 3J, REVIEW	AUG	14
	DABUG III	JUN	17
	DATA FILES EXPLAINED 650 - PI DATA RECORDER, PLOT BETA/65	SEP	14
	DATE IN WORDS, 65U	AUG	īø
	DAY OF WEEK IN WORDS, 65U	AUG	10
	DEBUGGING TOOLS 650	JUL	2
	DEPTH CHG TAPE TO DISK CONV.	AUG	19
	DEV 1 FIX 65D 3.2/3.3	JAN	19
	DIR EXPLAINED	NOV	2
	DISASSEMBLIES	APR	22
	DISK CONTROL 65D 3.3	FEB	8
	DMS STRING PADDING	APR	19
	DOLLARS IN WORDS	AUG	12
	EDUCATION - COMPUTER	JUN	18
	ERRATA - PIACOM	JAN	23
	ERROR CODES EXPLAINED 650	JUN	12
	ERROR TRAPPING U	JUN	21
	EXPAND FD CAPACITY 18%	MAY	18
	EXTENDED INPUT 65D 3.2/3.3	FEB	14
	FD CAPACITY EXPANDED 18%	MAY	18
	FILE HANDLING 65U	JUL	12
	FIX - 65D 3.3 BUG	OCT	8
	FIX - ASM/ED BUG, CASSETTE	JUL	12
	FIX - CONVERT TO 2 MHZ, CIP	JAN JUL	20
	FIX - DEV 1 65D 3.2/3.3	JAN	1 9
	FIX - D&N CONTROLLER	MAY	22
	FIX - ERROR 9 FIX - MINOS BUG	MAY	18
	FIX - MODEM C4P	FEB	22
	FIX - VIDEO DRIVER 65U	AUG	23
	FIX - WP0502 VI.2 FLAGS 6511 FXPLATNED	MAR	10
	FLOPPY DRIVE TURN OFF	MAY	4
	FLOPPY DRIVE TURN OFF	JUL	16
	FLOPPY DRIVE TURN OFF	AUG	13
	FLOPPY DRIVE TURN OFF	NOV	14 7
	FLOPPY DRIVE TURN OFF - BASIC	MAY	17
	FLOPPY DRIVE TURN OFF - DUAL	SEP	20
	FORTH - VIDEO DRIVER BUG	AUG	23
	FORTH, MORE PART II	AUG	17
	FORTH, REVIEW	JUL	8
	GAME PROGRAMMING GASP	DEC	0 8
	GET A KEY 65D	JUN	6
	GLOBAL VS LOCAL VARIABLES	DEC	3
	GRAFIX SEB-1 INSTALLATION	OCT	6
	GRAPHICS, PRINTING 65D	JUN	8
	GREAT LANGUAGE DEBATE	SEP	17
	HARDWARE, BASIC OVERVIEW	MAR	2
	HEAP SORT	JUN	16
	HI RES GRAPHICS	OCT	16
	HI/LO BYTE CALCULATING	JUN	12
	HOOKS - ADD A MOD	MAY	17
	HOSPITAL INSTALLATION DRT	SEP MAV	3 51
	IBM FLOPPIES ON OSI	APR	4
	IBM PC BEATEN BY OSI	SEP	7
	INDIRECT ADDRESSING	JUN	8 2
-	INPUT CONTROL 65D 3.3	MAR	14
~	INFUT DEFAULT 65D 3.3	OCT	14
-	KEYBOARD DECODE	JUN	17
	KEYWORD OPERATION TIPS	MAY FFR	25
	LABELS & ADDRESSES, ASSEMBLY	FEB	13

LABELS & ADDRESSES ASSEMBLY	
	APR 22
LANGUAGE DEBATE	SEP 17
LINE NO. LOOK-UP TABLE	JAN 8
LOCAL VS GLOBAL VARIABLES	DEC 3
LOOK-ANCLES SATELLITE	TAN 22
LOOK-ANGLES SATELLITE	
M. L. PROGRAMMING GUIDE 650	JUN Z
M. L. PROGRAMMING GUIDE 650	SEP 2
MACHINE LANGUAGE MAPPING	FEB 2
MACTC SOLIARES	SED 7
MACIC COUNDS	MAD 20
MAGIC SQUARS	MAR 20
MAPPING M.L. CODE	FEB 2
MAPPING M.L. CODE 5"	AUG 2
MAPPING M.L. CODE CONT'D.	MAR 7
MAPPING MI, CODE CONTUD	ADD 8
MADDING M L CODE CONT D.	MAY DG
MAPPING M.L. ODE ONITD.	MAI 20
MAPPING M.L. CODE, USE	AUG 2
MAX/MIN OF FUNCTION	AUG 6
MICRO MAGAZINE HISTORY	MAY 11
MINT-LOGO	APR 19
MINOC DIN ETY	NUC 22
MINUS BUG FIX	AUG 22
MINOS TAPE TO DISK	MAR 22
MINOS TAPE TO DISK	MAY 12
M.L. CODE COPY W/O EPROM	JUL 18
	CED 2
M.L. PROS. USING BASIC ROLL.	SEP 2
M.L. SIRING SEARCH ROUTINES	SEP Z
MnM ASSEMBLER, - REVIEW	JAN 11
MODEM FIX C4P	FFB 22
MODEM FOR \$29 00 - KTT	TAN 15
MODEM FOR \$25.00 - KII	UAN 15
MODEM HOOKUP EXPLAINED	MAR 2
MODEM PRINTER FOR OSI PROG.	JAN 14
MONEY FORMAT ROUTINE 650	AUG 10
MONTROP FOROM PEDATO	007 3
MONITOR EFROM REFAIR	
MOVING BASIC PROG. ADDRESS	MAY 10
MULTI-SYSTEM UTILITIES-REVIEW	MAY 21
NAME SWAP FIRST/LAST, 65U	AUG 12
NETAL A.C. CIRCUITT ANAL, PROG	SEP 3
ODTIMIZE ALCORTINA	NIC 6
OFI MIZE ALGORITHM	AUG 0
OSI BEATS IBM PC	SEP /
OSI HARDWARE INSIDE	mar 2
PACKING FILES D3.3	JAN 4
DACNET	SED 22
	3EF 22
PADDING FOR DMS FILLES	AFR 19
PARALLEL INTERFACE EXPLAINED	APR 2
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE	NOV 18
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA	NOV 18 IAN 23
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLICE FOR 600 HOME MADE	NOV 18 JAN 23
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE	APR 2 NOV 18 JAN 23 AUG 22
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM	APR 2 NOV 18 JAN 23 AUG 22 OCT 13
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65U FUELAINED	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKENC A DEEK DAMEN	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW	APR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. CENERATOR	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PUBLIC FOR 65D 3.2	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 22
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER FRGGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.2	AHR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFE 19
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A FEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG.	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 14
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER FRGGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER HOCKUP EXPLAINED	AHR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 AFR 2
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE FOINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 JAN 4 JAN 14 APR 20 JAN 4 JAN 14 APR 2
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 AFR 2 JUN 8
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JAN 14 AFR 2 JUN 8 MAY 21
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 14 AFR 2 JUN 8 MAY 21 JUN 8
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLVING APFROACH	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 14 AFR 2 JUN 8 MAY 21 JUN 18 JUN 18 JUN 18 JUN 18
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLVING APPROACH PROFILE	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JAFR 2 JUN 8 MAY 21 JUN 18 JUN 18 MAY 20
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D EXPLAINED POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER HOOKUP EXPLAINED PRINTING GRAPHICS 65D PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - DHONE SYSTEM	AHR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 14 AFR 20 JAN 4 JAN 21 JUN 18 JUN 18 JUN 20 MAY 21
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLVING APPROACH PROFILE - PHONE SYSTEM	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 AUG 12 JAN 4 JAN 14 AFR 2 JUN 8 MAY 21 JUN 8 JUL 16 MAY 20 JUL 20
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK (X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A FEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JAFR 2 JUN 18 JUN 18 JUN 18 JUN 20 NOV 20
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SMEMBLE PROGRAMMING - BEGINNERS	AHR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 14 AFR 20 JAN 4 JAN 21 JUN 18 JUL 6 MAY 21 JUN 18 JUL 6 MAY 20 MAR 11
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING - PROBLEM SOLVING	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 AUG 12 JAN 4 JAN 14 AFR 2 JUN 8 MAY 21 JUN 8 JUL 20 NOV 20 MAR 11 AFR 16
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BRGINNERS PROGRAMMING - BRGINNERS PROGRAMMING A DS 65U	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JAFR 20 JAN 44 AFR 21 JUN 18 JUN 18 JUN 18 JUN 20 NOV 20 MAR 11 AFR 16 OCT 21
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER HOOKUP EXPLAINED PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING - PROBLEM SOLVING PROGRAMMING AIDS 65U PROFMUE A DEC 65U DET 1	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 24 JAN 20 JAN 4 JAN 20 JAN 20 AUG 22 AFR 20 JAN 4 JAN 20 AUG 22 AFR 20 JAN 20 AUG 12 AFR 20 JAN 20 AFR 20 AUG 12 AFR 20 JAN 20 AFR 20 AUG 12 AFR 20 JAN 20 AFR 20
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING - PROBLEM SOLVING PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PART 1	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 FEB 19 AUG 12 JAN 4 JAN 14 AFR 2 JUN 8 MAY 21 JUN 8 JUL 20 NOV 20 MAR 11 AFR 16 JUL 2 JUN 12
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PLACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.2 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMMING RULES OF 65U	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JUN 18 JUN 18 JUN 18 JUN 20 NOV 20 MAR 11 AFR 16 JUL 20 NOV 20 MAR 11 AFR 16 JUL 20 JUN 20 AUG 21 AFR 20 JUN 20 AUG 22 AFR 20 JUN 20 AFR 20 JUN 20 AFR 20 JUN 20 AFR
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D 3.3 POKES 65D EXPLAINED POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING UNFRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROFILES OF 65U QUICK SORT	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 21 JUN 18 JUN 20 MAY 21 JUN 18 JUN 20 MAY 21 JUN 18 JUN 20 MAR 11 AFR 16 JUL 2 JUN 12 SEP 13
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE FOINTER PROGRAM POKE LIST 65U UPDATE POKEY, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GRAPHICS 65D PRINTING UNFINITABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS	ARY 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 JUN 14 JAN 14 AFR 20 JAN 4 JAN 14 AFR 20 JUN 8 JUL 20 NOV 20 NOV 20 NOV 20 NOV 20 II AUG 12 FEB 19 JUN 8 JUL 20 NOV 20 JUL 20 NOV 20 II AFR 16 JUL 2 JUN 12 JUN 12 JU
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROFILE - SWEMBLE PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PART 1 PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS PROBLEM BOCODEM	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JUN 18 JUN 18 JUN 20 NOV 20 MAR 11 AFR 16 JUN 20 NOV 20 MAR 11 AFR 17 JUN 18 JUN 18 JUL 20 NOV 20 MAR 11 AFR 16 JUN 12 JUN 18 JUN
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PLACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D EXPLAINED POKES 65D EXPLAINED POKES 65D EXPLAINED POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER HOOKUP EXPLAINED PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PART 1 PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS REGRESSION PROGRAM PROFILE SURVEY RESULTS REGRESSION PROGRAM	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 4 JAN 24 JAN 14 AFR 20 JAN 14 AFR 20 JUN 18 JUN 20 MAX 11 AFR 16 JUL 2 JUN 12 AFR 17 AUG 12 AFR 16 JUN 20 MAR 11 AFR 17 AFR 1
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE FOINTER PROGRAM POKE LIST 65U UPDATE POKEY, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65U EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GNERINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS REGRESSION PROGRAM RESERVED WORD LIST 65U PROFILES	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 JUN 16 OCT 21 OCT 10 AUG 12 FEB 19 JUN 14 JAN 14 JAN 14 JUN 18 JUN 18 JUN 18 JUN 18 JUN 18 JUN 18 JUN 18 JUN 18 JUN 20 MAY 20 JUN 20 MAY 21 JUN 18 JUN 20 AFR 20 JUN 18 JUN 12 JUN 18 JUN 12 JUN 18 JUN 12 JUN
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PIACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.2 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PROFILE ON SULV VS APP. SOFTWARE PROFILE – PHONE SYSTEM PROFILE – PHONE SYSTEM PROFILE – SWEMBLE PROFILE – SWEMBLE PROGRAMMING AIDS 65U PART 1 PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS REGRESSION FROGRAM RESERVED WORD LIST 65U RESMON III	ARV 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 MAY 17 JUN 6 OCT 21 OCT 10 AUG 12 AFEB 19 AUG 12 AFEB 19 AUG 12 AFF 20 JAN 14 JAN 20 JAN 4 JAFR 20 JAN 14 JUN 18 JUN 18 JUN 18 JUL 20 NOV 20 MAR 11 AFF 16 AUG 21 JUN 12 JUN 12
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PLACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D EXPLAINED POKES 65D EXPLAINED POKES 65D EXPLAINED POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER FOR OSI MODEM PRCG. PRINTER HOOKUP EXPLAINED PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PART 1 PROGRAMMING RULES OF 65U QUICK SORT READER SURVEY RESULTS REGRESSION ROGRAM RESERVED WORD LIST 65U RESERVED PART 2	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 AFR 20 JAN 14 JAR 20 JAN 14 JUN 18 JUN 20 MAX 11 AFR 16 JUL 20 JUN 12 AFR 17 AUG 22 AFR 20 JAN 14 JUN 20 AFR 11 AFR 16 JUN 20 AFR 17 JUN 17 MAR 14 JUN 7
PARALLEL INTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE FOINTER PROGRAM POKE LIST 65U UPDATE POKEY, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GNERINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMING AIDS 65U PROGRAMING AIDS 65U PROGRAMING AIDS 65U PROGRAMING AIDS 65U PROGRAMING AIDS 65U PROFILE	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 JUN 12 JAN 14 JAPR 20 JAPR 2
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PLACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKE Y, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRIMT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER HOOKUP EXPLAINED PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROFILE - SWEMBLE PROFILE - SWEMBLE PROGRAMMING AIDS 65U PART 1 PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMING AIDS 65U PROGRA	ARK 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 JUN 6 OCT 21 AUG 12 AUG 12 AUG 12 AUG 12 AUG 12 AUG 12 AUG 12 AUG 12 AUG 12 AUG 20 AUG 12 AUG 20 AUG 12 AUG 20 AUG 20 AUG 12 AUG 20 AUG
PARALLEL INTERFACE EXPLAINED PARALLEL INTERFACE EXPLAINED PLACOM - ERRATA PLUG FOR 600, HOME MADE POINTER PROGRAM POKE LIST 65U UPDATE POKES 65D 3.3 POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT PRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTER FOR OSI MODEM PROG. PRINTING UNPRINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PART 1 PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROFILE SOURCE PART 2 RESOURCE PART 2 RESOURCE PART 2 PROVIEW - COMPUSERVE - CSI-SIG PRUFINE - CSI-SIG	ARR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 21 OCT 10 AUG 12 FEB 19 AUG 12 APR 20 JAN 14 JAN 24 JAN 27 JAN 14 JUN 20 MAX 11 JUN 20 MAX 21 JUN 18 JUN 20 MAX 21 JUN 18 JUN 20 MAX 21 JUN 18 JUN 20 MAX 11 JUN 20 AUG 12 JUN 18 JUN 18 JUN 20 AUG 12 JUN 18 JUN 20 AUG 12 JUN 18 JUN 18 JUN 18 JUN 12 JUN 18 JUN 12 JUN
PARALLEL IMTERFACE EXPLAINED PEEK LIST 65-U UPDATE PIACOM - ERRATA PLUG FOR 600, HOME MADE FOINTER PROGRAM POKE LIST 65U UPDATE POKEY, PEEK(X) PROB. EXPLAIN POKES 65D 3.3 POKES 65D EXPLAINED POKING A PEEK PATCH PORTLAND BOARD - REVIEW PRECISION TO 32 BILLION 65U PRETTY PRINT FRIME NO. GENERATOR PRINT USING FOR 65D 3.2 PRINT USING FOR 65D 3.3 PRINTER FOR OSI MODEM PROG. PRINTING GRAPHICS 65D PRINTING GNERINTABLE CHAR. D PROBLEM SOLV VS APP. SOFTWARE PROBLEM SOLV VS APP. SOFTWARE PROFILE - PHONE SYSTEM PROFILE - SWEMBLE PROGRAMMING - BEGINNERS PROGRAMMING AIDS 65U PROGRAMMING AIDS 65U PROFILE PROGRAMMING AIDS 65U PROFILE PROFI	AFR 2 NOV 18 JAN 23 AUG 22 OCT 13 NOV 18 OCT 21 OCT 10 AUG 12 FEB 19 JUN 6 OCT 21 OCT 10 AUG 12 FEB 19 JUN 14 AFR 20 JAN 14 AFR 20 JAN 21 JUN 18 JUN 12 JUN 18 JUN 12 JUN 18 JUN 12 JUN

REVIEW - FOR'I'H PAR'I'	AIG	17
	7 3 31	11
REVIEW - MINI ASSEMBLER	JAN	ΤŦ.
REVIEW - OSI 700 SERIES	aug	9
REVIEW - POPTLAND BOARD	OCT	10
		1 1
REVIEW - SCRIBE WP	յա	14
REVIEW - TAX AIDS II	FEB	14
DEVISA_MILTT_CVCTEM_UTT.TTFC	ΜΔΥ	21
	1.8.77	~
RND FUNCTION	JUN	0.
ROACH MOD FOR C2-4	AUG	22
ROM - TERM REPATR	OCT	3
DEP THEFT DETATE THE BASTC 65D	TIN	13
CAMPERTUE LOOK_ANCIEC	TAN	22
SATTELITE LOOK-ANGLES	UAN	44
SCREEN CHAR. INPUT 65D 3.3	SEP	12
SCREEN DUMP 65D DISK	JUL	17
SODTRE WD - DEVIEW	.π.	14
	~~~	
SEB-1 GRAFIX INSTALLATION	œr	0
SECOND DRIVE DETECT. 65D 3.3	MAR	22
SET FOTTVE SEARCH/PRINT 650	MAY	14
	77767	
SELECTIVE SEARCH/PRINT 050	JUN	2
SEQ. FILE PROG. EDIT/ZERO 65D	SEP	11
SPOLIENTIAL FILE ED. 65D 3.3	AUG	4
	$\sim$	1 2
SEQUENTIAL FILE SORT PROG.	0C1	12
SERIAL INTERFACE EXPLAINED	APR	- 2
SHARED POINTERS - ASM	JUN	8
CUTTI DOM COOT	CFD	13
SHALL-DPM SORI	365	10
SLOW SCAN IV	OCT	70
SN-76489A - SOUND GENERATOR	MAY	1Ø
SOPT - HEAD	TIM	16
SORT - HEAP	UUN	10
SORT - IMPERFECT	NOV	22
SORT ALGORITHMS	SEP	13
SORT ALCORTTHMS	OCT	17
	TINI	16
SURT/MERGE, BUBBLE MUD	JUN	TO
SORTS	JAN	2
SOUND GENERATOR SN 76489A	MAY	10
COFFOU CONTUNTIONUTTED SST-263A	MAV	10
SPECH SINIHOIHIZER SSI 205A	000	- 5
SPEED - PROGRAM DESIGN	DEC	3
SPEED-BASIC	JUL	19
SPEED-COTO, COSTR TABLE	JAN.	8
CODENDCUEER FOR CID	MAV	22
SPREADSHEET FOR CIP	FIAT	22
SQUARE BRACKET PROBLEM	MAY	21
SSI-263A SPEECH SYNTH.	MAY	10
STOP THE DWARVE	MAY	6
	77757	č
STOP THE DWARVE CONT D.	JUN	0
SIRUCTURED PROGRAMMING	DEC	6
SUPERBOARD FIX-IT	NOV	6
SVNDUATU TZ FD BAADD	ΜΔΥ	٦a
SINFHOINIZER DURAD	. LUIT	10
	MAR	- 4
SYSTEM MAP		
SYSTEM MAP TANDON TM100-1 ON OSI	APR	4
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE	APR	4 19
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE	APR AUG	4 19
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW	APR AUG FEB	4 19 14
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED	APR AUG FEB JUN	4 19 14 22
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER	APR AUG FEB JUN MAY	4 19 14 22 23
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG.	APR AUG FEB JUN MAY	4 19 14 22 23 18
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG.	APR AUG FEB JUN MAY JUL	4 19 14 22 23 18
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX	APR AUG FEB JUN MAY JUL FEB	4 19 14 22 23 18 15
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PRCG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP	APR AUG FEB JUN MAY JUL FEB FEB	4 19 14 22 23 18 15 15
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PRCG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D	APR AUG FEB JUN MAY JUL FEB FEB MAY	4 19 14 22 23 18 15 15 15
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURECTOS DE DNT/SECOL	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB	4 19 14 22 23 18 15 15 18 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TUREODOS PRINT/SPOOL	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB	4 19 14 22 23 18 15 15 18 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PRCG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN	APR ADG FEB JUN MAY JUL FEB FEB MAY FEB	4 19 14 22 23 18 15 15 18 16 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS FRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2	APR ADG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB	4 19 14 22 23 18 15 15 18 16 16 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U	APR AJG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AJG	4 19 14 22 23 18 15 15 18 16 16 23
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PRCG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER DATCH 65U	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG	4 19 14 22 23 18 15 15 15 18 16 16 23 11
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2. NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TUREODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER PATCH 65U	APR AJG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG	4 19 14 22 23 18 15 15 15 16 16 23 11
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER PATCH 65U VIDEO DRIVER S - 65U	APR ADG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB ADG OCT JUN	4 19 14 22 23 18 15 15 15 16 16 23 11 11
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PRCG. TF-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVER PATCH 65U VIDEO DRIVERS - 65U WINDCWS 65D 3.3	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG OCT JUN SEP	4 19 14 22 23 18 15 15 15 18 16 16 23 11 11 12
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX ADS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROS. TP-2. NEC DRIVER FIX TP-2. NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBCDOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BREAK RECOVERY	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG OCT JUN SEP MAR	4 19 14 22 23 18 15 15 15 18 16 16 23 11 11 12 21
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER PATCH 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BRACK RECOVERY	APR AUG FEB JUN MAY JUL FEB FEB AUG OCT FEB AUG OCT SEP MAP	4 19 14 22 23 18 15 15 18 16 16 23 11 11 12 21 21
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BREAK RECOVERY WP6502 65D WARM START	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG OCT JUN SEP MAR	4 19 14 22 23 18 15 15 18 16 16 16 16 16 16 23 11 11 12 21 21
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX ADS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROS. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBCDOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVERS - 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BREAK RECOVERY WP6502 65D WARM START WP6502 MULTI-USER CRASHES	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB OCT FEB AUG OCT JUN SEP MAR MAR	4 19 14 22 23 18 15 15 18 16 16 23 11 11 21 21 23
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER ACCH 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D WREAK RECOVERY WP6502 MULTI-USER CRASHES WP6502 VI.2 EXPLAINED	APR AUG FEB JUN MAY JUL FEB FEB MAY FEB AUG OCT FEB AUG OCT JUN SEP MAR MAR	4 19 14 22 23 18 15 15 18 16 16 23 11 11 22 21 23 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BREAK RECOVERY WP6502 65D WARM START WP6502 VI.2 EXPLAINED WP6502 VI.2 EXPLAINED	APR AJR AJR FEB JUN MAY JUL FEB FEB MAY FEB MAY FEB OCT FEB AUG OCT JUN SEP MAR MAR MAR	4 19 14 22 23 18 15 15 16 16 16 23 11 11 22 21 23 16 16
SYSTEM MAP TANDON TM100-1 ON OSI TAPE/DISK CONV. DEPTH CHARGE TAX AIDS II - REVIEW TDS TIPS FOR SPEED TINY COMPILER TOKEN LOAD/SAVE PROG. TP-2 NEC DRIVER FIX TP-2 TERMINAL SET-UP TRACK HEADERS 65D TURBODOS PRINT/SPOOL TV SLOW SCAN VARIABLE LISTER 65D 3.2 VIDEO DRIVER, BUG 65U VIDEO DRIVER, BUG 65U VIDEO DRIVERS - 65U WINDOWS 65D 3.3 WP6502 65D BREAK RECOVERY WP6502 65D WARM START WP6502 V1.2 EXPLAINED WP6502 V1.2 FILES WP6502 V1.2 FILES	APR AJR AJR FEB JUNN MAY JUL FEB MAY FEB MAY FEB MAY FEB OCT FEB AUG OCT JUN SEP MAR MAR MAR	4 19 14 22 23 18 15 15 18 16 16 23 11 11 22 21 23 16 16 22

WATCH THIS SPACE GROW!





DELIVER TO:

, / [†]

', ż.

۰.

P.O. Box 347 Owings Mills, Md. 21117

1

BULK RATE U.S. POSTAGE PAID Owings Mills, MD PERMIT NO. 18

GOODIES	for	051	Users

P.O. Box 347 • Owings Mills, Md. 21117 • (301) 363-3268

) C1P Sams Photo-Facts Manual. Complete schematics, scope waveforms and board photos. All you need to be a C1P or SII Wizard, just

Asst Course

in the

- ) C4P Sams Photo-Facts Manual. Includes pinouts, photos, schematics for the 502, 505, 527, 540 and 1 542 boards. A bargain at
- ) C2/C3 Sams Photo-Facts Manual. The facts you need to repair the larger OSI computers. Fat with useful information, but just
- OSI's Small Systems Journals. The complete set, July 1977 through April 1978, bound and reproduced ( ) by PEEK (65). Full set only
- ) Terminal Extensions Package lets you program like the mini-users do, with direct cursor positioning, 1 mnemonics and a number formatting function much more powerful than a mere "print using." Requires 65U.
- RESEQ BASIC program resequencer plus much more. Global changes, tables of bad references, ( ) GOSUB's & GOTOs, variables by line number, resequences parts of programs or entire programs, handles line 50000 trap. Best debug tool I've seen. MACHINE LANGUAGE - VERY FAST! Requires 65U. Manual & samples only, \$5.00 Everything for
- (+) Sanders Machine Language Sort/Merge for 0S-65U. Complete disk sort and merge, documentation shows you how to call from any BASIC program on any disk and return it or any other BASIC program on any disk and return it or any other BASIC program. on any disk, floppy or hard. Most versatile disk sort yet. Will run under LEVEL I, II, or III. It should more but Sanders says, "...sell it for just ... "
- ) KYUTIL The ultimate OS-DMS keyfile utility package. This implementation of Sander's SORT/ME ( creates, loads and sorts multiple-field, conditionally loaded keyfiles. KYUTIL will load and sort a key over 15000 ZIP codes in under three hours. Never sort another Master File.
- ) Assembler Editor & Extended Monitor Reference Manual (C1P, C4P & C8P)
- ) 65V Primer. Introduces machine language programming.

C1P, C1P MF, C4P, C4P DF, C4P MF, C8P DF Introductory Manuals (\$5.95 each, please spec (

- Basic Reference Manual (ROM, 65D and 65U) ( )
- C1P, C4P, C8P Users Manuals (\$7.95 each, please specify)

How to program Microcomputers. The C-3 Series (

Professional Computers Set Up & Operations Manual - C2-OEM/C2-D/C3-OEM/C3-D/C3-A/ ( ) C3-C/C3-C'

( ) Cash enclosed Account No	( ) Master Charge	() VISA
Signature	· · ·	<u></u>
Name		
Street		
City	State	Zip

BASIC program		
IT: It should cost	•	
	\$89.00 \$	
s SORT/MERGE	•	
:	\$100.00 \$	<u> </u>
	\$6.95 \$	
	\$4.95 \$	· · · · · · · · · · · · · · · · · · ·
lease specify)	\$5.95 \$	
	,\$5.95 \$	
	\$7.95 \$	
	\$7.95 \$	• <u>.</u>
3-D/C3-A/C3-B/	\$8.95 \$	<u></u>
TOTAL		\$
MD Residents add 5% Tax		\$
C.O.D. orders add \$1.90		\$
Postage & Handling		\$ <u>3.70</u>
TOTAL DUE		\$
POSTAGE MAY V	ARY FOR OVER	RSEAS

\$7.95 \$

\$15.00 \$ _

\$30.00 \$ _

\$15.00 \$ ...

\$50.00 \$ _

\$50.00 \$ _