Ipso Facto

The A-C-E Magazine

INDEX INDEX PAGE

Ipso Facto 19

Errata—S-100 Interface 3
Errata—A Better Boot 4
Re: A Hardware Bug In the 1802 4
Another Monitor 6
Tektron Number Cruncher 22
The Case of A Display That Swims 23
Improved Tape Controller 25
TV Chess Board 26
The 1802 And TTL 28
Cassette Load Volume Sensitive? 29
Tic - Tac - Toe Modification 29

IPSO FACTO is published by the ASSOCIATION OF COMPUTER EXPERIMENTERS (A.C.E.), a non-profit, educational organization. Information in IPSO FACTO is believed to be accurate and reliable. However, no responsibility is assumed by IPSO FACTO or the ASSOCIATION OF COMPUTER EXPERIMENTERS for its use; nor for any infringements of patents or other rights of third parties which may result from its use.

Send all A.C.E. correspondence to:

Bernie Murphy
102 McCraney Street
Oakville, Ontario
Canada L6H 1H6

ASSOCIATION OF COMPUTER EXPERIMENTERS 1980-81 EXECUTIVE

PRESIDENT:	John Norris	3 Country Club Toronto, Ont. 416-239-8567
PAST President:	Ken Bevis	220 Cherry Post Rd., Mississauga, Ont., L5A 1H9 416-277-2495
SECRETARY/ Treasurer:	Mike Franklin	24 Duby Rd., Acton, Ont., L7J 2P1 519-853-3421
HARDWARE COORDINATOR:	Anthony Tekatch	RR 1 Caistor Centre, Ont., LOR 1EO 416-957-7556
HARDWARE PRODUCTION AND SALES:	Fred Pluthero	1013 Upper Wellington St. Hamilton, Ont., L9A 3S4 416-389-4070
PUBLISHING COMMITEE:	Dennis Mildon	44 Wildewood Ave., Hamilton, Ont., LBT 1X3 416-385-0798
	John Hanson	955 Harvey Place, Burlington, Ont., L7T 3E9 416-637-1076
PROGRAM COORDINATOR:	Jeff Davis	8 Hillview Dr., Grimsby, Ont., L3M 4E5
TRAINING COORDINATOR:	Fred Feaver	
	Ken Bevis	
SOFTWARE COORDINATOR:	Wayne Bowdish	149 East 33rd. St., Hamilton, Ont., L8V 3T5 416-388-7116
EDITOR:	John Myszkowski	99 Augusta St., Hamilton, Ont., 416-529-0250
	Fred Feaver	105 Townsend Ave., Burlington, Ont., L7T 1Y8 416-637-2513
CONSULTANT:	Bob Silcox	562 Forestwood Cr., Burlington, Ont., L7L 4K3 416-681-2848
DRAUGHTSMAN:	John Myszkowski	
MEMBERSHIP COORDINATOR:	Bernie Murphy	102 McCraney St., Oakville, Ont., L6H 1H6 416-845-1630
	Don MacKenzie	3124 Bonaventure Dr., Mississauga, Ont., L4T 2J2 416-676-9084

ARTICLE SUBMISSIONS

We can always use lots of software and hardware related articles of all types. Inasmuch as editing consists of taking the path of least resistance., 'camera ready' articles stand the best chance of getting in. Camera ready means typed , single spaced, reasonably error free and done with a dark ribbon. Diagrams should be large and clear (we can reduce them) and clearly labeled.

Don't let camera ready scare you off. If don't have access to a typewriter, by all means send in what you have, we still want to see what you've been up to.

Some important notes: First, please send us your original manuscript, not a photocopy. The quality of most photocopies is invariably poor and such articles get pushed to the back of the editorial 'stack'. Second, make sure your diagrams and programs are **accurate**. We have enough trouble with errors on our part, there's no way we'll ever catch yours.

MEMBERSHIP RENEWALS

This is a bit pointless; if you got this issue, you're fully paid up. To help you keep track, a note designating the status of your subscription should appear on the mailing label of your issue. This will most probably take the form of '2 of 6'.

US MEMBERS

We understand that a number of our US members are having trouble obtaining Canadian currency. Don't bother, send in your membership renewal (or order for DeFacto, hint, hint) in US funds.

This also applies to overseas members, if you can't come up with Canadian money easily enough, by all means send cheque or money order in US funds.

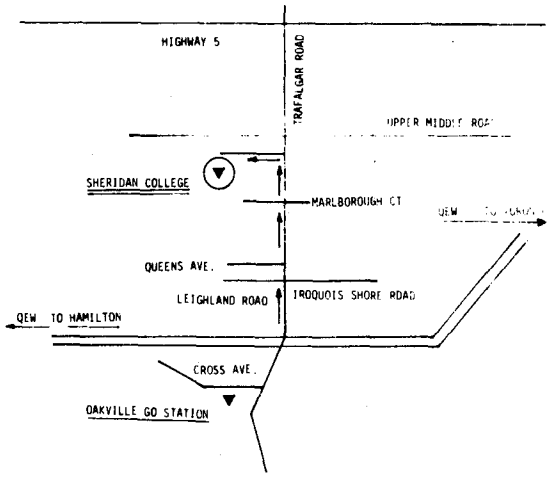
MEETINGS

Meetings are held on the second Tuesday of every month. The next two are:

November 11; A Look At Output Devices, Part I. John Myszkowski descrbes video displays and how to implement them.

December 9; A Look At Output devices, PartII. Bernie Murphy talks about using and interfacing with printers.

Hardware tutorials start at 6:45 PM and the actual meeting starts at 7:30—7:45 PM. Meetings will take place at Sheridan College on Trafalgar Road in Oakville. Once in go through the main entrance and across from the cafeteria is room B123, the meeting site. Don't drink the machine coffee.



DE FACTO

De Facto is now printed and ready to go. For new members who haven't heard, DeFacto is essentially a collection of reprints from the first three years of Ipso Facto. What makes De Facto better than back issues (which you can't get anyways) is that all known corrections have been incorporated. This makes De Facto probably the best source of hardware and software articles around! De facto consists of over 600 pages prepunched for easy insertion in a standard three ring binder. Also (for what it's worth) the collection comes with a pretty impressive three colour cover. Cost is $15.00 postpaid for Canadian members, $18.00 US to US members and $20.00 US for people overseas. Send cheque or money order to Bernie Murphy, 102 McCraney Street, Oakville, Ontario L6H 1H6.

SILICON ON SAPPHIRE 1802s

At the September meeting someone asked when we would see SOS 1802s. I think it was more a rhetorical question than anything else but it did stay in my mind until I came across an article in the July 31 issue of Electronics.

Essentially RCA has decided to dump SOS for the time being. It seems that the process just isn't viable at a commercial level.

Also RCA feels that it needs a 16 bit microprocessor, but is undecided whether to model it after the 1802. It is thought that it is much too late for RCA to break into the 16 bit market and might just end up second sourcing someone else's product.

Hopefully that answers the question.

ERRATA No. 15,'S100 INTERFACE'

The IC designations in theS100 Interface were missing from the circuit on page 30. They should be;

U1, 2	4042	4 bit latch
U3, 4, 5	4050	hex buffer
U6, 7	74367	bus driver
U8	4049	hex inverter buffer

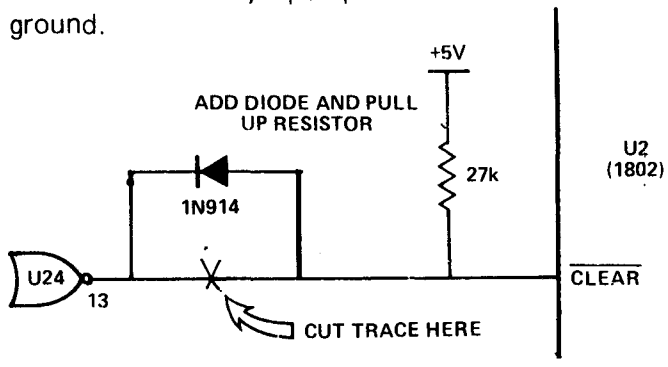
ERRATA No. 15, 'A BETTER BOOT'

One error inthe schematic. The A1 and A0 lines for the 2708 should be taken from the bus and not switch controlled while A8 and A9 (not shown in the schematic) are.

Second Tom Crawford has successfully built and tested the boot circuit. One change is required to make the circuit work: move the wire on pin10 of the 14017 ("Q4") to pin 1 ("Q5"). This results in 4 machine cycles instead of 3 being executed before deselecting control from the EPROM and releasing control of $\overline{MRD}$ back to the CPU. This is because of a special initialization cycle the CPU performs after a RESET/ RUN state change.

To operate the boot circuit, be sure you have made the $\overline{MRD}$ wiring change on the CPU board, place the control switch on the boot circuit to 'enable' the press the RESET, then RUN keys on the CPU board. The machine will execute a long branch to the page specified by the boot circuit DIP switches and execute the code it finds there.

If you would like the ultimate control of a single 'Boot Button', some additional changes are required. Assuming you have a TEC1802, you should make the changes detailed in the diagram. Now you can add a normally open push button switch between the bus $\overline{CLEAR}$ line (connector pin no.4) and ground.



Kenneth A. Mantei
1254 E. 26th St.
San Bernardino, Calif. 9240
August 1980

Two Other Uses For Netronics Elf II Load Switch

Execution of a program can be temporarily halted by flipping the LOAD switch on while the RUN switch is also on. This puts the 1802 in a WAIT state. Flipping the LOAD switch off allows execution to continue. One use of this might be to stop a fast-changing output display in BASIC for closer scrutiny, without aborting the program.

While troubleshooting address line problems it's handy to be able to put a voltmeter on the suspect address line and single-step across an address boundary that should cause a change in address line state. For example(s), to check A_7 (or A_{10}) use the monitor to load 00 at 007D (or 7FFD). Then load C0 00 7D (or C0 7FFD) at 0000. Turn LOAD off, RUN on. The CPU will jump to 007D (or 7FFD). Turn LOAD on while RUN is still on. Then turn RUN off leaving LOAD on and press the Input key to step across the address boundary.

Tom Pittman
P.O. Box 6539
San Jose, CA
95150

Re: A Hardware Bug In The 1802

I was appalled at the monstrous bug exposed by G. Pick in his communication on Page 25 of issue #18, as well as by the callous indifference of RCA to the plight of programmers exposed to this monster. Then I realized that for the same heinous crime we should also castigate Intel, Motorola, Zilog, TI, MOS Technology, National Semiconductor, Rockwell, DEC, HP, and all other manufacturers of computers with hardware stacks used by interrupts... Hmmm... the 1802 does not have a true hardware stack...

very well, we should also castigate Data General, Univac, and IBM, and all manufacturers of machines with software stacks used by interrupts, for not properly warning users that data popped off a stack that is also used by an interrupt service routine is no longer available.

You see, the 1802 encourages the use of R2 to stack register contents in servicing an interrupt. This is not a hardware requirement, for any register can be used (even R1, nonsensical though it may normally be), but to use a register other than R2 costs an extra instruction (SEX). RCA recommends dedicating R2 to a system stack, but this also is not a hardware requirement (the alternatives are quite expensive of time and code, and are not worth considering). One of the characteristics of such a system stack is that any time the stack is popped, the popped byte can no longer be considered available in memory. In fact, RCA makes explicit concession to the general utility of R2 as a scratchpad data pointer by recommending that it be decremented in the interrupt service routine before storing any data in the stack.

Mr. Pick's demonstration program is using the top of the system stack for temporary data storage during I/O; by setting X=2 he makes the OUT instruction into an "Output-and-Pop-Stack" instruction, after which, by the definition of a stack, he can no longer expect the byte that was output to remain around -- it was only his misfortune that the datum in question was not destroyed more often. Yes, you can program around this bug: you can define the top two bytes of the stack to be "free" in the sense that they are preserved below the stack pointer by adding extra DEC R2 instructions to your interrupt service routine (and matching INC R2 instructions at the end), or you can bite the bullet and treat R2 like a true stack pointer -- that is, NEVER increment it by any instruction past any byte you want preserved.

But hardware bug?? Aw, c'mon, you guys, this is no more a hardware bug than the fact that MARK decrements R2 after storing instead of before, and considerably less painful. It is simply in the definition of the instruction set. Mr. Pick's problem is a program bug, nothing more nor less. The 1802 may yet have hardware bugs, but this is not one.

FOR SALE: ELF II with Giant Board, (2) 4K memory boards, high/low address board, (3) kludge boards, power supply (-5V@1A, +5V@5A, +12V@5A, +28V@1A), and (300) 16 pin IC expansion board. All populated boards are fully socketed and enclosed in a custom wood cabinet with fan. Software includes Elfbug, Tiny Basic, UT4, and a 1K monitor with Alter memory, Breakpoint set, Copy memory, Dump memory (UT4 format), Fill memory, Inspect registers, jump to address, Load memory (UT4 format), Read cassette tape, and Write cassette tape commands. Documentation and a complete set of IPSO FACTO are included for $300 plus shipping.

Ernest Payne
1331 W. Whispering Hills Drive
Tucson, Arizona 85704
United States of America
Phone: (602) 887-1922

Richard M. Cox
2670 Calle Abedul
Thousand Oaks, CA91360

Another Monitor

AFTER USING UT4 AS A MONITOR FOR MY SYSTEM FOR AWHILE, I DECIDED THAT I NEEDED SOMETHING A LITTLE MORE FLEXIBLE. I LOOKED AROUND AT THE MONITORS AVAILABLE FOR OTHER SYSTEMS, AND DECIDED THAT I WOULD WRITE ONE INCORPORATING SOME OF THE FEATURES THAT I SAW. THE LISTING BELOW IS THE RESULT. IT PROVIDES NOT ONLY A VARIETY OF COMMANDS, BUT A GREAT MANY USEFUL SUBROUTINES THAT CAN BE USED WHEN WRITING OTHER PROGRAMS.

I WILL DESCRIBE THE COMMANDS AND SUBROUTINES AVAILABLE. MY MONITOR IS LOCATED AT F000H, BUT THE LISTING HAS AN ORIGIN AT 0000H. I HAVE LOCATED THE STACK AND SCRATCH RAM AREA AT FF00H. I CAN SUPPLY VERSIONS ON CASSETTE IN NETRONICS FORMAT FOR ORIGINS ON ANY 1K BOUNDARY FOR $20.00 IF ANYONE IS INTERESTED. SPECIFY THE TOP OF YOUR STACK AREA, ALLOWING AT LEAST 128 BYTES BELOW.

COMMANDS:

NOTE: ANY TIME THAT THE MONITOR IS WAITING FOR AN INPUT, A CONTROL-X WILL ABORT THE COMMAND, AND RETURN A PROMPT.

A---ASCII DUMP. THIS ROUTINE WILL FIND AND PRINT THE ASCII EQUIVALENT FOR ALL UPPERCASE ALPHA CHARACTERS. IT WILL ASK YOU FOR THE START AND STOP ADDRESSES OF THE AREA IN WHICH TO SEARCH. IT WILL PRINT THE ADDRESS FIRST, THEN THE ASCII CHARACTER. IF IT FINDS A CONTINUOUS STRING OF CHARACTERS, IT WILL PRINT THEM AS A STRING.

B---BREAKPOINT. THIS ROUTINE WILL INSERT A 00 BYTE INTO A GIVEN LOCATION IN MEMORY. IT WILL SAVE THE ADDRESS AND ORIGINAL BYTE FOR RESTORING LATER. (SEE UNSET).

C---CHANGE. THIS ROUTINE ALLOWS A CONTINUOUS STRING OF BYTES TO BE ENTERED INTO MEMORY. FIRST TYPE THE ADDRESS, THEN THE STRING OF BYTES TO BE ENTERED. TYPING A SEMICOLON WILL ALLOW YOU TO ENTER A NEW STARTING LOCATION ON THE NEXT LINE. A CARRIAGE RETURN TERMINATES THE COMMAND.

D---DO. THIS ROUTINE ALLOWS EXPANSION OF THE COMMAND WORDS TO ENABLE YOU TO RUN OTHER PROGRAMS. IF YOU DESIRE TO USE THIS FUNCTION, INSERT THE STARTING ADDRESSES OF YOUR PROGRAMS INTO DOTBL. AFTER THE COMMAND "DO", TYPE THE KEYLETTER CORRESPONDING TO YOUR PROGRAM. NOTE THAT WHEN ENTERING YOUR PROGRAM, P=3, X=2.

E---EXAMINE. THIS ROUTINE ALLOWS EXAMINATION AND MODIFICATION OF ANY LOCATION IN MEMORY. AFTER TYPING THE FIRST ADDRESS, THE PROGRAM WILL TYPE THE CONTENTS OF THAT LOCATION TO CHANGE THE CONTENTS, TYPE THE CHANGE. IF YOU DO NOT WANT TO CHANGE IT, TYPE SPACE, AND THE NEXT LOCATION WILL BE DISPLAYED. IF YOU WANT TO EXAMINE A DIFFERENT LOCATION, TYPE A SEMICOLON, THEN THE NEW ADDRESS. A CONTROL-X WILL TERMINATE THE COMMAND.

F---FILL. THIS ROUTINE WILL FILL ANY DESIRED BLOCK OF MEMORY WITH A SPECIFIED BYTE. IT WILL ASK YOU FOR THE START AND STOP LOCATIONS, THEN TYPE WITH. ENTER THE BYTE DESIRED.

G---GO. THIS ROUTINE WILL GO TO A NEW PROGRAM WITH P,X=0. TYPE IN THE ADDRESS, THEN HIT RETURN.

H---HEX DUMP. THIS ROUTINE WILL PRINT THE ADDRESS AND CONTENTS OF ANY BLOCK OF MEMORY IN UT4 FORMAT. IT WILL ASK YOU FOR THE START AND STOP LOCATIONS.

I---INPUT PORT. THIS ROUTINE WILL INPUT ONE BYTE FROM ANY PORT 1-7, AND DISPLAY THE CONTENTS. BE CAREFUL TO ENTER ONLY 1-7 FOR THE PORT NUMBER.

J---JUMP. THIS ROUTINE IS SIMILAR TO GO, EXCEPT THAT P=3, X=2. THIS IS USEFUL WHEN ENTERING INTO THE MIDDLE OF A PROGRAM WHICH USES SCRT.

L---LOAD. THIS PROGRAM WILL LOAD A PROGRAM FROM CASSETTE IN NETRONICS FORMAT. TYPE IN THE START AND STOP LOCATIONS, THEN HIT RETURN. IT WILL DISPLAY A "TAPE ERROR" MESSAGE IF AN ERROR OCCURS.
A DISCUSSION ABOUT THE SUBROUTINES TO READ OR WRITE A BIT TO CASSETTE ARE IN ORDER. THE SUBROUTINES CYCLE AND CTEST ARE IDENTICAL TO THOSE USED BY NETRONICS, AND HAVE THE SAME LOW ORDER ADDRESSES. THAT IS WHY I LOCATED MY MONITOR AT F000. THESE SUBROUTINES WILL THEN BE USED BY ALL NETRONICS SOFTWARE. IF YOU LOCATE THE MONITOR ELSEWHERE, AND RETAIN THE NETRONICS MONITOR, YOU CAN CHANGE THE INITIALIZATION OF REGISTER C IN THE SAVE AND LOAD ROUTINES, AND ELIMINATE CYCLE AND CTEST.

M---MOVE. THIS ROUTINE WILL MOVE A BLOCK OF MEMORY TO ANOTHER LOCATION. IT WILL ASK FOR THE START AND STOP LOCATIONS OF THE BLOCK TO BE MOVED, AND THE STARTING LOCATION OF THE AREA TO BE MOVED TO. NOTE THAT IF YOU OVERLAP THE AREAS BY MOVING A BLOCK TO A STARTING LOCATION WITHIN THAT BLOCK, IT WILL NOT WORK.

O---OUTPUT PORT. THIS ROUTINE WILL OUTPUT A SPECIFIED BYTE TO A SPECIFIED OUTPUT PORT. AGAIN, BE CAREFUL TO SPECIFY 1-7 FOR THE PORT NUMBER.

R---REGISTER DISPLAY. THIS ROUTINE WILL DISPLAY ALL 16 REGISTERS WHICH ARE STORED EACH TIME THE MONITOR IS ENTERED. R0, R1, AND R4.1 ARE CLOBBERED DURING THE STORE PROCESS. FOR DEBUGGING PURPOSES, IT IS HELPFUL TO STOP AT A DESIRED PLACE, AND EXAMINE THE REGISTERS. TO DO THIS, USE THE BREAK COMMAND TO STOP THE PROGRAM. THEN RESTART THE MONITOR, AND USE THE REGISTER DISPLAY COMMAND TO EXAMINE THE REGISTERS. FINALLY, USE THE UNSET COMMAND TO RESTORE THE BREAKPOINT.

S---SAVE. THIS ROUTINE WILL SAVE A PROGRAM ONTO CASSETTE IN NETRONICS FORMAT. AFTER ENTERING THE START AND STOP LOCATIONS, HIT RETURN. WHEN THE PROMPT (.) APPEARS, IT IS DONE.

T---TEST MEMORY. THIS IS A SIMPLE MEMORY TEST WHICH WRITES A BLOCK OF MEMORY FIRST FULL OF 00'S, THEN 01,02, ETC. AFTER EACH WRITE, MEMORY IS EXAMINED. ANY LOCATIONS WHICH DO NOT MATCH WILL BE DISPLAYED. TO TERMINATE, YOU MUST RESET THE CPU.

U---UNSET. THIS ROUTINE RESTORES THE PREVIOUSLY SET BREAK-POINT.

V---VERIFY. THIS ROUTINE WILL COMPARE TWO BLOCKS OF MEMORY. IT WILL ASK FOR THE START AND STOP LOCATIONS OF THE BLOCK TO BE COMPARED WITH. IT WILL THEN TYPE WITH. ENTER THE STARTING ADDRESS OF THE SECOND BLOCK. ANY LOCATION WHICH DOES NOT MATCH IN THE SECOND BLOCK WILL BE DISPLAYED.

W---WHERE IS. THIS ROUTINE WILL SEARCH A GIVEN BLOCK OF MEMORY FOR A SPECIFIED STRING OF BYTES. ANY BYTE WHICH IS A "DON'T CARE" MAY BE TYPED AS "XX". THE STRING MAY BE UP TO 48 BYTES, ANY NUMBER OF WHICH CAN BE "XX". AFTER THE "WHERE IS", TYPE THE STRING, TERMINATED WITH A CARRIAGE RETURN. THEN ENTER THE START AND STOP ADDRESSES OF THE BLOCK. ANY STRING FOUND WILL BE DISPLAYED BY THE ADDRESS OF THE LAST BYTE IN THE STRING.

P---PROM PROGRAM. I HAVE LEFT AN AREA OF 130 BYTES STARTING AT 0609H FOR A USER DEFINED PROM PROGRAMMER. LOCATION 0609H NOW CONTAINS A LBR TO COMAND. SIMPLY START YOUR PROGRAM THERE, AND THE P COMMAND WILL GET YOU THERE.

USEFUL SUBROUTINES IN THE MONITOR

CHRIN AND CHROUT:
KEYBOARD AND TERMINAL I/O. THE ONLY I/O ROUTINES USED IN THE MONITOR ARE CHRIN AND CHROUT. IN MY SYSTEM, A UART IS USED TO COMMUNICATE WITH MY TERMINAL. THE UART STATUS PORT IS NO. 6, WITH BIT 1 BEING DATA AVAILABLE, AND BIT 8 BEING TRANSMIT BUFFER EMPTY. THE UART RECEIVE DATA PORT IS NO. 5, AND THE TRANSMIT DATA PORT IS NO. 6. IF YOU NEED TO PATCH THESE ROUTINES, NOTE THAT, ON INPUT, THE CHARACTER IS PUT IN RF.1, AND ON THE STACK, THEN OUTPUT IMMEDIATELY WITHOUT CHECKING FOR TBMT. THE IDEA BEING THAT BY THE TIME A CHARACTER IS INPUT, THE TRANSMIT BUFFER WILL BE EMPTY. THE CHROUT ROUTINE HAS THREE ENTRY POINTS; LINKOT, WHICH OUTPUTS A CHARACTER WHICH IS PASSED AS AN INLINE PARAMETER FROM THE CALLING PROGRAM; CHROUT, WHERE THE CHARACTER IS ALREADY IN RF.1; AND STKOUT, WHERE THE CHARACTER IS ON THE STACK. NOTE THAT USING SCRT, THE CHARACTER MAY BE BURIED. STKOUT CANNOT, THEREFORE, BE USED EXCEPT WHEN THE CALLING PC=5.

OSTRNG:
THIS SUBROUTINE IS USED EXTENSIVELY IN THE MONITOR FOR OUTPUTING STRINGS OF CHARACTERS. WHEN CALLED, THE ASCII TEXT IS PUT AFTER THE CALLING ADDRESS, AND TERMINATED BY AN ASCII ETX (03 HEX).

HOME:
THIS SUBROUTINE IS CALLED WHEN YOU WANT TO CLEAR THE SCREEN. IF YOUR TERMINAL USES OTHER THAN 0C HEX, CHANGE LOCATION 00A2H. IT USES SUBROUTINE DELAY TO PROVIDE A .5 SECOND WAIT AFTER THE SCREEN CLEAR.

CRLF:
THIS SUBROUTINE WILL OUTPUT A CARRIAGE RETURN, LINE FEED, AND SIX NULLS (00 HEX). THIS WILL ALLOW TIME FOR THE CARRIAGE RETURN ON A TELETYPE.

GETASC:
THIS SUBROUTINE IS THE ONE USED TO EXAMINE EACH INCOMING CHARACTER FOR A CTRL-X. IF IT OCCURS IN THE INPUT, THE MONITOR WILL RESET BY EXECUTING A LBR COMAND. OTHERWISE, GETASC HAS NO EFFECT.

INTCTR:
THIS SUBROUTINE IS USED TO INITIALIZE A COUNTER (RD) WITH AN ADDRESS INPUT FROM THE KEYBOARD, AND COMPUTE THE DIFFERENCE, ADD 100H, AND PUT THE RESULT IN RE. IT IS USED WHENEVER YOU SEE THE START AND STOP PROMPTS.

HEXASC:
THIS SUBROUTINE CONVERTS ONE HEX BYTE IN RF.0 TO TWO ASCII CHARACTERS IN R8.

GETADD:
THIS SUBROUTINE WILL INPUT FOUR ASCII CHARACTERS FROM THE KEYBOARD, AND CONVERT THEM TO HEX, AND PUT IN RD. IT IGNORES ALL NON-HEX CHARACTERS (EXCEPT CTRL-X)

PKHEX:
THIS SUBROUTINE GETS TWO ASCII CHARACTERS FROM THE KEYBOARD, CONVERTS THEM TO ONE HEX DIGIT IN RF.0. IT IGNORES ALL NON-HEX CHARACTERS (EXCEPT CTRL-X).

ASCHEX:
THIS SUBROUTINE CONVERTS ONE ASCII CHARACTER IN RF.1 TO HEX IN R9.0. IF THE CHARACTER WAS HEX, DF IS SET TO "1". IF NOT HEX, DF IS SET TO "0", AND R9 WILL BE UNCHANGED.

DELAY:
THIS SUBROUTINE, WHEN USED WITH A 1.76 MHZ CLOCK, WILL RETURN AFTER A TIME SPECIFIED BY TWO BYTES PASSED AS INLINE PARAMETERS. THESE TWO BYTES SPECIFY THE TIME IN MILLISECONDS. THE LARGEST PARAMETER IS FFFEH, OR APPROX 65.5 SECONDS. FOR A DIFFERENT CLOCK, CHANGE LOCATION 0175H.

ASLMEM:
THIS SUBROUTINE WILL PRINT THE CONTENTS OF THE MEMORY LOCATION POINTED TO BY RD.

ASLADD:
THIS SUBROUTINE WILL PRINT THE ADDRESS CONTAINED IN RD, FOLLOWED BY A SPACE.

```
!M
0000 ;          0001 ..MONITOR VERSION 3.2
0000 ;          0002 ..BY RICHARD M. COX
0000 ;          0003 ..
0000 ;          0004 ..
0000 ;          0005 ORG#0000
0000 ;          0006 ..
0000 ;          0007 ..
0000 ;          0008 STACK=2
0000 ;          0009 PC=3
0000 ;          0010 SUB=4
0000 ;          0011 RETURN=5
0000 ;          0012 LINK=6
0000 ;          0013 SUB1=7
0000 ;          0014 SUB2=8
0000 ;          0015 TEMP=9
0000 ;          0016 ASL=#0D
0000 ;          0017 CHAR=#0F
0000 ;          0018 ..
0000 ;          0019 TOPSTK=#FFDF..STACK TOP
0000 ;          0020 REGSTO=#FFFF
0000 ;          0021 BSTOR=#FFB3
0000 ;          0022 STRING=#FFA0
0000 ;          0023 ..
0000 ;          0024 ..
0000 ;          0025 ..
0000 ;          0026 ..
0000 6301;      0027 MONITR: OUT 3,#01..UNIQUE T
0002 ;          0028         ..MY SYSTEM
0002 ;          0029
0002 ;          0030         ..THIS PCA ROUTINE
0002 ;          0031         ..STORES ALL REGISTERS,
0002 ;          0032         ..EXCEPT R0,R1,P4.1
0002 ;          0033         ..ARE CLOBBERED
0002 ;          0034 ..
0002 ;          0035 ..
0002 F8FF;      0036         LDI A.1(REGSTO)
0004 B1;        0037         PHI 1
0005 F8FE;      0038         LDI A.0(REGSTO)
0007 A1;        0039         PLO 1
0008 F8A0;      0040         LDI #A0
000A B4;        0041         PHI SUB
000B E1;        0042         SEX 1
000C F8D0;      0043 LOOP:   LDI #D0
000E 51;        0044         STR 1
000F 21;        0045         DEC 1
0010 94;        0046         GHI SUB
0011 FC70;      0047         ADI #70
0013 CF;        0048         LSDF
0014 FC21;      0049         ADI #21
```

OUTPRT:
THIS SUBROUTINE WILL OUTPUT A BYTE CONTAINED IN RF.0 TO A PORT NUMBER CONTAINED IN R8.0

INPRT:
THIS SUBROUTINE WILL INPUT A BYTE FROM A PORT NUMBER IN R8.0 TO RF.0

COMAND:
THIS IS WHERE THE MONITOR RESETS THE STACK, AND TYPES "MONITOR"

GTCMND:
THIS IS THE RETURN POINT TO THE MONITOR. IT ISSUES THE PROMPT (.), AND WAITS FOR A COMMAND.

CMDTBL:
CONTAINS THE JUMP ADDRESS FOR ALL COMMANDS A-Z. ALL UNUSED ENTRIES CONTAIN THE ADDRESS OF COMAND.

->

```
0016 FC7F;      0050          ADI#7F
0018 B4;        0051          PHI SUB
0019 51;        0052          STR 1
001A D1;        0053          SEP 1
001B 51;        0054          STR 1
001C 2121;      0055          DEC 1;DEC 1
001E 81;        0056          GLO 1
001F FBE0;      0057          XRI #E0
0021 3A0C;      0058          BNZ LOOP
0023 ;          0059          ..NOW INITIALIZE
0023 ;          0060          ..ALL REGISTERS
0023 F800;      0061          LDI A.1(CALL)
0025 B4;        0062          PHI SUB
0026 F83D;      0063          LDI A.0(CALL)
0028 A4;        0064          PLO SUB
0029 F800;      0065          LDI A.1(RET)
002B B5;        0066          PHI RETURN
002C F853;      0067          LDI A.0(RET)
002E A5;        0068          PLO RETURN
002F F8FF;      0069          LDI A.1(TOPSTK)
0031 B2;        0070          PHI STACK
0032 F8DF;      0071          LDI A.0(TOPSTK)
0034 A2;        0072          PLO STACK
0035 F801;      0073          LDI A.1(COMAND)
0037 B3;        0074          PHI PC
0038 F8E7;      0075          LDI A.0(COMAND)
003A A3;        0076          PLO PC
003B D3;        0077          SEP PC
003C ;          0078 ..................................
003C ;          0079
003C ;          0080 ..MULTIPURPOSE CALL ROUTINE
003C ;          0081 ..1. SAVES SUB1 (R7)
003C ;          0082 ..2. SAVES D
003C ;          0083 ..3. STACK LEFT FREE
003C ;          0084 ..
003C ;          0085 ..
003C D3;        0086 CALLEX: SEP PC
003D 22;        0087 CALL:   DEC STACK
003E E2;        0088         SEX STACK
003F 73;        0089         STXD..PUSH D
0040 96;        0090         GHI LINK
0041 73;        0091         STXD..PUSH LINK.1
0042 86;        0092         GLO LINK
0043 73;        0093         STXD..PUSH LINK.0
0044 97;        0094         GHI SUB1
0045 73;        0095         STXD..PUSH SUB.1
0046 87;        0096         GLO SUB1
0047 73;        0097         STXD..PUSH SUB.0
0048 93B6;      0098         GHI PC;PHI LINK
004A 83A6;      0099         GLO PC;PLO LINK
004C 46;        0100         LDA LINK
004D B3;        0101         PHI PC
004E 46;        0102         LDA LINK
004F A3;        0103         PLO PC
0050 303C;      0104         BR CALLEX
0052 ;          0105 ..................................
0052 ;          0106
0052 ;          0107 ..RETURN ROUTINE
0052 ;          0108 ..
0052 ;          0109 ..1. RESTORES D
0052 ;          0110 ..2. RESTORES SUB1 (R7)
0052 ;          0111 ..
0052 ;          0112 ..
0052 D3;        0113 RETEX:  SEP PC
0053 96;        0114 RET:    GHI LINK..RET ADD.1
0054 B3;        0115         PHI PC
0055 86;        0116         GLO LINK..RET ADD.0
0056 A3;        0117         PLO PC
0057 E2;        0118         SEX STACK
0058 60;        0119         IRX
0059 72;        0120         LDXA..POP SUB1.0
005A A7;        0121         PLO SUB1
005B 72;        0122         LDXA..POP SUB1.1
005C B7;        0123         PHI SUB1
005D 72;        0124         LDXA..POP LINK.0
005E A6;        0125         PLO LINK
005F 72;        0126         LDXA..POP LINK.1
0060 B6;        0127         PHI LINK
0061 72;        0128         LDXA..POP D
0062 3052;      0129         BR RETEX
0064 ;          0130 ..................................
0064 ;          0131 ..ROUTINE TO WRITE A BIT
0064 ;          0132 ..TO CASSETTE.
0064 ;          0133 ..
0064 ;          0134 ..
0064 D3;        0135 CYCLEX: SEP PC
0065 7B;        0136 CYCLE:  SEQ
0066 F81D;      0137         LDI #1D
0068 3B6D;      0138         BNF LEN
006A F807;      0139         LDI #07
006C 1D;        0140         INC ASL
006D 52;        0141 LEN:    STR STACK
006E FF01;      0142         SMI #01
0070 336E;      0143         BDF LEN+01
0072 3964;      0144         BNQ CYCLEX
0074 7A;        0145         REQ
0075 02;        0146         LDN STACK
0076 306F;      0147         BR LEN+01
0078 ;          0148 ..................................
0078 ;          0149
```

```
0078 ;
0078 ;
0078 ;
0078 ;
0078 ;
0078 ;
0078 ;
0078 6E;
0079 FA01;
007B 3278;
007D 6D;
007E 66;
007F 22;
0080 BF;
0081 D5;
0082 ;
0082 ;
0082 ;
0082 ;
0082 ;
0082 ;
0082 ;
0082 ;
0082 46;
0083 38;
0084 02;
0085 BF;
0086 9F;
0087 22;
0088 73;
0089 6E;
008A FA80;
008C 3289;
008E 12;
008F 66;
0090 D5;
0091 ;
0091 ;
0091 ;
0091 ;
0091 ;
0091 ;
0091 ;
0091 06;
0092 FB03;
0094 329D;
0096 46;
0097 BF;
```

```
0150
0151 ..CHARACTER INPUT ROUTINE.
0152 ..
0153 ..1. INPUTS CHAR TO STACK
0154 ..2. INPUTS CHAR TO CHAR.1 (RF.1)
0155 ..
0156 ..
0157 CHRIN:  INP 6..UART STATUS
0158         ANI #01..DAV?
0159         BZ CHRIN
0160         INP 5
0161         OUT 6
0162         DEC STACK
0163         PHI CHAR
0164         SEP RETURN
0165 ..................................
0166 ..CHARACTER OUTPUT ROUTINE
0167 ..
0168 ..1. CHROUT OUTPUTS FROM CHAR.1
0169 ..2. LINKOT OUTPUTS FROM @ LINK,
0170      ..THEN INCREMENTS LINK
0171 ..3. STKOUT OUTPUTS FROM STACK
0172 ..
0173 ..
0174 LINKOT: LDA LINK
0175         SKP
0176 STKOUT: LDN STACK
0177         PHI CHAR
0178 CHROUT: GHI CHAR
0179         DEC STACK
0180         STXD
0181 CKTRMT: INP 6..TRMT?
0182         ANI #80
0183         BZ CKTRMT
0184         INC STACK
0185         OUT 6
0186         SEP RETURN
0187 ..................................
0188
0189
0190
0191 ..ROUTINE TO OUTPUT CHARACTER
0192 ..STRING FROM @ LINK, UNTIL ETX (03)
0193 ..
0194 ..
0195 OSTRNG: LDN LINK
0196         XRI #03..ETX?
0197         BZ OSTEX
0198         LDA LINK
0199         PHI CHAR
```

```
009B D40086;
009B 3091;
009D 16;
009E D5;
009F ;
009F ;
009F ;
009F ;
009F ;
009F D40082;
00A2 0C;
00A3 D40164;
00A6 01F4;
00A8 D5;
00A9 ;
00A9 ;
00A9 ;
00A9 ;
00A9 ;
00A9 D40091;
00AC 0D0A;
00AE 000000000000;
00B4 03;
00B5 D5;
00B6 0000;
00B8 ;
00B8 ;
00B8 ;
00B8 ;
00B8 ;
00B8 1D;
00B9 D3;
00BA F80D;
00BC 35BC;
00BE 35B8;
00C0 FF01;
00C2 33BE;
00C4 3DC4;
00C6 30B9;
00C8 ;
00C8 ;
00C8 ;
00C8 ;
00C8 ;
00C8 ;
00C8 ;
00C8 D40078;
00CB 9F;
00CC FB18;
00CE 3AD6;
00D0 F801;
```

```
0200         SEP SUB,A(CHROUT)
0201         BR OSTRNG
0202 OSTEX:  INC LINK..PAST 03
0203         SEP RETURN
0204 ..................................
0205
0206 ..ROUTINE TO HOME TERMINAL
0207 ..
0208 ..
0209 HOME:  SEP SUB,A(LINKOT)
0210         ,#0C
0211         SEP SUB,A(DELAY)
0212         ,#01F4
0213         SEP RETURN
0214 ..................................
0215
0216 ..ROUTINE TO PRINT CR,LF,6 NULLS
0217 ..
0218 ..
0219 CRLF: SEP SUB,A(OSTRNG)
0220         ,#0D0A
0221         ,#000000000000
0222         ,#03
0223         SEP RETURN
0224         IDL;IDL
0225 ..............................
0226 ..ROUTINE TO READ A BIT
0227 ..FROM CASSETTE.
0228 ..
0229 ...
0230         INC ASL
0231 TESTEX: SEP PC
0232 CTEST:  LDI #0D
0233         B2 *
0234 WAIT:   B2 TESTEX-01
0235         SMI #01
0236         BDF WAIT
0237         BN2 *
0238         BR TESTEX
0239
0240 ..............................
0241 ..ROUTINE TO GET ONE ASCII CHAR.
0242 ..IF CONTROL X, GOES TO COMAND.
0243 ..
0244 ..
0245 GETASC: SEP SUB,A(CHRIN)
0246         GHI CHAR
0247         XRI #18..CTRLX?
0248         BNZ GTASCX
0249         LDI A.1(COMAND)
```

```
00D2 B6;          0250          PHI LINK
00D3 F8E7;        0251          LDI A.0(COMAND)
00D5 A6;          0252          PLO LINK
00D6 D5;          0253 GTASCX:  SEP RETURN
00D7 ;            0254 ..................................
00D7 ;            0255
00D7 ;            0256 ..ROUTINE TO INITIALIZE A
00D7 ;            0257 ..COUNTER.  START ADDRESS IS
00D7 ;            0258 ..RETURNED IN ASL, COUNT OFFSET
00D7 ;            0259 ..BY 100 RETURNED IN E.
00D7 ;            0260 ..
00D7 ;            0261 ..
00D7 D40091;      0262 INTCTR:  SEP SUB,A(OSTRNG)
00DA 5354415254;  0263          ,T'START'
00DF 3A2003;      0264          ,#3A2003
00E2 D40123;      0265          SEP SUB,A(GETADD)
00E5 9D73;        0266          GHI ASL;STXD
00E7 8D52;        0267          GLO ASL;STR STACK
00E9 D400A9;      0268          SEP SUB,A(CRLF)
00EC D40091;      0269          SEP SUB,A(OSTRNG)
00EF 53544F50;    0270          ,T'STOP'
00F3 3A2003;      0271          ,#3A2003
00F6 D40123;      0272          SEP SUB,A(GETADD)
00F9 8D;          0273          GLO ASL
00FA F7;          0274          SM
00FB AE;          0275          PLO E
00FC 60;          0276          IRX
00FD 9D;          0277          GHI ASL
00FE 77;          0278          SMB
00FF FC01;        0279          ADI #01
0101 BE;          0280          PHI E
0102 F0;          0281          LDX
0103 BD;          0282          PHI ASL
0104 22;          0283          DEC STACK
0105 F0;          0284          LDX
0106 AD;          0285          PLO ASL
0107 D5;          0286          SEP RETURN
0108 ;            0287 ..............................
0108 ;            0288
0108 ;            0289 ..ROUTINE TO CONVERT ONE HEX BYTE
0108 ;            0290 ..TO TWO ASCII CHARACTERS.  HEX IS
0108 ;            0291 ..IN CHAR.0, ASCII RETURNED IN SUB2
0108 ;            0292 ..
0108 8F;          0293 HEXASC:  GLO CHAR
0109 F6F6F6F6;    0294          SHR;SHR;SHR;SHR
010D FCF6;        0295          ADI #F6
010F 3B13;        0296          BNF *+04
0111 FC07;        0297          ADI #07
0113 FFC6;        0298          SMI #C6
0115 B8;          0299          PHI SUB2
```

```
0116 8F;          0300          GLO CHAR
0117 FA0F;        0301          ANI #0F
0119 FCF6;        0302          ADI #F6
011B 3B1F;        0303          BNF *+04
011D FC07;        0304          ADI #07
011F FFC6;        0305          SMI #C6
0121 A8;          0306          PLO SUB2
0122 D5;          0307          SEP RETURN
0123 ;            0308 ..............................
0123 ;            0309
0123 ;            0310 ..ROUTINE TO GET FOUR ASCII,
0123 ;            0311 ..CONVERT TO HEX, AND PUT IN ASL
0123 ;            0312 ..
0123 ;            0313 ..
0123 D4012E;      0314 GETADD:  SEP SUB,A(PKHEX)
0126 8F;          0315          GLO CHAR
0127 BD;          0316          PHI ASL
0128 D4012E;      0317          SEP SUB,A(PKHEX)
012B 8F;          0318          GLO CHAR
012C AD;          0319          PLO ASL
012D D5;          0320          SEP RETURN
012E ;            0321 ..............................
012E ;            0322
012E ;            0323 ..ROUTINE TO GET TWO ASCII,
012E ;            0324 ..CONVERT TO HEX, AND PUT
012E ;            0325 ..IN CHAR.0
012E ;            0326 ..
012E ;            0327 ..
012E D400C8;      0328 PKHEX:   SEP SUB,A(GETASC)
0131 D4014C;      0329          SEP SUB,A(ASCHEX)
0134 89;          0330          GLO TEMP
0135 3B2E;        0331          BNF PKHEX
0137 B9;          0332          PHI TEMP
0138 D400C8;      0333 GET2:    SEP SUB,A(GETASC)
013B D4014C;      0334          SEP SUB,A(ASCHEX)
013E 3B38;        0335          BNF GET2
0140 99;          0336          GHI TEMP
0141 FEFEFEFE;    0337          SHL;SHL;SHL;SHL
0145 52;          0338          STR STACK
0146 89;          0339          GLO TEMP
0147 FA0F;        0340          ANI #0F
0149 F1;          0341          OR
014A AF;          0342          PLO CHAR
014B D5;          0343          SEP RETURN
014C ;            0344 ..............................
014C ;            0345
014C ;            0346 ..ROUTINE TO CONVERT ASCII INPUT
014C ;            0347 ..TO HEX. CHAR IS IN CHAR.1. HEX
014C ;            0348 ..RETURNED IN TEMP.0.. IF HEX, DF=1
014C ;            0349 ..
```

SET PLAY !

```
014C ;          0350 ..
014C 9F;        0351 ASCHEX:   GHI CHAR
014D FF41;      0352           SMI #41
014F 3B59;      0353           BNF CKDEC
0151 FF06;      0354           SMI #06
0153 3361;      0355           BDF NFND
0155 FF00;      0356           SMI #00
0157 A9;        0357 FND:      PLO TEMP
0158 D5;        0358           SEP RETURN
0159 FC07;      0359 CKDEC:    ADI #07
015B 3361;      0360           BDF NFND
015D FC0A;      0361           ADI #0A
015F 3357;      0362           BDF FND
0161 FC00;      0363 NFND:     ADI #00
0163 D5;        0364           SEP RETURN
0164 ;          0365 ..................................
0164 ;          0366
0164 ;          0367 ..ROUTINE TO DELAY FOR 1 MSEC TO
0164 ;          0368 ..65 SEC.  PASS PARAMETERS 0000-FFFE
0164 ;          0369 ..
0164 ;          0370 ..
0164 9873;      0371 DELAY:    GHI SUB2;STXD
0166 8873;      0372           GLO SUB2;STXD
0168 46B8;      0373           LDA LINK;PHI SUB2
016A 46A8;      0374           LDA LINK;PLO SUB2
016C 18;        0375           INC SUB2
016D 98;        0376 LOOPD:    GHI SUB2
016E 3A73;      0377           BNZ DLOOP
0170 88;        0378           GLO SUB2
0171 327D;      0379           BZ DELAYX
0173 28;        0380 DLOOP:    DEC SUB2
0174 F823A7;    0381           LDI #23;PLO SUB1
0177 27;        0382           DEC SUB1
0178 87;        0383           GLO SUB1
0179 3A77;      0384           BNZ DLOOP+#04
017B 306D;      0385           BR LOOPD
017D 42A8;      0386 DELAYX:   LDA STACK;PLO SUB2
017F 42B8;      0387           LDA STACK;PHI SUB2
0181 D5;        0388           SEP RETURN
0182 ;          0389 ..................................
0182 ;          0390
0182 ;          0391 ..ROUTINE TO PRINT MEMORY
0182 ;          0392 ..FROM * ASL.
0182 ;          0393 ..
0182 ;          0394 ..
0182 9DB9;      0395 ASLMEM:   GHI ASL;PHI TEMP
0184 8DA9;      0396           GLO ASL;PLO TEMP
0186 0DAF;      0397           LDN ASL;PLO CHAR
0188 D40108;    0398           SEP SUB,A(HEXASC)
018B 98;        0399           GHI SUB2
```

```
018C BF;        0400           PHI CHAR
018D D40086;    0401           SEP SUB,A(CHROUT)
0190 98BF;      0402           GLO SUB2;PHI CHAR
0192 D40086;    0403           SEP SUB,A(CHROUT)
0195 89AD;      0404           GLO TEMP;PLO ASL
0197 99BD;      0405           GHI TEMP;PHI ASL
0199 D5;        0406           SEP RETURN
019A ;          0407 ..................................
019A ;          0408
019A ;          0409 ..ROUTINE TO PRINT ADDRESS OF
019A ;          0410 ..ASL, FOLLOWED BY SPACE.
019A ;          0411 ..
019A ;          0412 ..
019A 9DB9AF;    0413 ASLADD:   GHI ASL;PHI TEMP;PLO CHAR
019D 8DA9;      0414           GLO ASL;PLO TEMP
019F D40108;    0415           SEP SUB,A(HEXASC)
01A2 98BF;      0416           GHI SUB2;PHI CHAR
01A4 D40086;    0417           SEP SUB,A(CHROUT)
01A7 88BF;      0418           GLO SUB2;PHI CHAR
01A9 D40086;    0419           SEP SUB,A(CHROUT)
01AC 89AF;      0420           GLO TEMP;PLO CHAR
01AE D40108;    0421           SEP SUB,A(HEXASC)
01B1 98BF;      0422           GHI SUB2;PHI CHAR
01B3 D40086;    0423           SEP SUB,A(CHROUT)
01B6 88BF;      0424           GLO SUB2;PHI CHAR
01B8 D40086;    0425           SEP SUB,A(CHROUT)
01BB 89AD;      0426           GLO TEMP;PLO ASL
01BD 99BD;      0427           GHI TEMP;PHI ASL
01BF F820BF;    0428           LDI #20;PHI CHAR
01C2 D40086;    0429           SEP SUB,A(CHROUT)
01C5 D5;        0430           SEP RETURN
01C6 ;          0431 ..................................
01C6 ;          0432
01C6 ;          0433 ..ROUTINE TO OUTPUT A BYTE
01C6 ;          0434 ..FROM CHAR.0 TO PORT NO.
01C6 ;          0435 ..IN SUB2.0
01C6 ;          0436 ..
01C6 ;          0437 ..
01C6 22;        0438 OUTPRT:   DEC STACK
01C7 F8D3;      0439           LDI #D3
01C9 73;        0440           STXD
01CA 8F;        0441           GLO CHAR
01CB 73;        0442           STXD
01CC 88;        0443           GLO SUB2
01CD FC60;      0444           ADI #60
01CF 52;        0445           STR STACK
01D0 D2;        0446           SEP STACK
01D1 D5;        0447           SEP RETURN
01D2 ;          0448 ..............................
01D2 ;          0449
```

```
01D2 ;                  0450 ..ROUTINE TO INPUT A BYTE
01D2 ;                  0451 ..FROM PORT NO. IN SUB2.0
01D2 ;                  0452 ..TO CHAR.0
01D2 ;                  0453 ..
01D2 ;                  0454 ..
01D2 22;                0455 INPRT:  DEC STACK
01D3 92B7;              0456         GHI STACK;PHI SUB1
01D5 82A7;              0457         GLO STACK;PLO SUB1
01D7 22;                0458         DEC STACK
01D8 F8D3;              0459         LDI #D3
01DA 73;                0460         STXD
01DB F8AF;              0461         LDI #AF
01DD 73;                0462         STXD
01DE 89;                0463         GLO TEMP
01DF FC68;              0464         ADI #68
01E1 52;                0465         STR STACK
01E2 E7;                0466         SEX 7
01E3 D2;                0467         SEP STACK
01E4 1212;              0468         INC STACK;INC STACK
01E6 D5;                0469         SEP RETURN
01E7 ;                  0470 ..*********************************
01E7 ;                  0471 ..COMAND ROUTINE.
01E7 ;                  0472 ..
01E7 ;                  0473 ..
01E7 F8FFB2;            0474 COMAND: LDI A.1(TOPSTK);PHI STACK
01EA F8DFA2;            0475         LDI A.0(TOPSTK);PLO STACK
01ED D4009F;            0476         SEP SUB,A(HOME)
01F0 D40091;            0477         SEP SUB,A(OSTRNG)
01F3 4D4F4E49544F52;0478             ,T'MONITOR'
01FA 03;                0479         ,#03
01FB C00200;            0480         LBR GTCMND
01FE ;                  0481 PAGE
0200 ;                  0482 ..
0200 ;                  0483 ..
0200 D400A9;            0484 GTCMND: SEP SUB,A(CRLF)
0203 D40091;            0485         SEP SUB,A(OSTRNG)
0206 2E03;              0486         ,#2E03
0208 D400C8;            0487         SEP SUB,A(GETASC)
020B 9F;                0488         GHI CHAR
020C FF41;              0489         SMI #41
020E 3B2A;              0490         BNF NOCMND
0210 FE;                0491         SHL
0211 52;                0492         STR STACK
0212 F6;                0493         SHR
0213 FD19;              0494         SDI #19
0215 3B2A;              0495         BNF NOCMND
0217 F802;              0496         LDI A.1(GOJUMP)
0219 B9;                0497         PHI TEMP
021A F81E;              0498         LDI A.0(GOJUMP)
021C A9;                0499         PLO TEMP
021D D9;                0500         SEP TEMP
021E F842;              0501 GOJUMP: LDI A.0(CMDTBL)
0220 F4;                0502         ADD
0221 A6;                0503         PLO LINK
0222 F802;              0504         LDI A.1(CMDTBL)
0224 B6;                0505         PHI LINK
0225 46B3;              0506         LDA LINK;PHI PC
0227 06A3;              0507         LDN LINK;PLO PC
0229 D3;                0508         SEP PC
022A D40091;            0509 NOCMND: SEP SUB,A(OSTRNG)
022D 494C4C4547414C;0510             ,T'ILLEGAL'
0234 20;                0511         ,#20
0235 434F4D4D414E44;0512             ,T'COMMAND'
023C 03;                0513         ,#03
023D D400A9;            0514         SEP SUB,A(CRLF)
0240 3000;              0515         BR GTCMND
0242 ;                  0516 ..................................
0242 ;                  0517
0242 ;                  0518 ..CMDTBL- COMPUTES JUMP ADDRESS
0242 ;                  0519 ..BASED ON ASCII INPUT A-Z
0242 ;                  0520 ..
0242 ;                  0521 ..
0242 068B;              0522 CMDTBL: ,A(ASCDMP)....A
0244 02B9;              0523         ,A(BREAK).....B
0246 03AE;              0524         ,A(CHANGE)....C
0248 0591;              0525         ,A(DO)........D
024A 0276;              0526         ,A(EXAMIN)....F
024C 02D9;              0527         ,A(FILL)......F
024E 0302;              0528         ,A(GO)........G
0250 0315;              0529         ,A(HEXDMP)....H
0252 03FF;              0530         ,A(INPUT).....I
0254 05F0;              0531         ,A(JUMP)......J
0256 01E7;              0532         ,A(COMAND)....K
0258 048A;              0533         ,A(LOAD)......L
025A 0351;              0534         ,A(MOVE)......M
025C 01E7;              0535         ,A(COMAND)....N
025E 0419;              0536         ,A(OUTPUT)....O
0260 0609;              0537         ,A(PROM)......P
0262 01E7;              0538         ,A(COMAND)....Q
0264 0377;              0539         ,A(REGDIS)....R
0266 0445;              0540         ,A(SAVE)......S
0268 054D;              0541         ,A(TEST)......T
026A 0393;              0542         ,A(UNSET).....U
026C 06CF;              0543         ,A(VERIFY)....V
026E 04DD;              0544         ,A(WHERE).....W
0270 01E7;              0545         ,A(COMAND)....X
0272 01E7;              0546         ,A(COMAND)....Y
0274 01E7;              0547         ,A(COMAND)....Z
0276 ;                  0548 ...............................
0276 ;                  0549
```

```
0276 ;                0550 ..ROUTINE TO EXAMINE A
0276 ;                0551 ..SPECIFIC LOCATION.  TYPE A
0276 ;                0552 ..SPACE TO LOOK AT THE NEXT
0276 ;                0553 ..LOCATION, OR TYPE IN CORRECTION.
0276 ;                0554 ..
0276 ;                0555 ..
0276 D40091;          0556 EXAMIN: SEP SUB,A(OSTRNG)
0279 58414D494E45;    0557         ,T'XAMINE'
027F 03;              0558         ,#03
0280 D400A9;          0559         SEP SUB,A(CRLF)
0283 D40123;          0560 EXADD:  SEP SUB,A(GETADD)
0286 38;              0561         SKP
0287 1D;              0562 NEXT:   INC ASL
0288 D400A9;          0563         SEP SUB,A(CRLF)
028B D4019A;          0564         SEP SUB,A(ASLADD)
028E D40182;          0565         SEP SUB,A(ASLMEM)
0291 D400C8;          0566 ADVANC: SEP SUB,A(GETASC)
0294 9F;              0567         GHI CHAR
0295 FB20;            0568         XRI #20..SPACE?
0297 3287;            0569         BZ NEXT
0299 FB1B;            0570         XRI #1B..;?
029B 3283;            0571         BZ EXADD
029D D4014C;          0572         SEP SUB,A(ASCHEX)
02A0 3B91;            0573         BNF ADVANC
02A2 89B9;            0574         GLO TEMP;PHI TEMP
02A4 D400C8;          0575 ADVNC2: SEP SUB,A(GETASC)
02A7 D4014C;          0576         SEP SUB,A(ASCHEX)
02AA 3BA4;            0577         BNF ADVNC2
02AC 99;              0578         GHI TEMP
02AD FEFEFEFE;        0579         SHL;SHL;SHL;SHL
02B1 52;              0580         STR STACK
02B2 89;              0581         GLO TEMP
02B3 FA0F;            0582         ANI #0F
02B5 F1;              0583         OR
02B6 5D;              0584         STR ASL
02B7 3087;            0585         BR NEXT
02B9 ;                0586 ..................................
02B9 ;                0587
02B9 ;                0588 ..ROUTINE TO INSERT 00 (IDL)
02B9 ;                0589 ..AT SELECTED LOCATION.  LOCATION
02B9 ;                0590 ..AND ORIGINAL BYTE ARE STORED
02B9 ;                0591 ..AT BSTOR.
02B9 ;                0592 ..
02B9 ;                0593 ..
02B9 D40091;          0594 BREAK:  SEP SUB,A(OSTRNG)
02BC 5245414B;        0595         ,T'REAK'
02C0 03;              0596         ,#03
02C1 D400A9;          0597         SEP SUB,A(CRLF)
02C4 92;              0598         GHI STACK
02C5 B7;              0599         PHI SUB1
02C6 F8B3;            0600         LDI A.0(BSTOR)
02C8 A7;              0601         PLO SUB1
02C9 D40123;          0602         SEP SUB,A(GETADD)
02CC E7;              0603         SEX SUB1
02CD 0D;              0604         LDN ASL
02CE 73;              0605         STXD
02CF 9D;              0606         GHI ASL
02D0 73;              0607         STXD
02D1 8D;              0608         GLO ASL
02D2 57;              0609         STR SUB1
02D3 F8005D;          0610         LDI #00;STR ASL
02D6 C00200;          0611         LBR GTCMND
02D9 ;                0612 ..................................
02D9 ;                0613
02D9 ;                0614 ..ROUTINE TO FILL MEMORY
02D9 ;                0615 ..
02D9 ;                0616 ..
02D9 D40091;          0617 FILL:   SEP SUB,A(OSTRNG)
02DC 494C4C;          0618         ,T'ILL'
02DF 03;              0619         ,#03
02E0 D400A9;          0620         SEP SUB,A(CRLF)
02E3 D400D7;          0621         SEP SUB,A(INTCTR)
02E6 D40091;          0622         SEP SUB,A(OSTRNG)
02E9 20;              0623         ,#20
02EA 57495448;        0624         ,T'WITH'
02EE 3A03;            0625         ,#3A03
02F0 D40125;          0626         SEP SUB,A(PKHEX)
02F3 D400C8;          0627         SEP SUB,A(GETASC)
02F6 9E;              0628 FILOOP: GHI E
02F7 32FF;            0629         BZ FILLEX
02F9 8F;              0630         GLO CHAR
02FA 5D;              0631         STR ASL
02FB 1D;              0632         INC ASL
02FC 2E;              0633         DEC E
02FD 30F6;            0634         BR FILOOP
02FF C00200;          0635 FILLEX: LBR GTCMND
0302 ;                0636 ..................................
0302 ;                0637
0302 ;                0638 ..ROUTINE TO RUN PROGRAM
0302 ;                0639 ..WITH X,P=0
0302 ;                0640 ..
0302 ;                0641 ..
0302 D40091;          0642 GO:     SEP SUB,A(OSTRNG)
0305 4F;              0643         ,T'O'
0306 3A2003;          0644         ,#3A2003
0309 D40123;          0645         SEP SUB,A(GETADD)
030C D400C8;          0646         SEP SUB,A(GETASC)
030F 9DB0;            0647         GHI ASL;PHI 0
0311 8DA0;            0648         GLO ASL;PLO 0
0313 E0;              0649         SEX 0
```

```
0314 D0;
0315 ;
0315 ;
0315 ;
0315 ;
0315 ;
0315 ;
0315 D40091;
0318 455820;
031B 44554D5003;
0320 D400A9;
0323 D400D7;
0326 D400C8;
0329 D400A9;
032C D4019A;
032F D40182;
0332 2E;
0333 9E;
0334 C20200;
0337 1D;
0338 8D;
0339 FA0F;
033B 3A46;
033D D40082;
0340 3B;
0341 D400A9;
0344 302C;
0346 F6;
0347 3B4B;
0349 302F;
034B D40082;
034E 20;
034F 302F;
0351 ;
0351 ;
0351 ;
0351 ;
0351 ;
0351 D40091;
0354 4F564503;
0358 D400A9;
035B D400D7;
035E 9DB8;
0360 8DA8;
0362 D40091;
0365 544F3A2003;
036A D40123;
036D 9E;
036E C20200;
0371 48;
```

```
0650          SEP 0
0651 ..............................
0652
0653 ..ROUTINE TO DISPLAY MEMORY
0654 ..FROM START AND STOP ADDRESSES
0655 ..
0656 ..
0657 HEXDMP:  SEP SUB,A(OSTRNG)
0658          ,T'EX',#20
0659          ,T'DUMP',#03
0660          SEP SUB,A(CRLF)
0661          SEP SUB,A(INTCTR)
0662          SEP SUB,A(GETASC)
0663          SEP SUB,A(CRLF)
0664 LINE:    SEP SUB,A(ASLADD)
0665 TLOOP:   SEP SUB,A(ASLMEM)
0666          DEC E
0667          GHI E..DONE?
0668          LBZ GTCMND
0669          INC ASL
0670          GLO ASL
0671          ANI #0F..DIV BY 16?
0672          BNZ TL2..NO
0673          SEP SUB,A(LINKOT)
0674          ,#3B
0675          SEP SUB,A(CRLF)
0676          BR LINE
0677 TL2:     SHR..DIV BY 2?
0678          BNF TSP..IF YES
0679          BR TLOOP
0680 TSP:     SEP SUB,A(LINKOT)
0681          ,#20
0682          BR TLOOP
0683 ..............................
0684
0685 ..ROUTINE TO MOVE MEMORY BLOCK
0686 ..
0687 ..
0688 MOVE:    SEP SUB,A(OSTRNG)
0689          ,T'OVE',#03
0690          SEP SUB,A(CRLF)
0691          SEP SUB,A(INTCTR)
0692          GHI ASL;PHI SUB2
0693          GLO ASL;PLO SUB2
0694          SEP SUB,A(OSTRNG)
0695          ,T'TO',#3A2003
0696          SEP SUB,A(GETADD)
0697 MLOOP:   GHI E..DONE?
0698          LBZ GTCMND
0699          LDA SUB2
```

```
0372 5D;
0373 1D;
0374 2E;
0375 306D
0377 ;
0377 ;
0377 ;
0377 ;
0377 ;
0377 D40091;
037A 4547495354455 2;
0381 3A03;
0383 D400A9;
0386 92;
0387 BD;
0388 F8E0;
038A AD;
038B F801;
038D BE;
038E F81F;
0390 AE;
0391 302C;
0393 ;
0393 ;
0393 ;
0393 ;
0393 ;
0393 D40091;
0396 4E534554;
039A 03;
039B D400A9;
039E 92;
039F B7;
03A0 F8B3;
03A2 FF02;
03A4 A7;
03A5 47AD;
03A7 47BD;
03A9 075D;
03AB C00200;
03AE ;
03AE ;
03AE ;
03AE ;
03AE ;
03AE ;
03AE D40091;
03B1 48414E4745;
```

```
0700          STR ASL
0701          INC ASL
0702          DEC E
0703          BR MLOOP
0704 ..............................
0705
0706 ..ROUTINE TO DISPLAY REGISTERS
0707 ..
0708 ..
0709 REGDIS:  SEP SUB,A(OSTRNG)
0710          ,T'EGISTER'
0711          ,#3A03
0712          SEP SUB,A(CRLF)
0713          GHI STACK
0714          PHI ASL
0715          LDI A.0(REGSTO-#1E)
0716          PLO ASL
0717          LDI #01
0718          PHI E
0719          LDI #1F
0720          PLO E
0721          BR LINE
0722 ..............................
0723
0724 ..ROUTINE TO RESTORE ORIGINAL
0725 ..BYTE REPLACED BY BREAKPOINT
0726 ..
0727 ..
0728 UNSET:   SEP SUB,A(OSTRNG)
0729          ,T'NSET'
0730          ,#03
0731          SEP SUB,A(CRLF)
0732          GHI STACK
0733          PHI SUB1
0734          LDI A.0(BSTOR)
0735          SMI #02
0736          PLO SUB1
0737          LDA SUB1;PLO ASL
0738          LDA SUB1;PHI ASL
0739          LDN SUB1;STR ASL
0740          LBR GTCMND
0741 ..........................
0742
0743 ..ROUTINE TO CHANGE MEMORY.
0744 ..ENTER ADDRESS, SPACE, DATA.
0745 ..IF (;), ENTER NEW ADDRESS.
0746 ..
0747 ..
0748 CHANGE:  SEP SUB,A(OSTRNG)
0749          ,T'HANGE'
```

```
03B6 3B03;          0750          ,#3B03
03B8 D400A9;        0751 CKADD:   SEP SUB,A(CRLF)
03BB D40123;        0752          SEP SUB,A(GETADD)
03BE D400C8;        0753          SEP SUB,A(GETASC)
03C1 9F;            0754          GHI CHAR
03C2 FB20;          0755          XRI #20..SPACE?
03C4 CA0200;        0756          LBNZ GTCMND
03C7 38;            0757          SKP
03C8 1D;            0758 CNEXT:   INC ASL
03C9 D400C8;        0759          SEP SUB,A(GETASC)
03CC 9F;            0760          GHI CHAR
03CD FB3B;          0761          XRI #3B..;?
03CF 32B8;          0762          BZ CKADD
03D1 D4014C;        0763          SEP SUB,A(ASCHEX)
03D4 CB0200;        0764          LBNF GTCMND
03D7 89B9;          0765          GLO TEMP;PHI TEMP
03D9 D400C8;        0766          SEP SUB,A(GETASC)
03DC D4014C;        0767          SEP SUB,A(ASCHEX)
03DF CB0200;        0768          LBNF GTCMND
03E2 99;            0769          GHI TEMP
03E3 FEFEFEFE;      0770          SHL;SHL;SHL;SHL
03E7 52;            0771          STR STACK
03E8 89;            0772          GLO TEMP
03E9 FA0F;          0773          ANI #0F
03EB F1;            0774          OR
03EC 5D;            0775          STR ASL
03ED 30C8;          0776          BR CNEXT
03EF ;              0777 ..........................
03EF ;              0778
03EF ;              0779 ..ROUTINE TO INPUT AND TYPE
03EF ;              0780 ..A BYTE FROM ANY INPUT PORT
03EF ;              0781 ..
03EF ;              0782 ..
03EF D40091;        0783 INPUT:   SEP SUB,A(OSTRNG)
03F2 4E505554;      0784          ,T'NPUT'
03F6 20;            0785          ,#20
03F7 504F5254;      0786          ,T'PORT'
03FB 3A03;          0787          ,#3A03
03FD D400C8;        0788          SEP SUB,A(GETASC)
0400 D4014C;        0789          SEP SUB,A(ASCHEX)
0403 D401D2;        0790          SEP SUB,A(INPRT)
0406 D400A9;        0791          SEP SUB,A(CRLF)
0409 D40108;        0792          SEP SUB,A(HEXASC)
040C 98BF;          0793          GHI SUB2;PHI CHAR
040E D40086;        0794          SEP SUB,A(CHROUT)
0411 88BF;          0795          GLO SUB2;PHI CHAR
0413 D40086;        0796          SEP SUB,A(CHROUT)
0416 C00200;        0797          LBR GTCMND
0419 ;              0798 ............................
0419 ;              0799
```

```
0419 ;              0800 ..ROUTINE TO OUTPUT A BYTE
0419 ;              0801 ..TO ANY OUTPUT PORT
0419 ;              0802 ..
0419 ;              0803 ..
0419 D40091;        0804 OUTPUT: SEP SUB,A(OSTRNG)
041C 5554505554;    0805         ,T'UTPUT'
0421 20;            0806         ,#20
0422 504F5254;      0807         ,T'PORT'
0426 3A03;          0808         ,#3A03
0428 D400C8;        0809         SEP SUB,A(GETASC)
042B D4014C;        0810         SEP SUB,A(ASCHEX)
042E 89A8;          0811         GLO TEMP;PLO SUB2
0430 D400A9;        0812         SEP SUB,A(CRLF)
0433 D40091;        0813         SEP SUB,A(OSTRNG)
0436 4441544I;      0814         ,T'DATA'
043A 3A03;          0815         ,#3A03
043C D4012E;        0816         SEP SUB,A(PKHEX)
043F D401C6;        0817         SEP SUB,A(OUTPRT)
0442 C00200;        0818         LBR GTCMND
0445 ;              0819 ............................
0445 ;              0820
0445 ;              0821 ..ROUTINE TO SAVE PROGRAM
0445 ;              0822 ..ON CASSETTE.
0445 ;              0823 ..
0445 D40091;        0824 SAVE: SEP SUB,A(OSTRNG)
0448 415645;        0825         ,T'AVE'
044B 0D0A03;        0826         ,#0D0A03
044E D400D7;        0827         SEP SUB,A(INTCTR)
0451 9DBA;          0828         GHI ASL;PHI A
0453 8DAA;          0829         GLO ASL;PLO A
0455 D400C8;        0830         SEP SUB,A(GETASC)
0458 F800BC;        0831         LDI A.1(CYCLE);PHI C
045B F865AC;        0832         LDI A.0(CYCLE);PLO C
045E F880BD;        0833         LDI #80;PHI ASL
0461 FF00;          0834 HEADER: SMI #00
0463 DC;            0835         SEP C
0464 9D;            0836         GHI ASL
0465 3A61;          0837         BNZ HEADER
0467 7B;            0838 BYTE:   SEQ
0468 EA;            0839         SEX A
0469 0A;            0840         LDN A
046A 64;            0841         OUT 4
046B BB;            0842         PHI B
046C FC00;          0843         ADI #00
046E F809ABAD;      0844         LDI #09;PLO B;PLO ASL
0472 DC;            0845 BIT:    SEP C
0473 2B8B;          0846         DEC B;GLO B
0475 327C;          0847         BZ PARITY
0477 9BFEBB;        0848         GHI B;SHL;PHI B
047A 3072;          0849         BR BIT
```

```
047C 8DF6;          0850 PARITY: GLO ASL;SHR
047E DC;            0851         SEP C
047F 2E9E;          0852         DEC E;GHI E
0481 3A67;          0853         BNZ BYTE
0483 DCDCDCDC;      0854         SEP C;SEP C;SEP C;SEP C
0487 C00200;        0855         LBR GTCMND
048A ;              0856 ..............................
048A ;              0857
048A ;              0858 ..ROUTINE TO LOAD PROGRAM
048A ;              0859 ..FROM CASSETTE.
048A ;              0860 ..
048A ;              0861 ..
048A D40091;        0862 LOAD:   SEP SUB,A(OSTRNG)
048D 4F4144;        0863         ,T'OAD'
0490 0D0A03;        0864         ,#0D0A03
0493 D400D7;        0865         SEP SUB,A(INTCTR)
0496 9DBA;          0866         GHI ASL;PHI A
0498 8DAA;          0867         GLO ASL;PLO A
049A D400C8;        0868         SEP SUB,A(GETASC)
049D F800BC;        0869         LDI A.1(CTEST);PHI C
04A0 F8BAAC;        0870         LDI A.0(CTEST);PLO C
04A3 F8F9BD;        0871 TEST1:  LDI #F9;PHI ASL
04A6 DC;            0872 TEST2:  SEP C
04A7 3BA3;          0873         BNF TEST1
04A9 9D;            0874         GHI ASL
04AA 3AA6;          0875         BNZ TEST2
04AC DC;            0876 READ:   SEP C
04AD 33AC;          0877         BDF READ
04AF F801BDAD;      0878         LDI #01;PHI ASL;PLO ASL
04B3 DC;            0879 RBIT:   SEP C
04B4 9D7EBD;        0880         GHI ASL;SHLC;PHI ASL
04B7 3BB3;          0881         BNF RBIT
04B9 DC;            0882         SEP C..CHK PARITY
04BA 8DF6;          0883         GLO ASL;SHR
04BC 33C9;          0884         BDF TAPERR
04BE 9D5A;          0885         GHI ASL;STR A..PUT AWAY
04C0 EA;            0886         SEX A
04C1 64;            0887         OUT 4
04C2 2E9E;          0888         DEC E;GHI E
04C4 3AAC;          0889         BNZ READ
04C6 C00200;        0890         LBR GTCMND
04C9 D400A9;        0891 TAPERR: SEP SUB,A(CRLF)
04CC D40091;        0892         SEP SUB,A(OSTRNG)
04CF 54415045;      0893         ,T'TAPE'
04D3 20;            0894         ,#20
04D4 4552524F52;    0895         ,T'ERROR'
04D9 03;            0896         ,#03
04DA C00200;        0897         LBR GTCMND
04DD ;              0898 ..............................
04DD ;              0899
```

```
04DD ;              0900 ..ROUTINE TO SEARCH FOR
04DD ;              0901 ..A STRING.  INSERT XX TO
04DD ;              0902 ..IGNORE A BYTE.
04DD ;              0903 ..
04DD ;              0904 ..
04DD D40091;        0905 WHERE:  SEP SUB,A(OSTRNG)
04E0 4845524520;    0906         ,T'HERE';,#20
04E5 49533A2003;    0907         ,T'IS';,#3A2003
04EA 92B7;          0908         GHI STACK;PHI SUB1
04EC F8A0A7;        0909         LDI A.0(STRING);PLO SUB1
04EF D400C8;        0910 GTSTRG: SEP SUB,A(GETASC)
04F2 E7;            0911         SEX SUB1
04F3 9F73;          0912         GHI CHAR;STXD
04F5 FB0D;          0913         XRI #0D
04F7 3AEF;          0914         BNZ GTSTRG
04F9 D400A9;        0915         SEP SUB,A(CRLF)
04FC D400D7;        0916         SEP SUB,A(INTCTR)
04FF D400A9;        0917         SEP SUB,A(CRLF)
0502 9DBB;          0918 NXSTRG: GHI ASL;PHI B
0504 8DAB;          0919         GLO ASL;PLO B
0506 300D;          0920         BR GOSTRG
0508 1B;            0921 INCSTG: INC B
0509 9BBD;          0922         GHI B;PHI ASL
050B 8BAD;          0923         GLO B;PLO ASL
050D 92B7;          0924 GOSTRG: GHI STACK;PHI SUB1
050F F8A0A7;        0925         LDI A.0(STRING);PLO SUB1
0512 2E;            0926 FDSTRG: DEC E
0513 9E;            0927         GHI E
0514 324A;          0928         BZ WHEREX
0516 07BF;          0929         LDN SUB1;PHI CHAR
0518 D4014C;        0930         SEP SUB,A(ASCHEX)
051B 3B38;          0931         BNF SKIP
051D 89B9;          0932         GLO TEMP;PHI TEMP
051F 27;            0933         DEC SUB1
0520 47BF;          0934         LDA SUB1;PHI CHAR
0522 D4014C;        0935         SEP SUB,A(ASCHEX)
0525 3B4A;          0936         BNF WHEREX
0527 99;            0937         GHI TEMP
0528 FEFEFEFE;      0938         SHL;SHL;SHL;SHL
052C 52;            0939         STR STACK
052D 89;            0940         GLO TEMP
052E FA0F;          0941         ANI #0F
0530 F1;            0942         OR
0531 A9;            0943         PLO TEMP
0532 ED;            0944         SEX ASL
0533 F3;            0945         XOR
0534 1D;            0946         INC ASL
0535 3A08;          0947         BNZ INCSTG
0537 38;            0948         SKP
0538 1D;            0949 SKIP:   INC ASL
```

```
0539 2727;          0950 MATCH:  DEC SUB1;DEC SUB1
053B 07;            0951         LDN SUB1
053C FB0D;          0952         XRI #0D
053E 3A12;          0953         BNZ FDSTRG
0540 2D;            0954         DEC ASL
0541 D4019A;        0955         SEP SUB,A(ASLADD)
0544 D400A9;        0956         SEP SUB,A(CRLF)
0547 1D;            0957         INC ASL
0548 3002;          0958         BR NXSTRG
054A C00200;        0959 WHEREX: LBR GTCMND
054D ;              0960 ......................
054D ;              0961
054D ;              0962 ..ROUTINE TO TEST MEMORY
054D ;              0963 ..
054D ;              0964 ..
054D D40091;        0965 TEST:   SEP SUB,A(OSTRNG)
0550 45535420;      0966         ,T'EST';,#20
0554 4D454D4F525903;0967         ,T'MEMORY';,#03
055B D400A9;        0968         SEP SUB,A(CRLF)
055E D400D7;        0969         SEP SUB,A(INTCTR)
0561 F800A7;        0970         LDI #00;PLO SUB1
0564 9DBB;          0971         GHI ASL;PHI B
0566 8DAB;          0972         GLO ASL;PLO B
0568 9EBA;          0973 AGAIN:  GHI E;PHI A
056A 8EAA;          0974         GLO E;PLO A
056C 9BBD;          0975         GHI B;PHI ASL
056E 8BAD;          0976         GLO B;PLO ASL
0570 87;            0977 GOCHK:  GLO SUB1
0571 5D;            0978         STR ASL
0572 ED;            0979         SEX ASL
0573 F3;            0980         XOR
0574 3A81;          0981         BNZ RAMERR
0576 2A9A;          0982         DEC A;GHI A
0578 327D;          0983         BZ CKDONE
057A 1D;            0984         INC ASL
057B 3070;          0985         BR GOCHK
057D 17;            0986 CKDONE: INC SUB1
057E 87;            0987         GLO SUB1
057F 3068;          0988         BR AGAIN
0581 D4019A;        0989 RAMERR: SEP SUB,A(ASLADD)
0584 D40182;        0990         SEP SUB,A(ASLMEM)
0587 D400A9;        0991         SEP SUB,A(CRLF)
058A 2A9A;          0992         DEC A;GHI A
058C 3268;          0993         BZ AGAIN
058E 1D;            0994         INC ASL
058F 3070;          0995         BR GOCHK
0591 ;              0996 ...................
0591 ;              0997 ..ROUTINE TO DO OTHER PROGRAMS
0591 ;              0998 ..
0591 ;              0999 ..
```

```
0591 D40091;        1000 DO:     SEP SUB,A(OSTRNG)
0594 4F3A2003;      1001         ,T'O';,#3A2003
0598 D400C8;        1002         SEP SUB,A(GETASC)
059B 9F;            1003         GHI CHAR
059C FF41;          1004         SMI #41
059E CB022A;        1005         LBNF NOCMND
05A1 FE52F6;        1006         SHL;STR STACK;SHR
05A4 FD19;          1007         SDI #19
05A6 CB022A;        1008         LBNF NOCMND
05A9 F805B9;        1009         LDI A.1(GODO);PHI TEMP
05AC F8B0A9;        1010         LDI A.0(GODO);PLO TEMP
05AF D9;            1011         SEP TEMP
05B0 F8BC;          1012 GODO:   LDI A.0(DOTBL)
05B2 F4;            1013         ADD
05B3 A6;            1014         PLO LINK
05B4 F805B6;        1015         LDI A.1(DOTBL);PHI LINK
05B7 46B3;          1016         LDA LINK;PHI PC
05B9 46A3;          1017         LDA LINK;PLO PC
05BB D3;            1018         SEP PC
05BC 01E7;          1019 DOTBL:  ,A(COMAND)....A
05BE 01E7;          1020         ,A(COMAND)....B
05C0 01E7;          1021         ,A(COMAND)....C
05C2 01E7;          1022         ,A(COMAND)....D
05C4 01E7;          1023         ,A(COMAND)....E
05C6 01E7;          1024         ,A(COMAND)....F
05C8 01E7;          1025         ,A(COMAND)....G
05CA 01E7;          1026         ,A(COMAND)....H
05CC 01E7;          1027         ,A(COMAND)....I
05CE 01E7;          1028         ,A(COMAND)....J
05D0 01E7;          1029         ,A(COMAND)....K
05D2 01E7;          1030         ,A(COMAND)....L
05D4 01E7;          1031         ,A(COMAND)....M
05D6 01E7;          1032         ,A(COMAND)....N
05D8 01E7;          1033         ,A(COMAND)....O
05DA 01E7;          1034         ,A(COMAND)....P
05DC 01E7;          1035         ,A(COMAND)....Q
05DE 01E7;          1036         ,A(COMAND)....R
05E0 01E7;          1037         ,A(COMAND)....S
05E2 01E7;          1038         ,A(COMAND)....T
05E4 01E7;          1039         ,A(COMAND)....U
05E6 01E7;          1040         ,A(COMAND)....V
05E8 01E7;          1041         ,A(COMAND)....W
05EA 01E7;          1042         ,A(COMAND)....X
05EC 01E7;          1043         ,A(COMAND)....Y
05EE 01E7;          1044         ,A(COMAND)....Z
05F0 ;              1045 ......................
05F0 ;              1046
05F0 ;              1047 ..ROUTINE TO GO TO USER PROGRAM
05F0 ;              1048 ..WITH P=3, X=2
05F0 ;              1049 ..
```

```
05F0 ;              1050 ..
05F0 D40091;        1051 JUMP:    SEP SUB,A(OSTRNG)
05F3 554D502003;    1052          ,T'UMP';,#2003
05F8 D40123;        1053          SEP SUB,A(GETADD)
05FB D400C8;        1054          SEP SUB,A(GETASC)
05FE 93B7;          1055          GHI PC;PHI SUB1
0600 F804A7;        1056          LDI A.0(DOJUMP);PLO SUB1
0603 D7;            1057          SEP SUB1
0604 9DB3;          1058 DOJUMP:  GHI ASL;PHI PC
0606 8DA3;          1059          GLO ASL;PLO PC
0608 D3;            1060          SEP PC
0609 ;              1061 ......................
0609 ;              1062
0609 ;              1063 ..ROUTINE TO PROGRAM EPROMS
0609 ;              1064 ..
0609 ;              1065 ..
0609 C001E7;        1066 PROM:    LBR COMAND
060C ;              1067 ORG*+#7F
068B ;              1068 ..130 BYTES RESERVED
                         ......................

                         ..ROUTINE TO PRINT UPPERCASE
                         ..ASCII ALPHA FROM MEMORY
                         ..
                         ..
068B D40091;        1069 ASCIMP:  SEP SUB,A(OSTRNG)
068E 53434949;      1070          ,T'SCII'
0692 20;            1071          ,#20
0693 44554D50;      1072          ,T'DUMP'
0697 0D0A03;        1073          ,#0D0A03
069A D400D7;        1074          SEP SUB,A(INTCTR)
069D D400C8;        1075          SEP SUB,A(GETASC)
06A0 D400A9;        1076          SEP SUB,A(CRLF)
06A3 30A9;          1077          BR ASCII+02
06A5 F800;          1078          LDI #00..CLEAR FLAG
06A7 AC;            1079 ASCII:   PLO C
06A8 1D;            1080          INC ASL
06A9 2E;            1081          DEC E
06AA 9E;            1082          GHI E
06AB C20200;        1083          LBZ GTCMND
06AE 0D;            1084          LDN ASL
06AF FF41;          1085          SMI #41..CHAR-41
06B1 3BC7;          1086          BNF NOASC
06B3 0D;            1087          LDN ASL
06B4 FD5A;          1088          SDI #5A..5A-CHAR
06B6 3BC7;          1089          BNF NOASC
06B8 8C;            1090          GLO C
06B9 3ABE;          1091          BNZ NOT1ST
06BB D4019A;        1092          SEP SUB,A(ASLADD)
06BE 0DBF;          1093 NOT1ST:  LDN ASL;PHI F
06C0 D400B6;        1094          SEP SUB,A(CHROUT)
06C3 F801;          1095          LDI #01
06C5 30A7;          1096          BR ASCII
06C7 8C;            1097 NOASC:   GLO C
06C8 32A7;          1098          BZ ASCII
06CA D400A9;        1099          SEP SUB,A(CRLF)
06CD 30A5;          1100          BR ASCII-02
                         ......................

                         ..ROUTINE TO COMPARE TWO BLOCKS
                         ..OF MEMORY, AND PRINT DIFFERENCES
                         ..
                         ..
06CF D40091;        1101 VERIFY:  SEP SUB,A(OSTRNG)
06D2 455249465903;  1102          ,T'ERIFY',#03
06D8 D400A9;        1103          SEP SUB,A(CRLF)
06DB D400D7;        1104          SEP SUB,A(INTCTR)
06DE D400A9;        1105          SEP SUB,A(CRLF)
06E1 D40091;        1106          SEP SUB,A(OSTRNG)
06E4 57495448;      1107          ,T'WITH'
06E8 0D0A03;        1108          ,#0D0A03
06EB 9D;            1109          GHI ASL..START ADDR
06EC B7;            1110          PHI SUB1
06ED 8D;            1111          GLO ASL
06EE A7;            1112          PLO SUB1
06EF D40123;        1113          SEP SUB,A(GETADD)..TO ASL
06F2 D400C8;        1114          SEP SUB,A(GETASC)
06F5 D400A9;        1115          SEP SUB,A(CRLF)
06F8 30FC;          1116          BR COMP+02
06FA 171D;          1117 COMP:    INC SUB1;INC ASL..NEXT
06FC 2E;            1118          DEC E
06FD 9E;            1119          GHI E
06FE C20200;        1120          LBZ GTCMND
0701 E7;            1121          SEX SUB1
0702 0D;            1122          LDN ASL
0703 F3;            1123          XOR
0704 C206FA;        1124          LBZ COMP
0707 D4019A;        1125          SEP SUB,A(ASLADD)
070A D40182;        1126          SEP SUB,A(ASLMEM)
070D D400A9;        1127          SEP SUB,A(CRLF)
0710 C006FA;        1128          LBR COMP
0713 ;              1129

0713 ;              1130 END
```

Tektron Number Cruncher

By A. Tekatch

The TEC NOM (Tektron Equipment Corporation Number Oriented Microprocessor) is a self contained hardware addition for the TEC 1802 (it can also easily be adapted to any other micro system). The capabilities of this unit are all the mathematic functions of a scientific calculator and its programming technique is RPN (Reverse Pollock Notation), which makes it easier to work with .

The major purpose for this board is to abolish the memory hogging BASIC MATH package.

The board itself is extremely compact (2"X4" with 11 I.C.'s) , and was designed to sit on the I.F. board of the TEC 1802 system . It's only peculiar requirment is -5v which may be supplied by your +12 ±5V supply . Fortunately the TEC-NOM only requires TPB , $\overline{MRD}$,$\overline{MWR}$, N lines , and the data lines (similar control and select signals may also be used if you own another type of processor).

All handshake sinals are done on the board and with a small USER PROGRAM (≃ 256 bytes).

There are two basic Modes that the TEC NOM will run in

1. Floating Point Mode
2. Scientific Mode

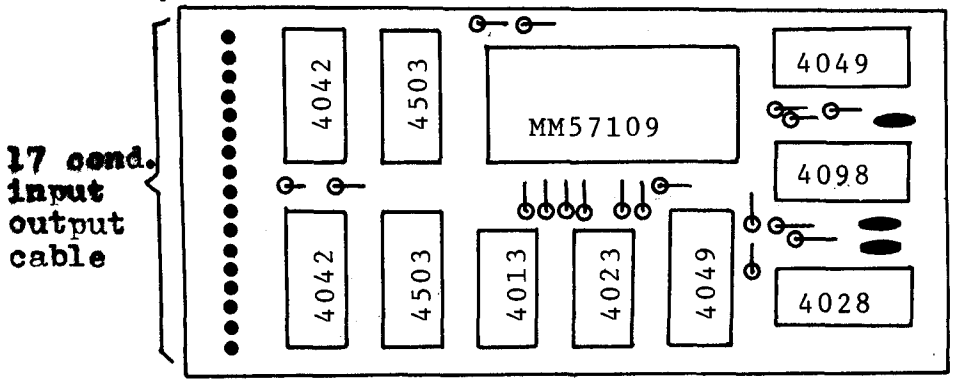
When data is given to the TEC NOM it is sent as in an RPN calculator ie.

1. digit(s)
2. ENTER
3. digit(s)
4. function
5. Output

When data is to be recieved from the calculator , an "OUT" instruction is given to the TEC NOM and the answer digits are recieved in sequence with such information as decimal placement and the sign of answer located in data train .

Data may be entered by simply entering each individual digit as you would in a regular hand held calculator, another way that data can be entered is to load a whole chunk of data at one time in sequence .

The cost of this kit is $65.00 and is available from Tektron Equipment Corporation , 263 Barton St. , Unit 19, Stoney Creek,Ont. L8E 2K4 ph. 662-7820 .



Actual Size

R. C. Francis
111 Guelph Street
Oshawa, Ontario
L1H 6H8

The Case Of a Display That Swims

Okay, you finally convinced the wife that "we are living in the computer age" and that it would be a wise investment to buy your own personal computer. Especially since it would keep your six and seven year old off the bottle and other hard drugs (or whatever the story you made up) and besides being a source of family entertainment, you could also have all her recipes available for display on a monitor sitting on the back burner of the range. Please remember to remove the fuse for that element! Phew!! What a line. Anyhow, she gave the idea her blessing and you bought the Elf II from Netronics. The assembly required a matter of a few hours and everything went well. But as you began to exploit the video graphics of the 1861 composite video generator on board, something fascinating was discovered. The Elf II is indeed a very clever little devil. Not only can it display graphics but the graphics can also "swim". For heaven's sake. Obviously, you did not realize what a real bargain you got for the price. Think about it, a computer with swimming graphics? You bet your sweet bippy. But don't rush out and enter your Elf II in the try-outs of candidates for your Olympic swim team. Not yet anyway. The Elf has one drawback - it is not water proof.

The problem humourously alluded to above is not serious but does give rise to annoyance. It is due mainly to the use of an incorrect crystal frequency from which video sync is derived. In North America, all television signals must conform to a standard established by NTSC (National Television Standard Committee) which requires a horizontal scanning rate of 15750 Hz and a vertical scan rate of 60 Hz for black and white. For displaying a steady "jitter-free" picture it is therefore required to have synchronization pulses of those frequencies in the composite video. Our Elf II uses a simple Colpitts type oscillator configured around an easily available colour burst crystal of 3.579545 MHz frequency. This frequency is divided by 2 to provide a clock frequency of 1.7897725 MHz to the 1802 CPU and the 1861 video display. This clock frequency is divided by 8 and 14 to provide a horizontal sync frequency of 15980 Hz. Further division of the horizontal sync frequency by 262 within the 1861 gives a vertical sync frequency 60.993 Hz. Here is the culprit. Most (if not all) TV monitors or receivers use the line frequency of 60 Hz as a reference for synchronization. Thus, the composite video of the 1861 with a veritcal sync of 60.993 Hz is about 1 Hz away and hence the difficulty of proper synchronization for a stable display on the monitor. Because the vertical sync is obtained by dividing the crystal frequency by a factor of 58688 (2 x 8 x 14 x 262), it would be necessary to change the oscillator by 58.688 kHz in order to change the vertical sync frequency by 1 Hz. This change is too great a percentage to allow "pulling" the crystal frequency as I attempted to accomplish - see Figure 2. (Note! The opposite would be a very easy matter to do if we were concerned with obtaining a higher frequency by multiplication - i.e. a change of 1 Hz multiplied

by 58688 would result in an increase of 58.688 kHz.) The exact crystal frequency required is calculated as: Fc = 60 x 262 x 14 x 8 x 2 = 3.521280 MHz. It is believed that Netronics chose the 3.579545 MHz crystal because it is cheap and easily available - thus we have the classical case of price before quality. It should also be pointed out here that most monochrome TV monitors and receivers are more tolerant of inaccuracies in the horizontal than the vertical sync frequency. So instead of taking your Elf II to the Olympic swimming pool or putting your foot through the television, why not give Lesmith a call and order a 3.521280 MHz crystal. The cost is $6.95 - a lot cheaper than a new Elf or TV set. Lesmith can be reached at 416 844-4505, or write Lesmith Ltd., 54 Shepherd Road, Oakville, Ont.

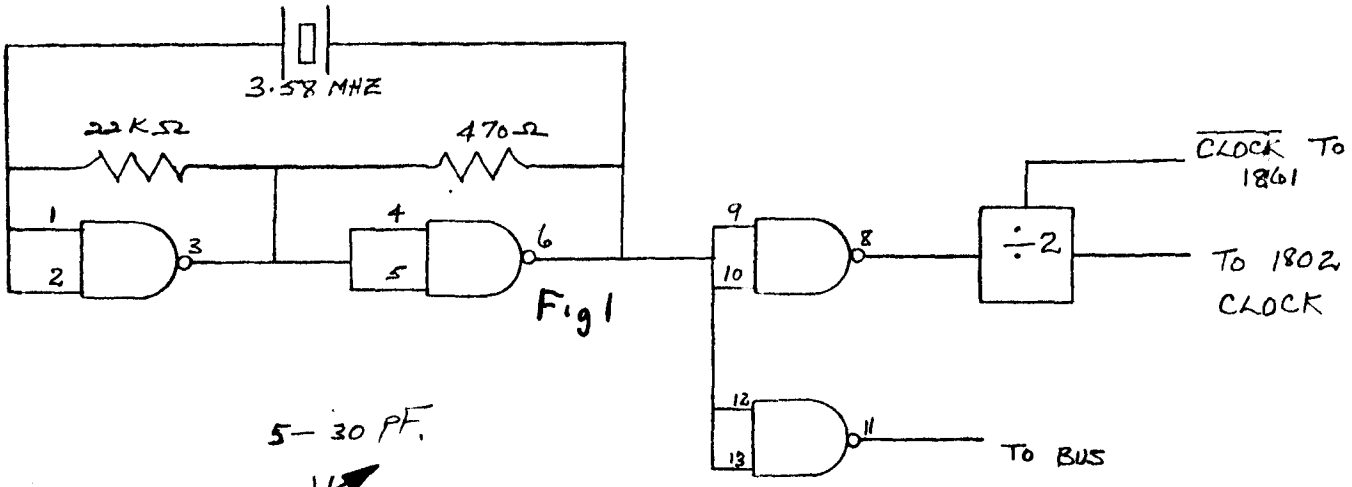


Fig 1

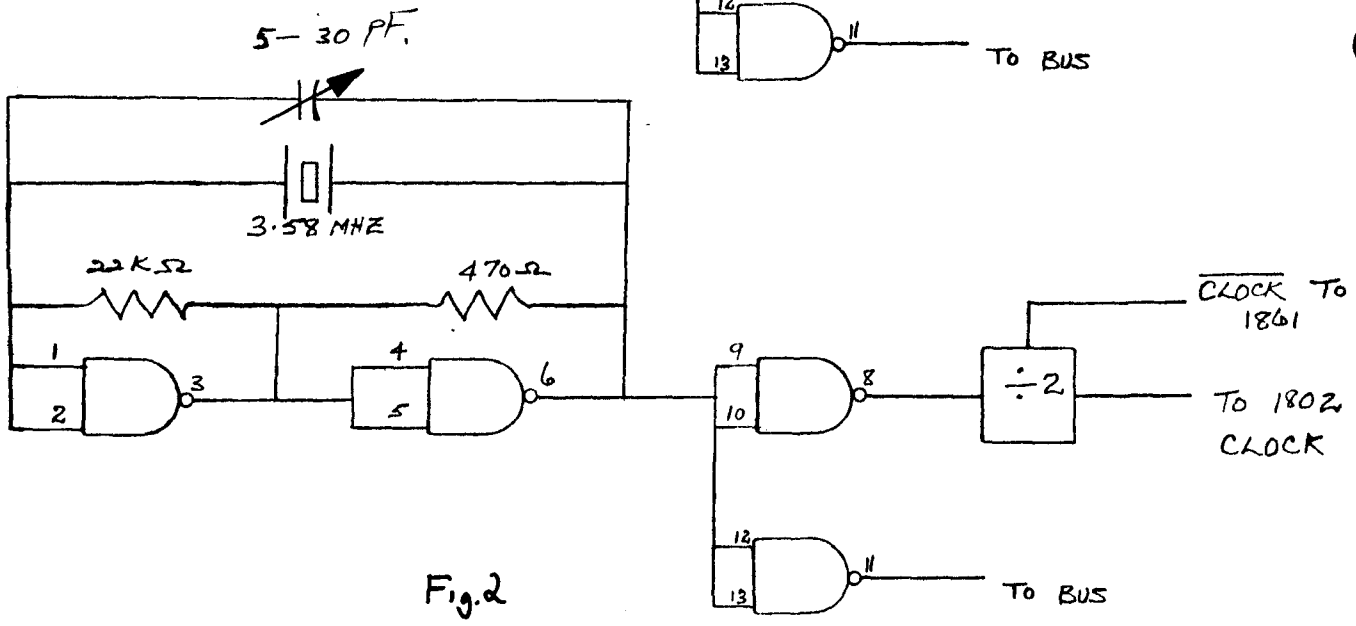


Fig.2

David W Schuler
3032 Avon Road
Bethlehem, Pa.
18017 USA

Improved Tape Controller

I recently purchased the Netronics Text Editor for my Elf II. After examining the software manual, I discovered that I had one of two choices before I could run the program; to either buy or build a cassette tape recorder control board. After looking at the price ($17.95 + p&h) I decided to build the circuit myself.

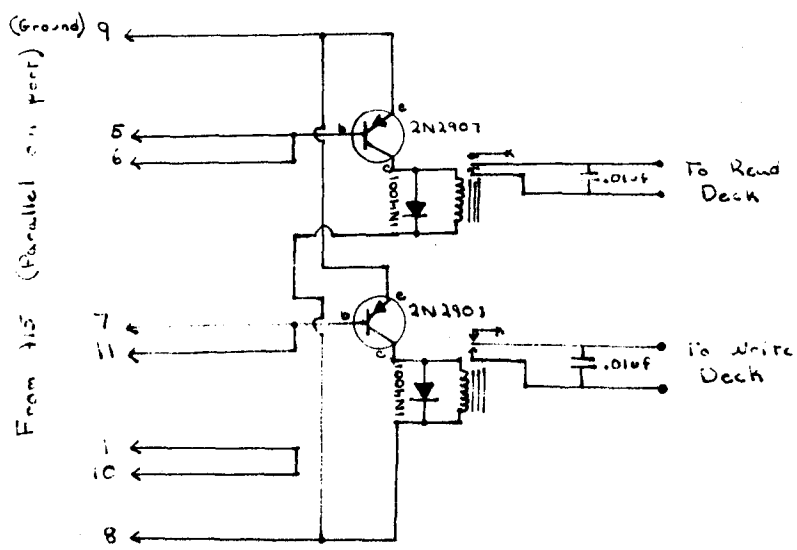
The two relays (one for read and one for write) must have a coil resistance of at least 500 ohms, but I had two DIP relays from Radio Shack that had a resistance of only 50 ohms. The outputs of the 4050 buffer on my Giant Board could not provide enough current to drive these relays.

To drive these relays I built a simple one transistor driver. It uses one 2N2907 transistor for each circuit (other PNP transistors could probably be used just as easily). This provided enough drive for the relays, while at the same time keeping the signal at the same polarity

I also simplified the connections to the recorder itself, while still keeping the operation of the circuit normal (See Figure 1 for the changes.) The ground connection at pin #9 of A15 (parallel out port) must be added in order for this circuit to operate properly (it will not affect normal operation of the port since this pin is not normally used).

I am also in the process of interfacing a Teletype Model 35 (RO) to my Elf. I would appreciate it if anyone who has used one of these would send me his experiences (good or bad) with this machine. I also need a schematic and connection diagram for the electrical interface unit and information on how to connect it to my computer

Figure 1



Dirk Doerr
146 Roberts cres.
Kitchener, Ont.
N2E 1A5

TV Chess Board

After playing with the chess tutor in IPSO FACTO #11 I decided to use my TV screen as a chess board for my ELF II. This program may easily be changed to incorporate the chess tutor.

I use two pages for display and one page of actual program, which is located in page 0200. Stack area is in 01FF. The shape of the board and characters is stored in page 0300-04FF. On start up these two pages will be automatically transposed into pages 0500-06FF. I did not include the configuration of the board and shape of the figures. These can be made as you wish. Each square is 1 word wide by 8 lines high. To distinguish between white and black, I put two dots into the second line from the top of each of the white figures. Do not obliterate the first line, this line is used to reproduce the box once the figure has left. If the figure moves onto an opposite colour it will automatically change to the colour of the field it will occupy.

This program does not check for illegal moves. To move a figure enter the line letter (a-h)(top to bottom) then row number (1-8)(left to right) on which the piece is resting, then the number of the square to which you want the piece to be moved. Anything already occupying that space will be removed. If a piece is removed accidentally then it could be recalled from page 0300-04FF by entering "S" then the original placement of the figure. It may also be used for the promoting of an advanced pawn or to remove an en passant. Note, blank spaces may also be moved as ordinary pieces and anywhere it is placed will become blank although the playing field will never be effected. If a typing error occurs then the escape key will erase the previous inputs.

For subroutines I used my SCRT from Tiny Basic but there is plenty of room within the registers to use a standard SEP technique.

To restart the game just restart the computer. If memory is tight page 0300-04FF may be eliminated but to restart the game the program must be reloaded from tape and memory loc. 02AF must be changed to 05. But then only figures which still remain on the screen may be reproduced with the "S". Note: if ASCII input is not used then the branches in the push button routine, location 0254,56,57, must be changed to suite your input.

TV CHESS BOARD

0200	90B1C4B3	
0204	F884B4B5C4	
0209	F82AA4	
0C	F83CA5	
0F	C4C4C4	
12	F89CA3	main
15	F8FFA2	stack
18	F835A1	interupt
1B	F801B2	
1E	F8FFAFAEC4	set up for
23	F806BEF804BFEE	transfer
2A	0F2F739FFB02	transfer
30	3A2AD3	picture
33	7270C422782252	
3A	F805B0	
3D	F800A0	
40	C4C4E280	
44	E220A0E2	picture
48	3C43	routine
4A	80E220A0E2	
4F	348A	
51	3033D5	
54	3E546F3657BC	
5A	FB1B3A61F89EA3	(ESC.)
61	9CFA0FA8C4	pushbutton
66	3254FCF7	routine
6A	335428C4	
6E	3053	
70	D588	
72	3270	
74	F840AC	
77	172C8C	
7A	3A7728	
7D	3071D5C4 D5	
81	3185 9C5E	
85	8EFC08AE	
89	3B8F	
8B	9EFC01BE	move
8F	8AFC08AA	figure
93	3B99	
95	9AFC01BA	
99	2C307F	
9C	E2697A	start T.V.
9F	F805B7F800A7	
A5	D40254	readP/B
A8	9CFB53	
AB	3AB37BF803B7	
B1	30A5	
B3	D40271	
B6	87AE97BE	
BA	D40254	readP/B
BD	8832C4	
C0	1E28	
C2	30BDF805B7F800A7	
CA	D40254	readP/B
CD	D40271	
D0	87AA97BA	
D4	D40254	readP/B
D7	8832DE	
DA	1A28	
DC	30D7	
DE	6122	
E0	F808ACEA	
E4	0EBCF3	check for colour
E7	3AF30E5A	
EB	D402808C3AE9	
F1	30FD0EFBFF5A	
F7	D402808C3AF3	
FD	309CC4	

Registers in use:

R1 interupt
R2 stack
R3 main
R4 SCRT
R5 SCRT
R6 SCRT
R7 tempory
R8 tempr.
RA points to place where figure is to arrive
RC tempr.
RE points to figure to be moved

Richard N. Thornton
1403 Mormac Road
Richmond, VA 23229

The 1802 And TTL

In the July, 1980 issue of Kilobaud, there is an article entitled "ELF Meets a New Friend" on page 54. In this article, Gregory Harris describes an interface to a SWTP memory board. The startling thing to me was his direct connection from the 1802 microprocessor to TTL chips: DM8097, DM8835, and 74LS138.

While CMOS is excellent in any low power, low speed application, I have often wished for TTL speeds for data and address bus control and decoding. I had come to believe the 1802 could not provide sufficient drive for TTL, or would overheat and burn out. This was based on cautionary statements in magazine articles, as well as frequent use of 4049 and 4050 buffers on the buses. Checking back issues of IPSO FACTO, I found two other circuits where the 1802 is connected to TTL: Issue 4, page 29, and Issue 12, page 26. Referring to the RCA Preliminary Data for the 1802, file number 1023, dated 8/78, I found the following information:

Page 16: "The internal voltage supply V_{DD} is isolated from the Input/Output voltage supply V_{CC} so that the processor may operate at maximum speed while interfacing with various external circuit technologies, <u>including T^2L at 5 volts</u>. All outputs swing from V_{SS} to V_{CC}. The recommended input voltage swing is V_{SS} to V_{CC}." (underline is mine)

Page 2: "Power Dissipation Per Package (P_D): for T_A -55 to 100°C....500mW."

"Device Dissipation Per Output Transistor for T_A = Full Package Temperature Range....100mW".

Page 4: Chart showing typical power dissipation as a function at clock frequency for branch instruction and idle instruction shows that the 1802 uses perhaps 150mW at V_{CC}=10V and Clock = 8 MH_Z (worst case).

Since a TTL input normally requires 1.6MA at 5 volts, or 8mW, the total package dissipation should not be exceeded even if all 24 output pins on the 1802 were buffered by TTL devices. This would result in only 192 MA for the TTL plus 150mW for instruction execution, well within the 500mW maximum.

As I am planning to move my micro to a new enclosure, beef up the power supply, and increase the clock speed per Dan Carrigan's discussion in IPSO FACTO #5, page 44, the increased speed and drive of TTL are very attractive. Below is a list comparing some commonly used bus devices when operated at 5 volts. Delay times shown are taken from National Semiconductor data books, and are typical values.

DEVICE DESCRIPTION	TTL TYPE	DELAY	CMOS TYPE	DELAY
Hex Inverter	7404	8-12	4049	30-45
Hex Tristate Buffer	8097	12	80C97	60-85
Quad Latch	7475	14-16	4042	175
4-50-16 Line Decoder	74154	17-21	74C154	265-275
BCD-to-Decimal Decoder	7442	10-17	74C42	200

Bill Eckel
7711 So. 73 Ave.
Omaha, Ne 68128
U.S.A.
Phone: (402) 592-3044

Cassette Load Volume Sensitive?

If you are like most of us using the 1802 you are using a cassette recorder for mass storage. You may use one you had around the house or even went out and bought one only to find that it was very critical on the volumn setting for it to load into your computer. It wasn't to bad though when loading from one page to even 1K of memory. But now you have more memory and you find that a long program will not load all the time especially a program like tiny basic. If you are like me you get to the point that when you finally got it loaded you didn't ever want to turn off the power or even considered putting it on rom.

I have the Netronics Giant Board on my Elf II and had just those above problems. The Giant Board by Netronics is an excellent I/O board with a decent monitor and worth every penny. But after trying all kinds of recorders, I only found one that would work good with my computer and that was the worst - junkiest recorder I had.

If this sounds like you, try what I discovered. Try switching the wires on the play back head around, either at the play head or at the P.C. board. After doing that I was able to load into my computer with a volumn setting from one to 10 without a single problem. I tried this with several rcorders including those with combined erase heads and those with seperate erase heads. I now have every recorder in my house from the best of quality to the cheapest one loading like a champ.

So don't through away that recorder or go buy a new one. Try this first, it worked great for me. I would like to hear from others that are also successful with this.

'J-G Cayer.
Box 2034,
Hinton, Alt a

Tic-Tac-Toe Modification

Re Guy Gilbert's Tic-Tac-Toe program, I.P. #13 page 15, I made a minor software change to enable two players to use the same hex keyboard. The "O" player enters his selection by punching 01 through 09, while the "X" player enters 31 through 39 for his selection.

There was a slight error at MA 001E. Should be 7D instead of 2D.

Software change:

```
MA 0061  3F 61
     63  37 63
     65  6C
     66  FF 30
     68  32 36
     6A  AA
     6B  59
```

My thanks to Mr Gilbert for a fine program.

HARDWARE PROJECTS AND YOU 27 Oct. 1980.

ACE, is embarking upon a new and additional area of activity in support of the 1802, -the production of quality hardware boards and supporting software. After three years of publishing software and minor hardware projects, sufficient interest, ability, and indeed, demand for hardware support has developed throughout our 600 odd membership.

Our first four projects - a 44 pin, 11 slot backplane, an 8k Eprom(2708) board, a VDU (6847) 16k (2114) memory board and a wirewrap Kluge board, have just now come to fruition after a long summer of trial and a few errors, testing and production hassles. The Eprom and VDU boards are ready to be shipped, and probably will have been by the time you read this note. The backplane and Kluge are still in production, but will be ready shortly.

Our costs have been kept as low as possible, but they do reflect the economies of small production runs. Our prices also include first class postage for North America and airmail overseas, as well as insurance. The club Executive and the Hardware Commitee believe you will be satisfied with the clubs products.

UT WHERE TO FROM HERE? - THAT IS LARGELY UP TO YOU!!!

Now that the club has a common back plane and pin out, the club is in a position to produce other boards of broad interest. Many of you have designed and built boards - Memory, DMA, Real Time Clocks, Interupt Controllers, Remote Sensors, UART circuits, Burgular Alarms, Joysticks, Smoke/heat/wet detectors, Remote switching of appliances- what ever you needed to may you micro work for you. And possibly these circuits will be of interest and use to other club members. If you send us a circuit, you won't get any royalties, you won't get rich, but you will get the credit, a free board, and probably a lot af fan mail.

Interested? Send the club (Fred Plethero) an accurate schematic and a description of the function of the board, a copy of the operating software if required, and if available, a copy of the artwork. The club will evaluate the "marketability" of the project, and then return to you your material. If the Committee believes that the board will generate sufficient interest(15 buyers minimum) to warrent production, they may request you to send us your proto type, or they may use your artwork to produce a board to test. If the origional does not comform to the ACE buss, they will redesign it. The final project board will be built, tested, and assembly and operating software prepared.

Board production based upon this process, especially with voluntary help, will not be fast, but the boards will work, the documentation will be clear, and you will always be able to count on the help of the club if you need it.--ACE guarentees it!!

We have all had our problems with the various suppliers of hardware and software for the 1802, and suppliers in general. Some are helpful, especially as you are placing your order, other mail order houses never seem to answer questions or help, other than repair services at a fee. ACE wont produce, market, and service its products that way, because the "market" is our own club.

OK- lets hear from you..

CLUB HARDWARE PROJECT ORDER FORM

QUANTITY	ITEM	PRICE	SUB-TOTAL
	2708 Eprom	$20.00	
	ACE Backplane	20.00	
	Kluge	20.00	
	VDU - Memory	35.00	

The price of the project boards is:	Canadian Addresses	U.S. and Overseas
EPROM	$20.00 Cdn funds	$20.00 U.S.funds
Backplane	20.00 "	20.00 "
Kluge	20.00 "	20.00 "
VDU/Memory	35.00	35.00

The above price includes first class, insured postage (airmail overseas) and instructions

PLEASE MAKE CHEQUE OR MONEY ORDER PAYABLE

***
*** "THE ASSOCIATION OF COMPUTER EXPERIMENTERS",
*** c/o Bernie Murphy,
***
*** 102 McCraney St.,
*** OAKVILLE, Ontario, Canada.
*** L6H 1H6
***

Non-Canadian residents can minimize order costs by sending a postal money order or if in the U.S.A., send us a personal cheque in US funds.

FIRST NAME

LAST NAME

FIRST LINE OF ADDRESS

SECOND LINE OF ADDRESS

CITY & PROVINCE

POSTAL CODE

MEMBERSHIP RENEWAL/CHANGE OF ADDRESS

The dues for the new club year, September 1980 to August 1981 are as follows:

Canada..... $15.00
Elsewhere..... $18.00 (Canadian)

Non-Canadian residents can minimize order costs by sending a postal money order or if in the U.S.A., send us a personal cheque for $15.50 in US funds.

These new rates include provision for FIRST CLASS POSTAGE which should ensure your copies of IPSO FACTO arrive as quickly as possible and in one piece! Renew your subscription now to ensure that you will continue to receive your IPSO FACTOs as soon as they roll off the press in the fall!!

PLEASE MAKE CHEQUE OR MONEY ORDER PAYABLE IN CANADIAN FUNDS AND SEND TO:

```
***
***  "THE ASSOCIATION OF COMPUTER EXPERIMENTERS",
***  c/o Bernie Murphy,
***  A.C.E.,
***  102 McCraney St.,
***  OAKVILLE, Ontario, Canada.
***  L6H 1H6
***
```

ADDRESS CHANGE: ___ NEW SUBSCRIPTION: ___ RENEWAL: ___

FIRST NAME

LAST NAME

FIRST LINE OF ADDRESS

SECOND LINE OF ADDRESS

CITY & PROVINCE

POSTAL CODE

TELEPHONE NUMBER

ATTACH OLD MAILING LABEL HERE FOR CHANGE OF ADDRESS