DOCUMENTATION

Contents

Please let us know if you find any errors in this documentation.
Thank you.

August 1983

SLICER Computers Inc.
2543 Marshall Street NE
Minneapolis, MN 55418
612-788 9481

Notice for SLICER Kit Builders

You may be an experienced kit builder and you may intend to ignore our assembly instructions. OK, the SLICER is really quite easy to build and there is no reason why you should not succeed.

However, we strongly recommend that you read the sections on "DRAM Sockets" and "Inserting and Testing of the RAM Chips".

We have found the procedures in those sections very helpful for a trouble free assembly of the SLICER and we believe that they will be helpful to almost anybody.

The Bugs in the 80186

There are several bugs in the 80186. Only one of them affects an existing 8086 instruction: IDIV gives incorrect results if the divisor is a memory reference and its value is negative. Obviously, this can affect existing software.

Intel expects to have these bugs fixed by 1984, until that time the 80186 is sold as is and Intel does not intend to exchange the present version of the 80186 for the later version.

We at SLICER have to follow Intels lead in this matter.

Using the A Version of the 80186

The first samples of the 80186 (the A version) did not have working counter/timers. The present monitor uses a timer only for turning the disk drives off. All the other functions of the SLICER will work with this version.
We experienced one problem with the reset circuit: It does not work at power up. The fix: remove the reset capacitor C52 and apply a manual reset.

Cooling the 80186

The 80186 runs very hot. Make sure that the SLICER is mounted so that the airflow to the 80186 is not restricted or use a fan to cool it.

Mounting the SLICER

Note that the fat conductor trace surrounding the SLICER board on top carries +5 volt and the one underneath is ground. Take care not to short these traces when mounting the SLICER.

8080 Simulator

We found an 8080 simulator which works very nicely. It is called SIM80 and is by

Northwest Microsystem Design
P.O.Box 10853
Eugene, OR 97401
(503)484 7129

The price is $50 (this is not a binding quote!). It is written in ASM86 and it comes with the source code. It makes it possible to run CPM80 programs written in 8080 code on the SLICER.

SLICER Support

Dave Thompson of Micro Corncopia has agreed to support the SLICER. Dave is running a user group for single board systems like the Big Boards from Digital Research Computers and Caltex and he publishes Micro Cornucopia, a bimonthly magazine. The address is

Micro Cornucopia
P.O.Box 223
Bend, OR 97709
(503)382 8048

Silk Screen Errors and Modifications on REV B

U2 is a 74LS04, not a 74S04.
U26 is a 74LS32, not a 74S32.
U33 is a 74S240, not a 74LS240.
The assembly drawing shows the correct parts.

All the boards of REV B level come with some modifications installed. The insulating sleeve used for this will not melt when touched with the soldering iron.

Copyright Notice

Installing CPM86 on the SLICER

CPM86 must be installed, that is customised to fit the SLICER, before it can be run. The installation process takes our BIOS and combines it with the CPM86 system from Digital Research to create a loader and a CPM systems file. The loader is located on the first two reserved tracks of the systems disk. The boot command of our ROM monitor gets the loader into memory and executes it. The loader then finds the CPM.SYS file and loads it.

The SLICER needs presently an 8" drive for drive A.

To install CPM86 you will need a CPM86 disk with the following files: LDCPM.H86, LDBDOS.H86, CPM.H86 and GENCMD.COM. You need further a CPM2.2 system which can read your CPM86 disk and which can read and write 8" single density disks.

You need from our disk the files LDBIOS.H86, TXBIOS.H86 and SYSGEN.COM.

On the CPM2.2 system the files PIP.COM and DDT.COM should be available.

Get all those files on one disk and have an empty single density 8" disk ready. This will be your CPM86 systems disk.

First create the loader:

```
PIP LOADER.H86=LDCPM.H86,LDBDOS.H86,LDBIOS.H86

GENCMD LOADER 8080 CODE[A500]

DDT
ILOADER.CMD
R800
^C

SYSGEN
```

Skip source drive with a <ret>, put your systems disk into the destination drive and enter the destination drive.

The SYSGEN program on our disk works with most systems. Let us know if you have any problems. We will help you to get going and install your system for you if necessary.

Now the Systems file:

```
PIP CPMX.H86=CPM.H86,TXBIOS.H86

GENCMD CPMX 8080 CODE[A80]
```

Assuming that your systems disk is in drive B:

```
PIP B:CPM.SYS=CPMX.CMD
```

You have now a systems disk for the SLICER.

The systems disk can be removed after CPM86 has been loaded. You can now insert disks with double density into drive A and warm boot them with ^C.

You can of course modify the BIOS to fit your needs. The source code is on our disk and it's comments should help you. The INSTAL.DOC file on our disk gives you more information for modifying the BIOS.

SLICER PARTS LIST

MANUFACTURER	PART NUMBER	DESCRIPTION	QUANTITY
INTEL	80186	MICROPROCESSOR	1
TEXTOOL	268-5400-00	80186 SOCKET	1
TI	TMS4500	RAM CONTROLLER CHIP	1
SIGNETICS	SC2681CSN40	SERIAL PORT CHIP (2 PORTS)	1
SMC	FDC9229BT	DATA SEPARATOR CHIP	1
Western Digital	FDC1797P	FLOPPY DISC CONTROLLER	1
SMC, Siemens			
INTEL	2732A	300nsec EPROM	2
TI	4164	DRAM 150nsec	16 or 32
INTEL	8283	SASI INTERFACE BUFFER CHIP	1
	74LS04		3
	74HC04		1
	7406		1
	74LS32		3
	74LS74		1
	74S240		1
	74LS240		2
	74LS244		1
	74LS245		4
	74LS259		1
	74LS373		3
MOTOROLA	MC1488		2
MOTOROLA	MC1489		2
M-TRON	MP1 16MHz	CRYSTAL	1
NDK	NDK037	CRYSTAL 3.6864MHz	1
ANSLEY	PDS-16	64K DRAM DOUBLE DECK SOCKET	16
AUGAT	240AG39D	40 PIN IC SOCKET	3
BURNDY	DILB28P-108	28 PIN IC SOCKET	2
AUGAT	220AG39D	20 PIN IC SOCKET	13
BURNDY	DILB16P-108	16 PIN IC SOCKET	1
AUGAT	214AG39D	14 PIN IC SOCKET	13
BOURNS	4308R-104-221/331	SIP 8 PIN TERMENATOR NETWORK	1
CENTRAL LAB	HC2110103G	SIP 10 PIN 10K W/1 COMMON	2
DALE	MSP06C03-473	SIP 6 PIN W/3 47K RESISTORS	1
SPRAGUE	420CF101X2PD	SIP 6 PIN 10K W/1 COMMON	1
	1N914	DIODE	2
KEMET	C320C104K5U5CA	.1mfd BYPASS CAPACITOR	29
SPRAGUE	199D106X9010B1	10mfd 10V(MIN) TANTALUM RADIAL	6
CENTRAL LAB	CN15A220K	220pf 50V(MIN) CERAMIC CAP	14
CENTRAL LAB	CN15C101K	100pf 50V(MIN) CERAMIC CAP	3

NOTE: ALL CAPACITORS HAVE RADIAL LEADS WITH .1 LEAD SPACING

EECO	230057GB	WRITE PRECOMPENSATION SWITCH	1
WEIDMULLER	1120.6	POWER CONNECTOR MALE	1
WEIDMULLER	5392.6	POWER CONNECTOR FEMALE	1
AP PRODUCTS	929836-01	DOUBLE ROW HEADER PINS	9
AP PRODUCTS	929834-01	SINGLE ROW HEADER PINS	1
	33 ohm 1/4W 5%	CARBON FILM RESISTORS	2
	4.7 kohm 1/4W 5%	Carbon film resistor	2
	10 kohm 1/4W 5%	CARBON FILM RESISTORS	4

SLICER ASSEMBLY INSTRUCTIONS

You have purchased a very powerful computer kit and are now staring at a pile of parts wondering "where do I start?". Well, have no fear, these assembly instructions are here to guide you step-by-step through the assembly of this kit. Although we cannot guarantee that if you follow these instructions you will have no problems, we feel that these instructions will make the assembly easier and the results more pleasing.

Before you fire up your soldering torch and blaze away you should obtain all of the items on the following list of assembly tools:

Soldering Iron.....25-40 watt with fine tip!
Rosin core electronic solder
Volt/Ohm meter
Component Lead Trimmer
Longnose Pliers
Exacto Knife
Small Screwdriver

Also useful:

A flat piece of soft 1" thick foam at least as large as the SLICER. Put the printed circuit board on top of it. It is now easy to insert components with long leads: the leads just penetrate the foam.

And:

A flat piece of wood, metal or other material at least as large as the SLICER. Put it on top of the components when you are ready to turn the board over for soldering.

In addition to these tools you will also need certain parts of your system to complete the assembly of the SLICER. They are:

Power supply....+5V @ 3A with OVP!
+12V @ 60ma
-12V @ 60ma
Serial terminal.RS232C,300 to 38.4K baud
Disk drives.....5.25" or 8"
Complete parts for SLICER

Bare Board Checkout

Begin your assembly with a careful visual inspection of your board. Look for shorts and broken traces checking with the Ohm meter if in doubt. If needed, consult the schematics to verify any connection. Once your visual check is complete, use the ohm meter to check that there are no shorts between any of the power lines by measuring the resistance between the power input connections at the location for the power connector-J8. These checks may seem trivial, but might save much grief later!

Diodes, Resistors

The assembly should start with the parts with the lowest profile. This would be the diodes and resistors.

[] Insert diodes: 1N914 or similar type. OBSERVE POLARITY! !
   Locations: D1,D2

[] Insert discrete resistors: 33 ohm (orange, orange, black)
   Locations: R6,R7

[] Insert discrete resistors: 4.7k ohm (yellow,purple,red)
   Locations: R2,R3

[] Insert discrete resistors: 10k ohm (brown, black, orange)
   Locations: R4,R5,R8,R9

[] Solder and trim components

Sockets

When installing IC sockets, it is important to match the socket orientation with that of the circuit board silkscreen legend. It is also important to use good soldering techniques to avoid solder bridges and cold solder joints, both of which will provide headaches later. Try not to use too much or too little solder. If you are in doubt as to your ability to solder, please seek out a local electronics guru or friend for some tips on the art.

Should you use sockets? Opinions vary wildly. The full SLICER kit comes of course with sockets for all ICs, but should you use them?
Consider this: Sockets are marvelous if you have a problem, suspect an IC and want to remove it. At that moment you are grateful for the presence of a socket, in particular if changing the IC solves the problem.
What if you remove an IC and find that a pin was bent and not inserted? You found the problem of course, but are you still grateful, particularly if you spend hours tracing the fault? This particular problem seems to occur more frequently than the problem of bad ICs.

Sockets can help to solve problems, but they also create problems.
I compromise: I use sockets for the large and expensive ICs .
All other ICs I solder into the board.
Keep one thing in mind: If you ever have to remove a soldered IC,
do not go to any great length to remove the IC undamaged, it is
preferable to destroy IC and not the board in the removal process.

These are the sockets I would use and which should be inserted now:

[] U6,U7 EPROMS

[] U18 SC2681 serial interface

[] U25 TMS4500 DRAM controller

[] U30 FDC9229 floppy disk interface

[] U32 FDC1797 floppy disk controller

So you want to use sockets for all ICs? In this case insert now:

[] 20 pin sockets: U1, U4, U5, U8...U12, U23, U24, U33, U34

   WATCH OUT for U23: 2 feed tru holes are next to pins 1 and 20 of
   this IC and it is quite easy to insert this socket wrongly.

[] 16 pin sockets: U29

[] 14 pin sockets: U2, U3, U13, U14, U15, U19...U22, U26, U27, U28, U31

Note: There is no U17.

Put your flat piece of material on top of the assembly to stop the
sockets from falling out, turn the board over and solder the sockets.

Soldering the ICs

Now is the time to solder the ICs if you choose that route. Insert the following ICs:

[] 74LSØ4 U2, Ul3, Ul5

[] 74HCØ4 Ul4 This is a CMOS device and particularly sensitive to static discharge!

[] 74Ø6 U31

[] 74LS32 U3, U26, U28

[] 74LS74 U27

[] 74S24Ø U33 Make sure you are not using a 74LS24Ø!

[] 74LS24Ø Ul, U5

[] 74LS244 U34

[] 74LS245 U8, U9, U23, U24 Watch out for U23: 2 feed thru holes are next to its pins 1 and 2Ø and it is quite easy to insert the IC wrongly.

[] 74LS259 U29

[] 74LS373 UlØ, Ull, Ul2

[] P8283 (i8283) U4

[] 1488 (75188) U2Ø, U22

[] 1489 (75189) Ul9, U21

I solder ICs so that their pins are flush with the lower side of the board. This makes it easier to peek underneath them and they can now be inspected from both sides of the board.

Capacitors

[] 220pF (221) C9   above U14

[] 100pF (101) C7, C8, C10

[] 10uF tantalum

Observe the polarity of the tantalum capacitors. Wrongly connected tantalums can provide quite spectacular fire and smoke displays.

C5 next to C7
C30 below J1
C51 left of power connector J8
C52 above power connector marking "RES"
C53 between U28 and R8

[] 220 pF (221) C16...C23, C25, C26, C27, C29

[] 0.1uF (104) C1...C4, C6, C11, C15
C24, C28, C31...C50, C54, C55

Note: there is no C14.

CPU Socket

[] The Square socket for the 80186 U16.
There should be an orientation pin on its lower left corner and a matching hole on the pc board.

Resistor Packs

All resistor packs except RP5 can easily be replaced by discrete resistors.
RP1 and RP2 are nine 1k ohm resistors each with one common end connected to +5V (pin 1).
RP4 consists of six 10k ohm resistors with one common end to +5V (pin 1).
RP3 consists of 3 individual 47k ohm resistors connected to adjacent holes.

[] RP1, RP2 adjacent to J1, SASI connector

[] RP3 (only marked as 47k) below C8,C9,C10.
Some crystals interfere with RP3. Check this and use 3 individual 47 kOhm resistors if this is the case.

[] RP4 above U18 (a 40 pin socket)

[] RP5 next to J6, the 5.25" floppy connector

Connectors, Jumpers

5 dual in line headers with 72 pins each are supplied.
They have to be cut to length with an Exacto knife. Headers can be made up from smaller pieces, so don't despair if you make a mistake when you cut them.

[] J1, trim to 50 positions.

[] J2, trim to 40 positions. Trim the remainder for

[] J4, 26 positions

[] J3, trim to 50 positions

[] J6, trim to 34 positions. Trim the remainder for

[] J5, 26 positions.

[] J7, trim to 50 positions.

[] Cut from the remaining pieces seven 6 pin headers and install them as JB2, above of RP4 and as JB3..JB8, next to J4.

[] Install a 3 pin single in line header as JB1 between U7 and U10.

[] Power connector J8, the 8 pin orange contraption.

Switch, Crystals

[] Install SW1 near J8

[] Install CPU crystal: 16MHz or optional lower frequency
Location: X1

Notes for installation of optional lower frequency crystal:

A lower frequency crystal may be used if the CPU is not rated for 8MHz operation and/or if the dynamic RAM devices are slower than 150nsec access time. If this is done, the following modifications MUST be made:

1. 16MHz crystal must be installed in location X3.
2. 100pf capacitors must be installed in locations C12 and C13.
3. Jumper trace X16 near U14 pin 1 must be cut.
4. Slower crystal may now be installed in location X1.

[] Serial interface crystal X2, 3.6864 MHz

DRAM Sockets

[] U40...U55

The piggy back sockets for the RAM have two offset pins for seperate RAS and CAS signals. These two pins can easily be pushed up when the socket is inserted. Try to avoid this - do not force the sockets into their holes.
Check the alignment of the pins if you have problems and straighten them if necessary. Better yet, check the alignment before you have any problems.
The offset pins can quite easily be pulled down again if they have been pushed up. Note however that the tops of these pins must be inside of the golden colored pockets for the top RAMs. These pockets are sometimes pushed to the outside and show very little opening. The opening can be restored by pressing on the back of the pockets from the outside of the socket.
Tag solder only two corner pins of the sockets after they have been inserted. Then check that all pins are properly seated and correct any problems. Now solder all the pins.

<<<<<<<<<<<<<<<<<<<<<<<<<< S T O P >>>>>>>>>>>>>>>>>>>>>>>>>

DO NOT INSERT ANY COMPONENTS BEFORE PERFORMING THE FOLLOWING CHECKS!!!!

Circuit Board Test With Power Supply

Begin this portion of the assembly procedure by wiring the power supply according to the manufacturer's recommendations. Attach the correct power supply outputs to the screw terminal portion of connector J8 and unplug the connector from the board. The power inputs are marked on the silkscreen and are as follows from left to right:

-12 volt, ground, +5 volt, +12 volt

There are four more terminals on J8: the reset switch has to be connected between RES and the adjacent ground. SR and the adjacent +5 is intended for a solid state relay which will control the motors of 8" disk drives.

Now you are ready for your first smoke test. Check your power supply connections again and then plug the power supply into a wall outlet and check that the voltages on J9 are what they should be. NOTE: If you are using a switching power supply the voltages may not regulate properly when they are under no load. In most cases a load is needed on the 5V output to make the supply regulate properly and this may be done by using a resistor (about 100 ohm) between +5v and ground. RP5 provides a load of 100 ohm, but make sure that this is sufficient before you connect the power supply.

Unplug your supply from the wall outlet and plug J8 into the circuit board. Be sure that you have aligned it properly. Now you are ready to apply power to the SLICER for the first time. Get the fire extinguisher out and put on your saftey goggles, or better yet shield yourself behind a 4 foot thick concrete wall. Or if you have been a good assembler forget this nonsense and continue: plug the power supply into the wall outlet and verify the voltages on the board.

NOTE: Never poke an oscilloscope probe or similar device into an IC socket. It will be destroyed by this and definitely fail when you are about to save a days work on disk. This in turn leads to grey hair, tantrums and alcoholism.

[] With the - lead of your voltmeter connected to J8-ground, measure the voltage at the socket for U32 pin 20. It shold read zero volts.

[] Measure the voltage at U32 pin 21. It should read 5 volts.

[] Measure the voltage at U32 pin 40. It should be +12 volts.

[] Now with the + lead of the voltmeter attached to J8-ground, measure the voltage at U20 pin 1. The voltmeter should read +12 volts when measured in this way (that means on pin 1 is -12 volts!).

If you have any doubts, repeat this section!! Incorrect power supply connection will cause the destruction of many dollars worth of components!!!!!

IC Insertion

A major source of problems with sockets is bent IC legs. They are difficult to spot and often make initial contact only to fail at the most awkward moment. The threat of alcoholism looms! Therefore make sure that the pins of the ICs are aligned. Adjust them to proper spacing (ICs usually come with their pins spread apart). You may want to open up the contacts of the sockets: CAREFUL! Use only a tool like a needle with a diameter not larger than an IC leg. Push it into the contacts of the sockets without overstressing them. This will also show you if there is too much solder in a contact. The socket has to be removed if that is the case. The plastic body of some type of sockets can be removed from the contacts. Then things are easy: Remove the contacts individualy, clean the holes and solder in a new socket, and be a bit more careful next time around!
Crunch the plastic body up if it does not come off. Remember, destroy the socket, not the board!

Insert the ICs gently. If it takes too much force you are probably bending something. Check what is going on, maybe a contact is soldered shut.

These are the ICs which you may have soldered in:

[] 74LSØ4 U2, U13, U15

[] 74HCØ4 U14 This is a CMOS device and particularly sensitive to static discharge!

[] 74Ø6 U31

[] 74LS32 U3, U26, U28

[] 74LS74 U27

[] 74S24Ø U33 Make sure you are not using a 74LS24Ø!

[] 74LS24Ø U1, U5

[] 74LS244 U34

[] 74LS245 U8, U9, U23, U24

[] 74LS259 U29

[] 74LS373 U1Ø, U11, U12

[] P8283 (i8283) U4

[] 1488 (75188) U2Ø, U22

[] 1489 (75189) U19, U21

The Big Buck ICs

[] FDC9229 U30

[] SC2681 U18

[] FD1797 U32 Make sure you get this right!
There is +12V on this socket and this
would in all likelihood destroy the
the TMS4500 and the SC2681 !

[] TMS4500 U25

[] Insert SLICER monitor ROMs. Note that the 2732 type EPROMs have only 24 pins and the sockets have 28 (this allows the installation of larger ROMs like the 2764 or 27128). The EPROMs are to be installed so that pins 1,2,27 and 28 of the sockets are empty.

SLICER.odd Location: U6
SLICER.evn Location: U7

[] Jumper JB1 for correct EPROM size: 4k for 2732.

[] Install 80186 CPU. Remove cover from CPU socket and correctly orient the CPU chip before closing the lid.
The chip will only go in one way!
Note that the flat side of the CPU is up when the CPU is correctly installed. Close the cover of the socket.

[] RECHECK YOUR WORK!!!!

<<<<<<<<< STOP! DO NOT INSTALL DRAM CHIPS AT THIS TIME!!! >>>>>>>>>>>>

Console Connection and Initial Checkout

If you have now completed everything to this point you should be ready to bring your SLICER to life. Soon you will have millions of electrons pulsing at your command as the power flows from your fingertips!

Note that we have quite intentionally left the DRAM chips off the board. The SLICER will come to life without them thanks to the superb Level I monitor created by Otto Baade and Larry XXXXX.

Connect an RS232C serial terminal to serial port A (J4). Jumper JB3,JB4 and JB5 to match your terminal. The "modem" configuration is likely to be the right choice. See sheet 4 of the schematics.
You have the choice of one of the following baud rates for you terminal:

300
600
1200
2400
4800
9600
19200
38400.

At this time your adrenaline should be pumping and your heart should be beating in anticipation of seeing your work come to life. Connect the power supply correctly and apply the power, Typing a carriage return on your terminal should allow the SLICER to determine the terminal's baud rate and put you in the RAM debug monitor.

If for some reason nothing happens immediately remove power and check the following areas:

1. Incorrect components.....recheck IC installation
2. MON186 EPROMS incorrectly placed...correct them.
3. No jumper on JB1....jumper according to EPROMs
4. RS232C terminal incorrectly connected..recheck!
5. Terminal baud rate incorrect...change it!
6. IS YOUR POWER SUPPLY CORRECT?????

The chances are that if you followed the instructions carefully your SLICER will now be running properly. Read "Bringing UP the SLICER" if it is not working.
It is working? Great, then it is time for

Inserting and Testing of the RAM Chips

The following tools are useful at this point: A side cutter, a pair of longnose pliers with narrow jaws of 2" length and a needle, an Exacto knife with a pointed blade (#11) or a nail with a point filed or ground to the shape of an IC leg. (Use a 16 penny nail for this.)
The legs of the RAM chips should be trimmed by 1/16 of an inch. This will seat the lower bank as low as possible and thus keep the legs of the upper RAMS from causing shorts between lines. I found that this particular step avoids a lot of problems with the RAM and I highly recommend it.
Now grip a RAM lengthwise with the pliers and work it into the lower position of the socket for U40/U60. This is actually U40. It may be necessary to push the golden colored pockets out of the way, particularly if you are using ceramic RAMs.
Next insert U41,42,43.
Now comes the one truly tricky spot in the assembly process: The insertion of the upper memory bank. Patience, a sharp eye, a steady hand and patience are invaluable at this point.
First make sure that the clamping cams (the comblike parts on top of the sides of the sockets) are in their upper position. Now check the golden insertion pockets for the upper chips. Use the pointed tool of your choice to pull them out of the wall of the socket without deforming them. Grip the RAM for U60 with the plier and carefully work its legs into the pockets. Inspect your work carefully by peeking underneath the chip. When you are satisfied that all legs are in their proper place press the RAM down with moderate force . You will deform the pockets and probably cause shorts to the lower RAM if you press too hard. Press the insertion cams down. Repeat for IC61,62,63.

Now the least significant nibble for both banks is installed and it is time for a test. Turn on the power, enter <cr>. This gets you the sign on message of the RAM test monitor. Then type S<cr>. This selects the first 64k of RAM. Now hit T (the RAM test). After a short time the error display will begin. Stop the display by hitting any key. You can now see the address offset within the selected segment, the data from the RAM and the test data. The least significant digit of the data words will be equal if there is no error.

>>>>>> Turn off the power before inserting or removing or wiggling
>>>>>> any RAM chip.

If there is no equality between the two least significant nibbles try to find out if a particular bit causes the error. This points to a faulty or badly seated RAM chip.
Shorts between pins will generally affect all bits equally. This is the most likely error to occur: the pins of the upper chips can easily be bent sideways and this is often not visible. Remove the upper chips one at a time and see if any leg is bent. It is not necessary to lift the insertion cam in order to remove a chip.

Exit from the test by hitting <cr> if the nibble of the lower bank checks out. Select the upper bank by entering S2000<cr> and test it by entering T.

>>>>>> Turn off the power before inserting or removing or wiggling
>>>>>> any RAM chip.

Repeat the above procedure for each nibble. The sign on message should change after you have inserted the last nibble, it should say now among other things:

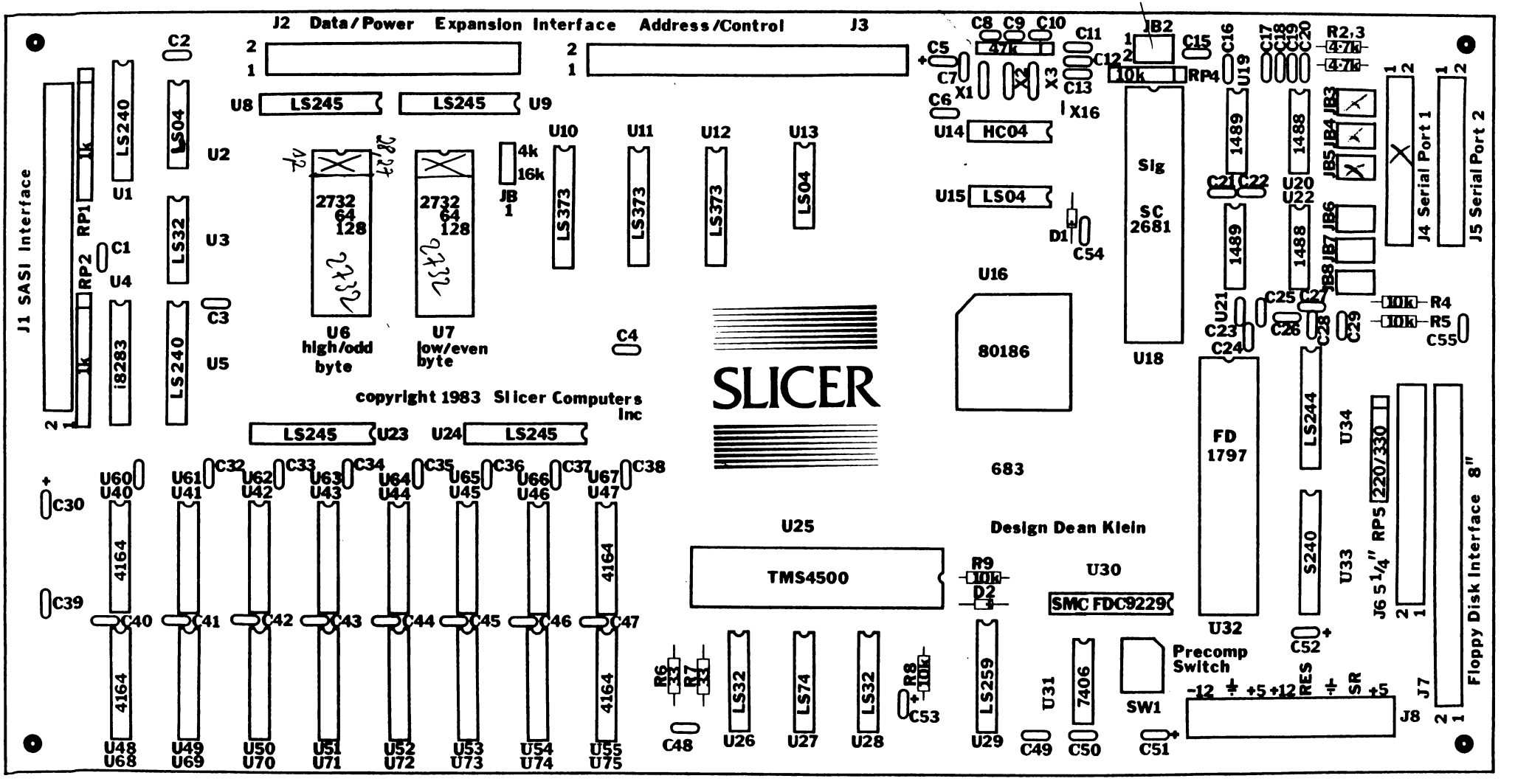
256k RAM installed

The initial RAM test was succesful and you are now in the main monitor. It may indicate only 128k of RAM. This means that there is a problem with the upper bank. At any rate, MT<cr> will get you back to the RAM test monitor and you can chase the problem.
Run the memory test for the four segments for some time, say over night or while you are at work. Enter a S1800 for this. The test will now run with the last 32 kbytes of the lower bank and the first 32 kbytes of the upper bank. Further, bang the board around a bit while the test is running. This will show up bad or missing solder joints.

Congratulations.

[] Give yourself a pat
   Location: On your back.

Have fun with your SLICER!!!


J2 Data/Power Expansion Interface Address/Control J3
J1 SASI Interface
J4 Serial Port 1
J5 Serial Port 2
J7 Floppy Disk Interface 8"
J8
U1 LS240
U2 LS04
U3 LS32
U4 i8283
U5 LS240
U8 LS245
U9 LS245
U10 LS373
U11 LS373
U12 LS373
U13 LS04
U6 high/odd byte
U7 low/even byte
U14 HC04
U15 LS04
U16 80186
U18 Sig SC 2681
U19 1489
U20 1488
U21 1489
U22 1488
U23 LS245
U24 LS245
U25 TMS4500
U26 LS32
U27 LS74
U28 LS32
U29 LS259
U30 SMC FDC9229
U31
U32 FD 1797
U33 S240
U34 LS244
U40 U41 U42 U43 U44 U45 U46 U47 4164
U48 U49 U50 U51 U52 U53 U54 U55 4164
U60 U61 U62 U63 U64 U65 U66 U67 4164
U68 U69 U70 U71 U72 U73 U74 U75 4164
copyright 1983 Slicer Computers Inc
SLICER
683
Design Dean Klein
Precomp Switch
SW1
RP1 1k
RP2 1k
RP4 10k
R2,3 4.7k
R4 10k
R5 10k
R6 33
R7 33
R8 10k
R9 10k
D1
D2
JB1 4k 16k
JB2
JB3
JB4
JB5
JB6
JB7
JB8
RP5 220/330
J6 5 1/4"
-12 +5 +12 RES +5 SR +5
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11 C12 C13 C15 C16 C17 C18 C19 C20 C21 C22 C23 C24 C25 C26 C27 C28 C29 C30 C32 C33 C34 C35 C36 C37 C38 C39 C40 C41 C42 C43 C44 C45 C46 C47 C48 C49 C50 C51 C52 C53 C54 C55
X1 X2 X3 X16
47k
10k
2732 64 128

10"

The SLICER

Theory of Operation

INTEL has developed a marvelous piece of hardware: the 80186. This microprocessor provides a new concept of integration that makes it a designer's delight. The features of a standard microprocessor have been extended close to system level on a chip by adding DMA channels, interrupt controller, timers and programable device select lines. The execution speed of the 8086 has been considerably increased and finally, all this resides in a package with 68 connections taking less than 1 square inch of space.
The SLICER is a good example of the compact but powerful systems which can be designed with the 80186.

A detailed description of the design follows.

The SLICER circuitry is shown on four drawings:
- Page 1: CPU, ROM, Expansion Interface
- Page 2: RAM, Reset Logic, Power Connector.
- Page 3: Floppy Disk Controller, Interface to SASI Controller.
- Page 4: RS 232 Ports, 80186 Pin Out.

Let us begin with page 1.

CPU Clock

The 80186 divides the external clock by 2 for its internal operation and it provides this divided clock for the system. The oscillator is shown at A1 on page 1. An external oscillator is used so that its output can be used for the floppy disk controller on page 3. The oscillator uses a high speed CMOS inverter (U14-5/6). The diode D1 insures a symmetric wave form.
The SLICER normally runs at 8MHz, the crystal X1 has to be 16MHz. This crystal must be changed if a different clock frequency for the CPU is needed. A 16MHz clock is still needed for the floppy disk controller: cut jumper trace X16 and install X3= 16MHz, C12 and C13= 100pF.

Address and Data Bus

The 80186 multiplexes address and data onto the AD0..AD15 lines. The data lines of all on board peripherals are directly connected to these signals. The address is strobed by the ALE signal into the address latches U10...U12. The outputs provide then the required stable addresses for the peripherals and the expansion interface.
Note that the 80186 is a byte addressed CPU. The bus however is organized as a 16 bit word. A0 is as a consequence only used for selecting between even and odd byte (see under RAM).
Two buffers (U8,U9) are provided for the data lines of the expansion interface. The buffer enable (/BUFEN) and direction (/EXP IN) lines have to be controlled by the expansion units.

Important Control Signals

/RES Power on reset (generated by D2,R9 and C52) and reset key from J8 (page 2, A/B2).
/UCS Upper Chip Select. Used to enable ROM.
/LCS Lower Chip Select. Used to address DRAM controller TMS4500 (U25).
/PCS0 Peripheral Select 0. Used to address floppy disk controller 1797 (U32).
/PCS1 Used to address serial interface controller SC2681 (U18).
/PCS2 Used to address floppy drive selection LS259 (U29).
/PCS3 Used to address SASI port (U1, U4, U5).
DT/R Data Transmit/Receive. High for write, low for read.
/DEN Data enable. Indicates that the address/data bus is ready for data.

ROM

Three types of ROM can be used: The 2732 for 8k bytes, the 2764 for 16k bytes and the 27128 for a total of 32k bytes of ROM.
Jumper JB1 determines the selected ROM size:
Position 4 for 2732, position 16 for 27128.
The jumper position is irrelevant for 2764's.

/UCS is programmed for the full address range of 32 KBytes of ROM, beginning at 0F8000h. That means that the memory image of smaller ROMs appears several times within this address space.

RAM (page 2)

The RAM is organized in two banks of 64k words each.
The RAM chips used are 64 kbit devices of the 4164 type. Note that the IC numbering for the RAM chips coincides with the bit numbering:

Low bank U40...U55 --> bit 0...bit 15
High bank U60...U75 --> bit 0...bit 15.

Bank selection is accomplished with the /RAS0 and /RAS1 from the RAM controller. It is possible to read and write individual bytes. This is done by having a /LOWCAS and an /HIGHCAS signal. These signals are controlled by /CAS from the controller, A0 from the address latch and /BHE (bus high enable) from the CPU.

A0=0 and /BHE=0 --> access to low and high byte (word transfer)
A0=0 and /BHE=1 --> access to low byte only
A0=1 and /BHE=0 --> access to high byte only

The DRAM controller (U25) interfaces the RAM to the CPU. The controller used in the SLICER is a TMS4500 from Texas Instruments. This device seems to be designed for the 80186, it makes it possible to realize the speed potential of the 80186 with dynamic RAM.
Here are some of the features which make the 4500 so attractive:

- It uses the 8MHz clock from the 80186. This syncronous operation makes it quite easy to see with an oscilloscope what is going on. (No signal jitter as on controllers with their own clock).
- It multiplexes the 16 bit address from the CPU onto the 8 address lines of the RAMs.
- It generates refresh signals (/RAS0 and /RAS1) for refresh and an 8bit refresh address.
- It allows RAM operation at 8MHz without mandatory wait states.
- It arbitrates between memory requests from the CPU and refresh requests from it's internal refresh generator and issues a wait request to the CPU when a contention occurs.

A memory access goes thru the following sequence:
The trailing edge of the clock arbitrates between memory request and refresh request. ALE=1 and /CS(/LCS)=0 initiates the access cycle. The address byte on the row address inputs (RA0...RA7) prpagates to the address outputs (MA0...MA7). The trailing edge of ALE latches the 16 address bits into the controller and causes /RAS0 or /RAS1 (depending on the state of A17) to go low. This strobes the current address on MA0..MA7 into the RAMs. MA0...MA7 will be switched to the address byte on inputs CA0...CA7 after either /RD or /WR goes low. A short time later (short means here about 20 to 50 nanoseconds) /CAS goes low. The state of the /WR input (pin 3 on the RAMs) determines whether a write or read cycle is performed. Data will appear on the DO pins of the RAMS (read cycle) or the data on the DI pins will be written into memory.

Refresh cycles generate /RAS signals only. The controller will request a wait state from the CPU by pulling its ready line (pin 2) low if a memory request occurs during a refresh cycle. This wait request is delayed by the LS74 flip flop U27 on page 1A2, so that the CPU can sample it on its ARDY input at the end of its t2 state.

The RAM data bus is isolated from the system bus (AD0...AD15) by two LS245 bidirectional buffers U24 and U25. This is not really nescessary, the SLICER works perfectly if the memory is directly connected. However, the buffers make the debugging of the RAM much easier since now shorts on the memory bus will not influence the rest of the system and the memory test in the monitor still work with bad RAM.

The SLICER is delivered with 150 nano second RAM chips. This leaves a good margin for safe memory operation, I have had a SLICER running at 8 MHz without problems with 200 nano second parts.

The TMS4500 can be programmed to insert automatically one wait state into every access. This will not make the SLICER run with slower RAMs since it does not change the critical /RAS to /RAS interval. The only way to run slower RAMs is to reduce the clock frequency.

Floppy Disk Controller (page 3)

The floppy disk controller can control both single and double density drives, 5.25" and 8", single and double sided, in any combination. In addition, most of the new micro disk drives are compatible with 5.25" drives, so they may also be used.
The controller can address either 4 drives with the standard radial select option or 8 drives with binary select option. The octal addressable latch LS259 (U29) controls the drive selection. It also controls the selection of drive size (MINI) and density (/DDENS).

The heart of the controller is the long proven 1797 controller (U32) plus a FDC9229B (U30) interface chip.
The 1797 performs most of the functions of interfacing to the disk drives: Head positioning, data formating, serial <-> parallel conversion, error checking and control for write precompensation. The FDC9229 handles the remaining functions: Data separation, head load timing and data shift for write precompensation. This device is some what a compromise design. It is completely digital and needs very little adjustment, only the write precompensation time must be set. It allows for easy switching between 8" and 5.25 drives and it allows a very compact disk controller.

However its data separator does not work as well as a well designed and adjusted analog data separator with a phase locked loop. This shows up on some 8" drives on the inner tracks with double density. Appendix A will deal with this in more detail.
SW1 allows a range of write precompensation times to be selected. This adjustment does not affect single density operation and it need not be changed once it has been set. Settings 2,3 and 4 are the most likely choices.
The following tables shows the settings of SW1 and the jumpers for people who do not have this switch.

Setting	8" Time	5.25"	Jumpers
0	0 ns	0 ns	1-5, 4-5, 3-5
1	62.5 ns	125 ns	1-2, 4-5, 3-5
2	125 ns	250 ns	1-5, 4-2, 3-5
3	187 ns	375 ns	1-2, 4-2, 3-5
4	250 ns	500 ns	1-5, 4-5, 3-2
6	312 ns	625 ns	1-5, 4-2, 3-2

Connecting Floppy Disk Drives

Two connectors are provided: J6 for 5.25" drives and J7 for 8" drives. These connectors are connected in parallel and certain restrictions apply for the mixing of drives of different sizes.
Floppy disk drives are connected in a daisy chain arrangement. The last drive in this chain should have a standard 220/330 Ohm terminator network on its inputs. The SLICER sits nomally on the other end of the chain and it has in turn this terminator on its inputs (RP5).
5.25" drives and 8" drives cannot be directly connected to the same cable, since they use different types of connectors. The best way to mix drives is by adapting 5.25 drives to the 50 conductor cable of the 8" drives via a small translator board. An 8" drive at the end of the daisy chain should provide the termination.
If the different drives are connected by using both connectors (J6 and J7) then two terminating networks have to be driven by the SLICER. Each termination requires 20mA drive current. There are two different driver ICs on the interface: U33, a 74S240 rated at 64 mA output current and U31, a 7406 rated at 40 mA. U31 would work outside of its worst case specifications since it has to drive the inputs to the drives in addition to the terminators, however it will in all likelihood work.
There is still another problem with this arrangement: The SLICER with its termination sits now in the middle of the cable. It is therefore necessary to make one cable as short as possible. This cable should only be connected to one drive only. The termination network on this drive should be removed.

SASI Port (Page 3)

The SASI (Shugart Associates Standard Interface) port is primarily intended for interfacing to hard disk controllers but it can be used for other purposes too. For example it should be possible to configure it as an parallel printer port.
U4 and U5 provide a bidirectional 8 bit data path. U1 buffers the control signals from the controller. The /RST signal from U29 is a reset signal. /SEL low (also from U29) indicates that the data on the data path is a device address.
The signal /REQ from the controller indicates that the controller is ready for a one byte data transfer. This signal is qualified by the signals /I-O, /C-D and /MSG. /ACK is the response from the SLICER to each /REQ.
The /REQ signal can be used to operate the second DMA channel. For this it is nescessary to jumper pins 23 and 24 of J3 (see page 1, section A3). These pins are marked by an arrow on the SLICER boards REV C.
The SASI port of the SLICER has the following limitations: The terminator resistors are 1 k Ohm pull ups, not the standard 220/330 Ohm networks.
The outputs of the /RST, /SEL and /ACK signals can not drive the standard 220/330 Ohm termination network.
This makes it necessary to remove this network on the controller and to keep the connecting cable to the controller short.
The /ACK signal should be 750 nsec long. This requires 3 wait states for the I/O lines. The /PCS lines 0 - 3 are programmed for this.

Serial Communications (page 4).

The Signetics SC2681, U18, forms the core of the serial communications system. This circuit has two complete asynchronous serial channels, two baud rate generators programmable from 50 baud to 38.4k baud, 8 general purpose outputs and 7 general purpose inputs.
IP0 - IP3 and OP0 - OP3 are used as handshaking signal for the serial ports. OP7 controls the motor signal for the floppy disk drives and IP4 - IP7 are connected as sense inputs to jumper JB2 and can be used to select software options.
Serial port 1 (J4) is dedicated as the console port and uses the RS232 transmitter/receivers U20 and U19. Jumpers JB3 - JB5 allow the swapping of receiving and transmitting lines. This makes it easy to configure a port as "modem" or "terminal".
The connector J4 can be directly connected with a crimped flat cable to a DB25 connector.
Port 2 can be used as a printer or modem port. It has the same features as port 1.
The layout of the jumpers JB3 - JB8 on page 4 corresponds to the actual layout on the SLICER. This page also gives a translation of the pin numbering of the flat cable headers to the pin numbering of a DB25 connector.
The SC2681 requires a clock frequency of 3.6864 MHz, suplied by the oscillator consisting of X2 and U14 pin 3/4. This clock is also provided on the expansion connector J3-45 for use of additional serial ports.

Peripherals Addresses

The following list gives the address of the on board peripherals.

Floppy disk controller, PCS0:

Status register	0		
Command register	0		
Track register	2		
Sector register	4		
Data register	6		
Drive 0	10Ah	(pcs2)	bit0 = 1
Drive 1	108h		
Drive 2	106h		
Drive 3	104h		
Mini selection	10Ch		
Single density	10Eh		
Motor on	9Ch	(PCS1)	bit7 = 1 (don't change bit0..6)

Serial ports , PCS1:

Mode register 1	80h		
Status register 1	82h		
Baud rate select 1	82h		
Command register 1	84h		
Data register 1	86h		
Mode register 2	90h		
Status register 2	92h		
Baud rate select 2	92h		
Command register 2	94h		
Data register 2	96h		

SASI port, PCS3:

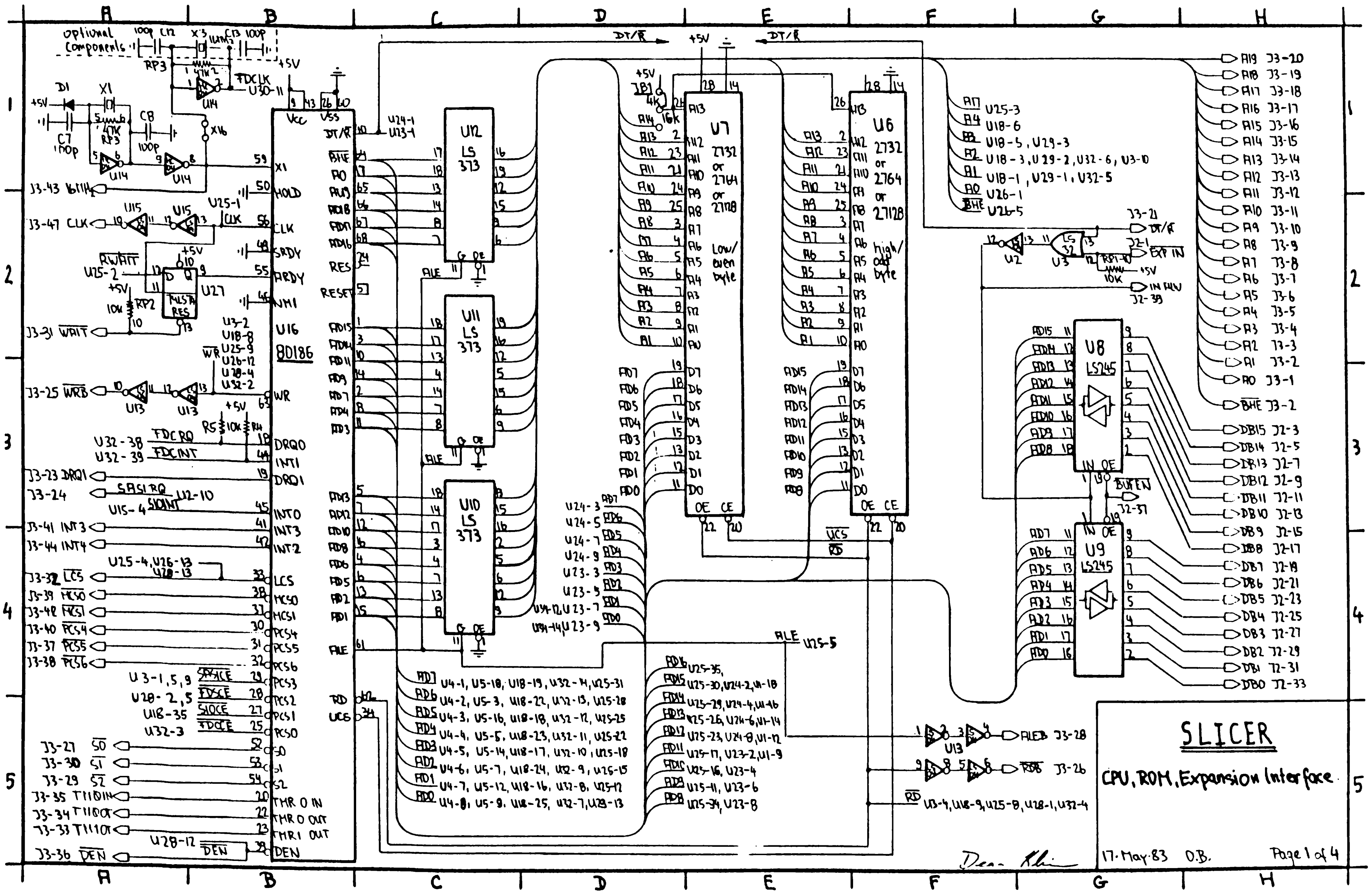
Read/write with /ACK	180h		
without /ACK	184h		
Select (/SEL)	100h	(PCS2)	bit0 = 0
Reset (/RST)	102h	(PCS2)	bit0 = 0

Data is read/written on the lower(even) byte, status appears on the higher(odd) byte.

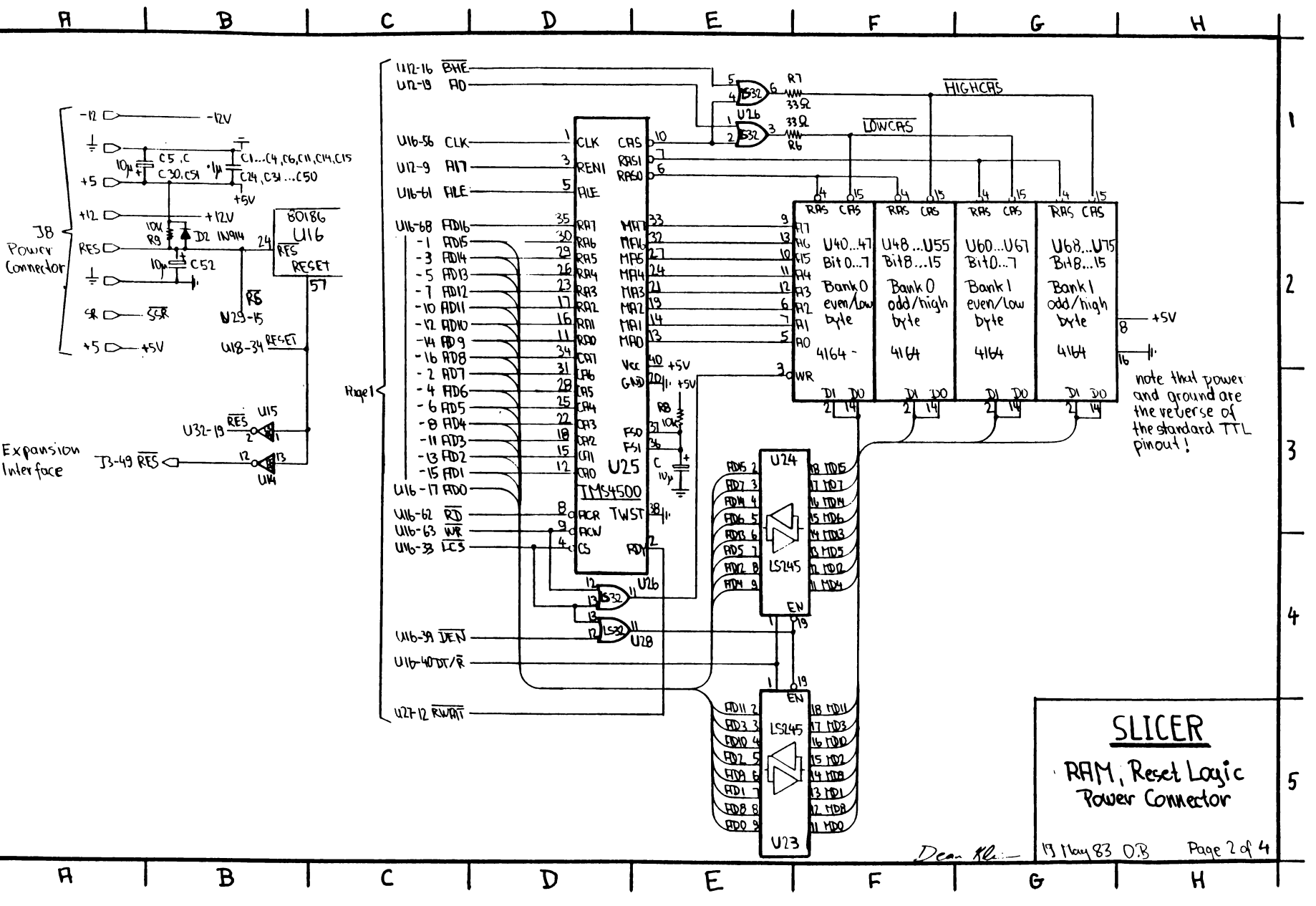
Appendix A

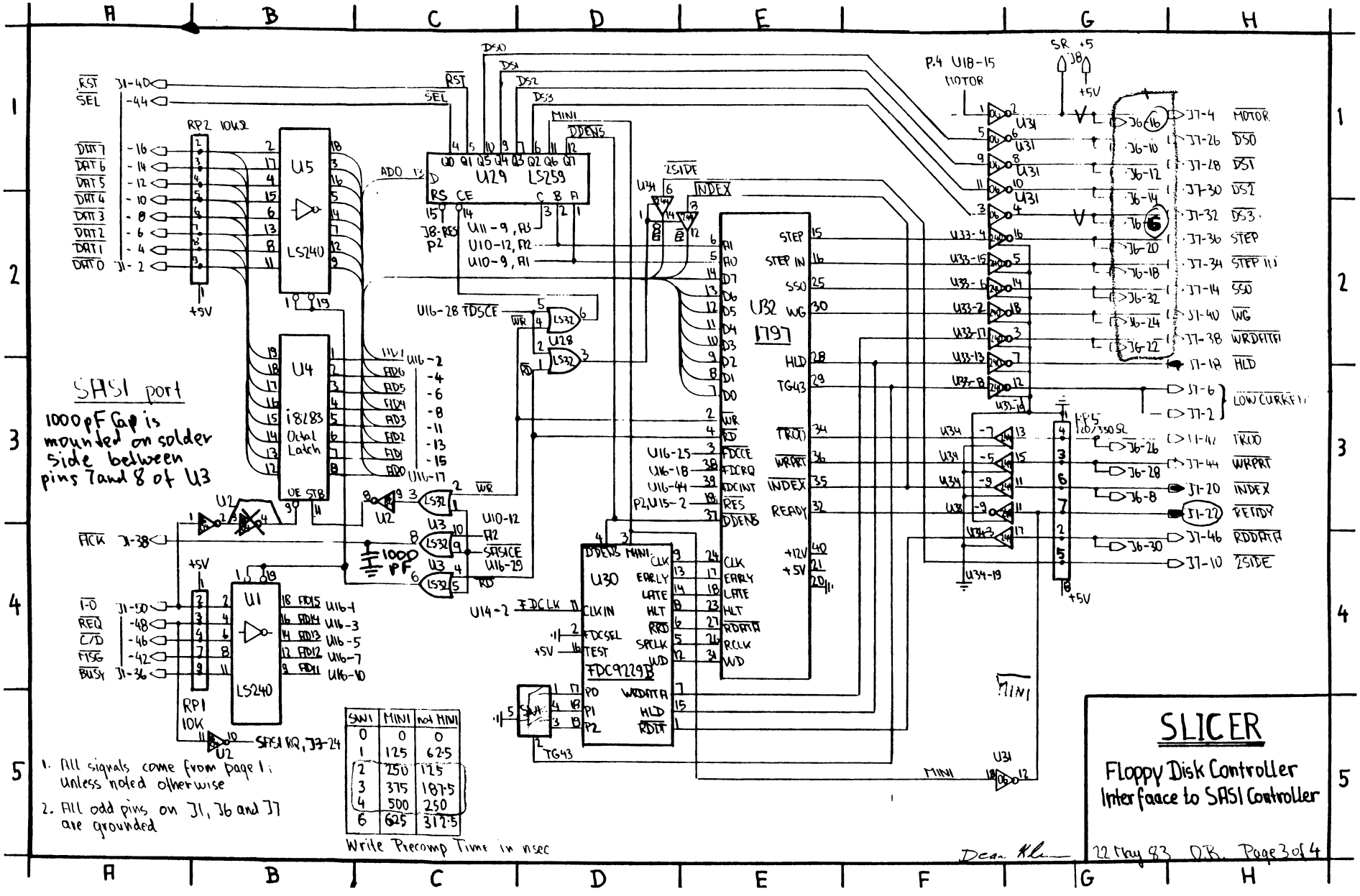
8" double density with the SLICER

The following is based mainly on investigations by Dennis Fox of Micro Time with the Big Board II from Caltex.
The BB II uses a FDC9216B by SMC for data separation. The FDC9229B on the SLICER incorporates the same data separation circuitry and it should therefore be safe to extrapolate from the BB II.
It was found that some types of drives do not work well on the BB II. They give read errors on the tracks above track 70. The drive in question were Shugart SA800 and DRI drives. The same drives would work well if the 9216 was replaced with an analog data separator.
Drives which work without problems are 8" Tandon slimline and Siemens FD100-8 (presently available for < $200 new).
The performance of a SA800 improved dramatically after it was cleaned, equipped with a new head load pad, and adjusted.
All drives worked well on single density.
5.25" drives seem to work without any problems at double density.
SMC indicated that the 9216 was originally designed for 5.25" drives and apparently it is successfully used on several computers, e.g. the Kaypro.
We at SLICER have not had the time for a detailed investigation of this problem. However we will update this note whenever new information becomes available and would appreciate feedback on this matter.

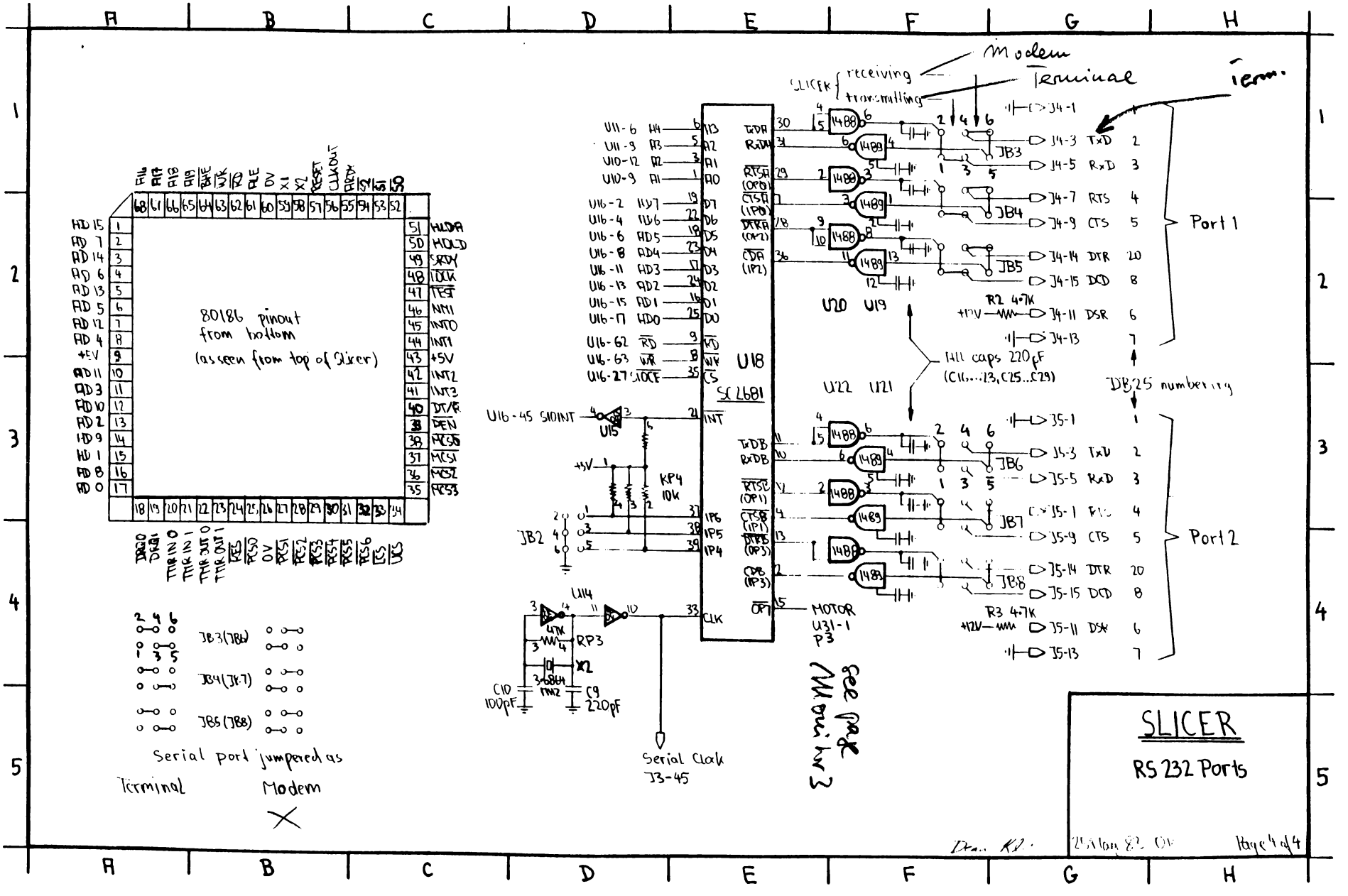


SLICER
CPU, ROM, Expansion Interface
17-May-83 O.B.
Page 1 of 4


J8 Power Connector
-12
-12V
+5
+5V
+12
+12V
RES
SCR
C5 ,C
C30,C31
C1...C4,C6,C11,C14,C15
C24,C31...C50
10µ
.1µ
R9
D2 1N914
C52
80186
U16
RESET
57
U29-15
U18-34 RESET
Expansion Interface
J3-49 RES
U32-19 RES
U15
U14
Page1
U16-56 CLK
U12-9 AD7
U16-61 ALE
U16-68 AD15
-1 AD5
-3 AD14
-5 AD13
-7 AD12
-10 AD11
-12 AD10
-14 AD9
-16 AD8
-2 AD7
-4 AD6
-6 AD5
-8 AD4
-11 AD3
-13 AD2
-15 AD1
U16-17 AD0
U16-62 RD
U16-63 WR
U16-33 LCS
U16-39 DEN
U16-40 DT/R
U27-12 RWAIT
U25
TMS4500
TWST
U26
U28
U24
U23
LS245
EN
U12-16 BHE
U12-19 A0
R7
33Ω
R6
HIGHCAS
LOWCAS
RAS CAS
U40...47 Bit0...7 Bank0 even/low byte 4164
U48...U55 Bit8...15 Bank0 odd/high byte 4164
U60...U67 Bit0...7 Bank1 even/low byte 4164
U68...U75 Bit8...15 Bank1 odd/high byte 4164
+5V
R8 10K
C 10µ
note that power and ground are the reverse of the standard TTL pinout!
SLICER
RAM, Reset Logic
Power Connector
19 May 83 D.B
Page 2 of 4



SW1	MINI	not MINI
0	0	0
1	125	62.5
2	250	125
3	375	187.5
4	500	250
6	625	312.5


80186 pinout
from bottom
(as seen from top of Slicer)
Serial port jumpered as
Terminal
Modem
JB3(7BW)
JB4(7B7)
JB5(7B8)
U18
SCL6891
U16-45 SIOINT
U15
+5V
RP4
10K
JB2
U114
4.7K
RP3
X2
3.6864
MHz
C10
100pF
C9
220pF
Serial Clock
J3-45
MOTOR
U31-1
P3
see page 3
SLICER
receiving
transmitting
Modem
Terminal
Term.
1488
1489
JB3
JB4
JB5
JB6
JB7
JB8
J4-1
J4-3 TxD 2
J4-5 RxD 3
J4-7 RTS 4
J4-9 CTS 5
J4-14 DTR 20
J4-15 DCD 8
J4-11 DSR 6
J4-13 7
Port 1
R2 4.7K
+12V
Hill caps 220pF
(C16...C13, C25...C29)
DB25 numbering
J5-1 1
J5-3 TxD 2
J5-5 RxD 3
J5-7 RTS 4
J5-9 CTS 5
J5-14 DTR 20
J5-15 DCD 8
J5-11 DSR 6
J5-13 7
Port 2
R3 4.7K
+12V
SLICER
RS 232 Ports
Page 4 of 4

Connectors and Jumpers

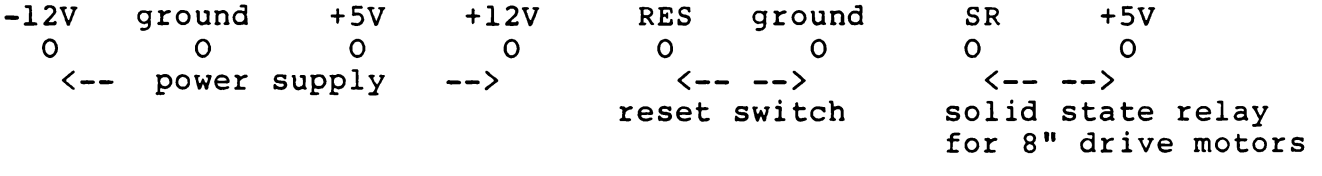
Connectors J1 thru J7 are intended for the use with standard ribbon cables. The cables can be connected straight thru to the ir destination. For example it is possible to make a connecting cable for the RS232 ports by crimping a cable with a 26 pin connector on the SLICER side and a 25 pin DB25 connector on the peripheral side after removing the 26th conductor on this side.

However, many printers use nonstandard pins for handshaking and in this case it will be necessary to rearrange the interconnections.

I/O definitions

bidirectional	
input	input to SLICER
output	output from SLICER

J8, Power Connector



J1, SASI port

J1

ground	1	!	2	/data0	bidirectional
ground	3	!	4	/data1	bidirectional
ground	5	!	6	/data2	bidirectional
ground	7	!	8	/data3	bidirectional
ground	9	!	10	/data4	bidirectional
ground	11	!	12	/data5	bidirectional
ground	13	!	14	/data6	bidirectional
ground	15	!	16	/data7	bidirectional
ground	17	!	18		
ground	19	!	20		
ground	21	!	22		
ground	23	!	24		
ground	25	!	26		
ground	27	!	28		
ground	29	!	30		
ground	31	!	32		
ground	33	!	34		
ground	35	!	36	/busy	input
ground	37	!	38	/acknowledge	output
ground	39	!	40	/reset	output
ground	41	!	42	/message	input
ground	43	!	44	/select	output
ground	45	!	46	data-/control	input
ground	47	!	48	/request	input
ground	49	!	50	output-/input	input

The 3 outputs can only drive 8 mA. The terminator network on the controller has to be disconnected for these lines.

J2, J3, Expansion Interface

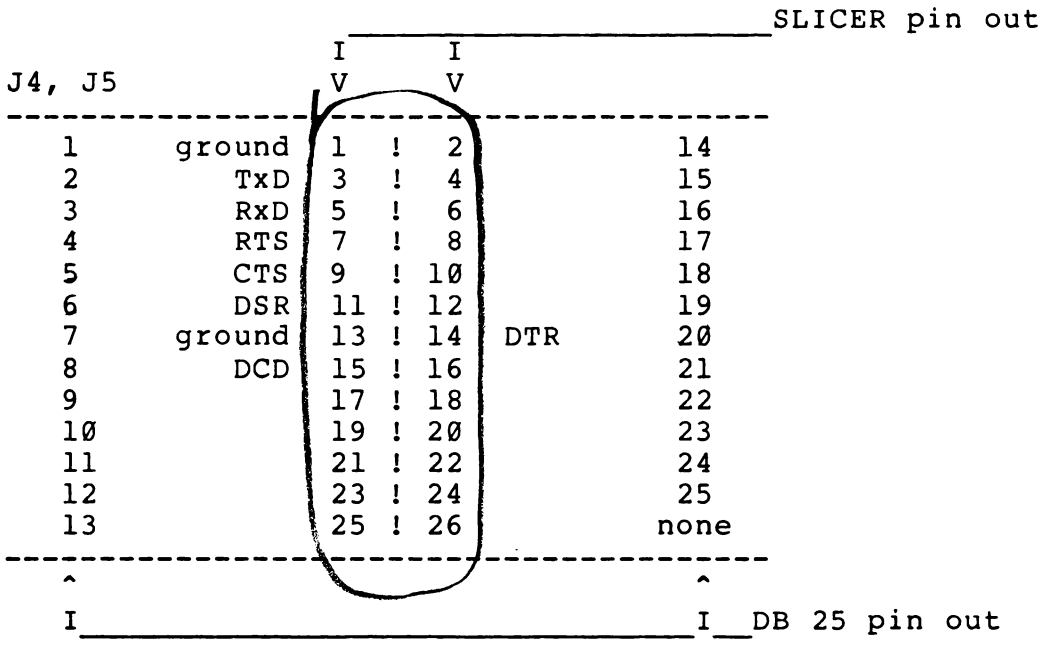
J2

input	expansion in	1	!	2	+5V
Bidirectional	data 15	3	!	4	+5V
bidirectional	data 14	5	!	6	+5V
bidirectional	data 13	7	!	8	+5V
bidirectional	data 12	9	!	10	+5V
bidirectional	data 11	10	!	12	+5V
bidirectional	data 10	13	!	14	+5V
bidirectional	data 9	15	!	16	+5V
bidirectional	data 8	17	!	18	+5V
bidirectional	data 7	19	!	20	+5V
bidirectional	data 6	21	!	22	ground
bidirectional	data 5	23	!	24	ground
bidirectional	data 4	25	!	26	ground
bidirectional	data 3	27	!	28	ground
bidirectional	data 2	29	!	30	ground
bidirectional	data 1	31	!	32	ground
bidirectional	data 0	33	!	34	ground
no connection		35	!	36	ground
output	/buffer enable	37	!	38	ground
input	input enabled	39	!	40	ground

J3

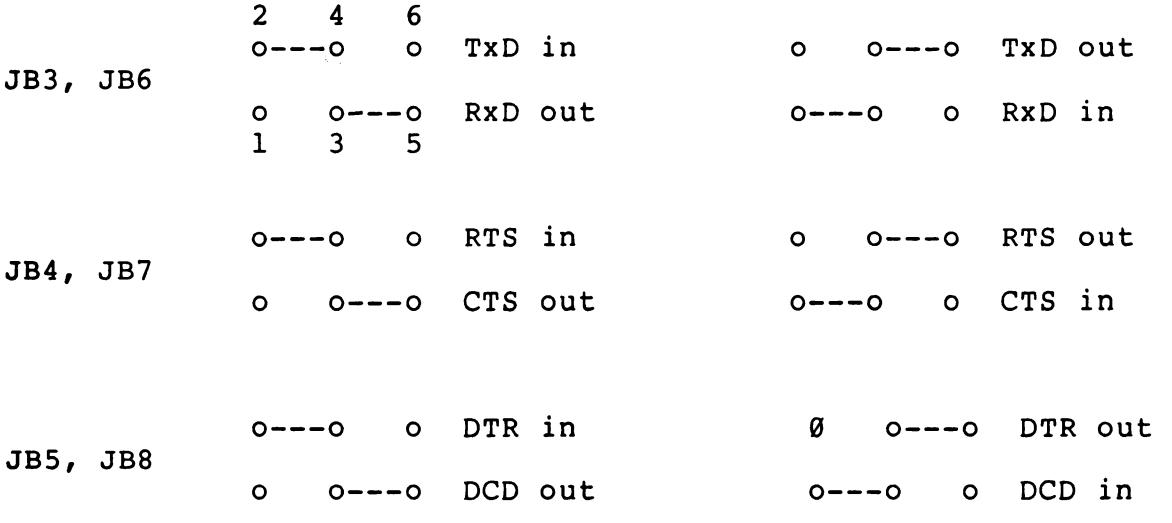
output	A0	1	!	2	A1	output
output	A2	3	!	4	A3	output
output	A4	5	!	6	A5	output
output	A6	7	!	8	A7	output
output	A8	9	!	10	A9	output
output	A10	11	!	12	A11	output
output	A12	13	!	14	A13	output
output	A14	15	!	16	A15	output
output	A16	17	!	18	A17	output
output	A18	19	!	20	A19	output
output	DT-/R	21	!	22	/BHE	output
input	DRQ	23	!	24	SASI-DRQ	output <-- jumper
output	/WR	25	!	26	/RD	output
output	/S0	27	!	28	ALE	output
output	/S2	29	!	30	/S1	output
input	/WAIT	31	!	32	/LCS	output
output	Timer1 Out	33	!	34	Timer2 Out	output
input	Timer0 In	35	!	36	/DEN	output
output	/PCS5	37	!	38	/PCS6	output
output	/MCS0	39	!	40	/PCS4	output
input	INT3	41	!	42	/MCS1	output
output	16 MHz clock	43	!	44	INT2	input
output	3.6864MHz clock	45	!	46	ground	
output	CPU clock	47	!	48	ground	
output	/Reset	49	!	50	ground	

J4, J5, Serial Ports



Port 1: J4, JB3, JB4, JB5
Port 2: J5, JB6, JB7, JB8

Serial Port Jumpers



J6, 5.25" drive, J7, 8" Drive

J6

ground	1	!	2		
ground	3	!	4		
ground	5	!	6	~~/motor on~~	output /select 3
ground	7	!	8	/index	input
ground	9	!	10	/select 0	output
ground	11	!	12	/select 1	output
ground	13	!	14	/select 2	output
ground	15	!	16	~~/select 3~~	output /motor on
ground	17	!	18	/step in	output
ground	19	!	20	/step	output
ground	21	!	22	/wr data	output
ground	23	!	24	/write	output
ground	25	!	26	/track 0	input
ground	27	!	28	/write prt	input
ground	29	!	30	/rd data	input
ground	31	!	32	/side sel 0	output
ground	33	!	34		

J7

ground	1	!	2	/low current	output
ground	3	!	4	/motor on	output
ground	5	!	6	/low current	output
ground	7	!	8	/index	input
ground	9	!	10	/side 2	input
ground	11	!	12		
ground	13	!	14	/side sel 0	output
ground	15	!	16		
ground	17	!	18	/head load	output
ground	19	!	20	/step	output
ground	21	!	22	/wr data	output
ground	23	!	24		
ground	25	!	26	/select 0	output
ground	27	!	8	/select 1	output
ground	29	!	30	/select 2	output
ground	31	!	32	/select 3	output
ground	33	!	34	/step in	output
ground	35	!	36	/step	output
ground	37	!	38	/wr data	output
ground	39	!	40	/write gate	output
ground	41	!	42	/track 0	input
ground	43	!	44	/write prt	input
ground	45	!	46	/rd data	input
ground	47	!	48		
ground	49	!	50		

Sources and Part Numbers for Ribbon Connectors

Connectors for ribbon cables are made by many manufactures. The parts numbers given here are for TB Ansley as sold by Hallmark, a nation wide distributor for electronics parts, and for Digi-Key, a distributor who sells in particular to the hobbyist. Their brand is Robinson Nugent.

Here is the address of Digi-key:

High Way 32 South
P.O. box 677
Thief River Falls, MN 56701

Tel. 1-800 346 5144
1-218 681 6674

Female connectors to the SLICER:

Type	Pos	Ansley	Digi-Key	Used for:
female	26	609-2600M	R304-ND	SLICER, serial ports
	34	609-3400M	R305-ND	SLICER, 5.25 floppy drive
	40	609-4000M	R306-ND	SLICER, expansion interface
	50	609-5000M	R307-ND	SLICER, SASI port
				expansion interface
				8" floppy drive
DB25				
male	25	609-25P	R602-ND	serial terminal
female	25	609-25S	R702-ND	serial terminal
edge	34	609-3415	R502-ND	5.25" floppy drive
edge	50	609-5015	R504-ND	8" floppy drive

Cross reference of SLICER parts
****

Unit: Identifies a functional unit on an IC with multiple units by an output pin #.
On: Identifies page and section of scematic for a particular unit.

	Type	Description	Unit	Used for	Function	On
U1	74LS240	octal inverting buffer		SASI port	status buffer	p3/4B
U2	74LS04	hex inverter	2	SASI port	data out enable	p3/3,4B
			8	SASI port	data write strobe	p3/3C
			10	SASI port	DMA request	p3/5B
			12	expansion interface	data buffer direction	p1/2F,G
			6,4	not used		
U3	74LS32	quad or	3	SASI port	data write strobe	p3/3C
			6	SASI port	data/status read strobe	p3/4C
			8	SASI port	acknowledge strobe /ACK	p3/4C
			11	expansion interface	data buffer direction	p1/2G
U4	i8283	octal inverting latch		SASI port	data out buffer	p3/3B
U5	74LS240	octal inverting buffer		SASI port	data in buffer	p3/1,2B
U6	2732A	32 k EPROM		monitor ROM	high/odd byte	p1/1,2,3F
U7	2732A	32 k EPROM		monitor ROM	low/even byte	p1/1,2,3E
U8	74LS245	octal bidirectional buffer		expansion interface	data buffer, high byte	p1/2,3G
U9	74LS245	octal bidirectional buffer		expansion interface	data buffer, low byte	p1/4G
U10	74LS373	octal latch		address bus	address latch	p1/4C
U11	74LS373	octal latch		address bus	address latch	p1/2,3C
U12	74LS373	octal latch		address bus	address latch	p1/1,2C
U13	74LS04	hex inverter	2,4	expansion interface	ALE buffer	p1/5F
			6,8	expansion interface	/RD buffer	p1/5F
			10,12	expansion interface	/WR buffer	p1/3A
U14	74HC00	CMOS hex inverter	2	floppy disk interface	clock buffer/oscilator	p1/1B
			4	serial interface	oscilator	p4/4D
			6		CPU clock	p1/1A
			8		CPU clock buffer	p1/1A
			10	serial interface	clock buffer	p4/4D
			12	expansion interface	reset buffer /RES	p2/3B
U15	74LS04	hex inverter	2	board reset	reset inverter	p2/3B
			4	serial interface	interrupt inverter	p4/3D
			6	not used		
			8	not used		
			10,12	expansion interface	systems clock buffer	p1/2A

	Type	Description	Unit	Used for	Function	On
U16	80186	CPU				p1/1..5B
U17				missing		
U18	SC2681	serial controller		serial interface		p4/1..4E
U19	1489	RS232 receiver		serial interface	inputs for port 1	p4/1,2F
U20	1488	RS232 transmitter		serial interface	outputs for port 1	p4/1,2F
U21	1489	RS232 receiver		serial interface	inputs for port 2	p4/1,2F
U22	1488	RS232 transmitter		serial interface	outputs for port 2	p4/1,2F
U23	74LS245	octal bidirectional buffer		DRAM interface	data buffer	p2/5E
U24	74LS245	octal bidirectional buffer		DRAM interface	data buffer	p2/3,4E
U25	TMS4500	DRAM controller		DRAM interface	address multiplex, timing	P2/1,2,3D
U26	74LS32	quad or	3	DRAM interface	/CAS driver,low byte	p2/1E
			6	DRAM interface	/CAS driver, high byte	p2/1E
			8	not used		
			11	DRAM interface	write line driver	p2/4D
U27	74LS74	dual d type flip flop	9	DRAM interface	wait request timing	p1/2A
			6	not used		
U28	74LS32	quad or	3	floppy disk interface	auxiliary status read	p3/2D
			6	floppy/SASI interface	write strobe for U29	p3/2D
			8	not used		
			11	DRAM interface	data buffer enable	p2/4D
U29	74LS259	octal addressable latch		floppy/SASI interface	drive select etc.	p3/1C,D
U30	FDC9229B	floppy interface circuit		floppy disk interface	data seperator and more	p3/4,5D
U31	7406	hex OC inverter/buffer	2	floppy disk interface	drive motor control	p3/1F,G
			4	floppy disk interface	drive 3 select	p3/2F,G
			6	floppy disk interface	drive 0 select	p3/1F,G
			8	floppy disk interface	drive 1 select	p3/1F,G
			10	floppy disk interface	drive 2 select	p3/1F,G
			12	floppy disk interface	force ready for mini drive	p3/5F,G
U32	FDC1797	floppy disk controller		floppy disk interface	data formating and more	p3/2..4E
U33	74S240	octal inverting buffer		floppy disk interface	output drivers	p3/2,3F,G
			9	floppy disk interface	ready input	p3/3F,G
U34	74LS244	octal buffer	3	floppy disk interface	data in buffer	p3/4F,G
			5,7,9	floppy disk interface	control inputs	p3/3F,G
			12	floppy disk interface	direct index input	p3/2E
			14	floppy disk interface	direct side information	p3/2D
U40..U55	4164	150ns DRAM		RAM, lower bank	bits 0..15	p2/2F
U60..U75	4164	150ns DRAM		RAM, upper bank	bits 0..15	p2/2G

	Type	Description	Unit	Used for	Function	On
J1		50 pin connector		SASI interface	all odd pins are grounded	p3/1..4A
J2		40 pin connector	even pins 2..20		+5V power	
			even pins 22..40		ground	
			1,39	expansion interface	bus control	p1/2G
			all other pins		data bus	p1/4,4H
J3		50 pin connector		expansion interface	address, /BHE	p1/1,2,3H
			21		DT/R	p1/2G
			26,28		/RDB,ALEB	p1/5F,G
			45		serial clock	p4/5D
			49		/RES	p2/3A,B
			24		SASI DMA request	p1/3A and p3/5B
			all other pins		control, select etc	p1/1..5A
J4		26 pin connector		serial port 1		p4/1,2G
J5		26 pin connector		serial port 2		p4/3,4G
J6		34 pin connector		for 5.25" floppy drive		p3/1..4G
J7		50 pin connector		for 8" floppy drive		p3/1..4H
J8		8 pin power connector		power, reset in		p2/1,2A
JB1		3 pin jumper		ROM size selction	4->2732, 16->27128	p1/1D
JB2		6 pin jumper		3 sense inputs		p4/3,4D
JB3		6 pin jumper		serial port 1	tx data <-> rx data	p4/1F
JB4		6 pin jumper		serial port 1	RTS <-> CTS	p4/1,2F
JB5		6 pin jumper		serial port 1	DTR <-> DCD	p4/2F
JB6		6 pin jumper		serial port 2	tx data <-> rx data	p4/3F
JB7		6 pin jumper		serial port 2	RTS <-> CTS	p4/3,4F
JB8		6 pin jumper		serial port 2	DTR <-> DCD	p4/4F
SW1		16 position switch		floppy disk interface	write precomp selection	p3/5D
X1	Xtal	16 MHz standard		CPU clock		p1/1A
X2	Xtal	3.6864 MHz		serial clock		p4/4D
X3	Xtal	16 MHz (optional)		floppy disk interface	used if X1 not 16 MHz	p1/1B
D1	1N914	diode		CPU clock oscillator	pulse squaring	p1/1A
D2	1N914	diode		reset logic	discharges C52	p2/2B
R2	4.7k	resistor		serial port 1	pull up for DSR	p4/2G
R3	4.7k	resistor		serial port 2	pull up for DSR	p4/4G
R4	10k	resistor		floppy disk interface	pull up for interrupt	p1/3B
R5	10k	resistor		floppy disk interface	pull up for DMA request	p1/3B
R6	33 Ohm	resistor		DRAM interface	driver for low /CAS	p1/1E,F
R7	33 Ohm	resistor		DRAM interface	driver for high /CAS	p1/1E,F
R8	10k	resistor		DRAM controller	reset logic	p2/3E
R9	10k	resistor		CPU	pull up for reset line	p2/2B

	Type	Description	Unit	Used for	Function	On
RP1	9 * 1k	resistor pack, pin 1 common	2,3,4	SASI port	pull ups for status inputs	p3/4B
			7,9	SASI port	pull ups for status inputs	p3/4B
			10	expansion interface	pull up for /EXP IN (J3-3)	p1/2G
			5,6,8	not used		
RP2	9 * 1k	resistor pack, pin 1 common	2-9	SASI port	pull ups for data lines	p3/1,2B
			10	expansion interface	pull up for /WAIT (J2-31)	p1/2A
RP3	3 * 47k	resistor pack	1-2	floppy clock		p1/1B
			3-4	serial clock		p4/4D
			5-6	CPU clock		p1/1A
RP4	5 * 10k	resistor pack, pin 1 common	2,3,4	sense inputs	pull up resistors	p4/3D
			5	not used		
			6	serial controller	pull up for interrupt	p4/3D
C1..C4	0.1 uF	ceramic film capacitor		5V bypass		
C6	0.1 uF	ceramic film capacitor		5V bypass		
C11	0.1 uF	ceramic film capacitor		5V bypass		
C14,C15	0.1 uF	ceramic film capacitor		5V bypass		
C24	0.1 uF	ceramic film capacitor		5V bypass		
C28	0.1 uF	ceramic film capacitor		5V bypass		
C31..C50	0.1 uF	ceramic film capacitor		5V bypass		
C54,C55	0.1 uF	ceramic film capacitor		5V bypass		
C5	10 uF	tantalum capacitor		5V bypass		
C30	10 uF	tantalum capacitor		5V bypass		
C51	10 uF	tantalum capacitor		5V bypass		
C52	10 uF	tantalum capacitor		CPU	reset	p2/2B
C53	10 uF	tantalum capacitor		DRAM controller	reset	p2/3E
C7,C8	100 pF	ceramic film capacitor		CPU clock		p1/1A
C9	220 pF	ceramic film capacitor		serial port clock		p4/4D
C10	100 pF	ceramic film capacitor		serial port clock		p4/4D
C12	100 pF	ceramic film capacitor		floppy disk clock	optional	p1/1A
C13	100 pF	ceramic film capacitor		floppy disk clock	optional	p1/1B
C16..C23	220 pF	ceramic film capacitor		serial interface	RFI/noise suppression	p4/1..4F
C25..C27	220 pF	ceramic film capacitor		serial interface	RFI/noise suppression	p4/1..4F
C29	220 pF	ceramic film capacitor		serial interface	RFI/noise suppression	p4/1..4F
	1000pF	ceramic film capacitor		SASI port	deglitch for /ACK	p3/4C

this capacitor is mounted on the solder side between pins 7 and 8 of U3

The SLICER

Check Out and Trouble Shooting

The low parts count makes the SLICER fairly easy to check. The procedures described in this part of our documentation will help you to get your SLICER working and to maintain it. An oscilloscope with a bandwidth of at least 25 MHz is a desirable tool, however much can be done with a logic probe.

All position references like up, down etc assume that the power connector J8 is in the lower right corner of the board.

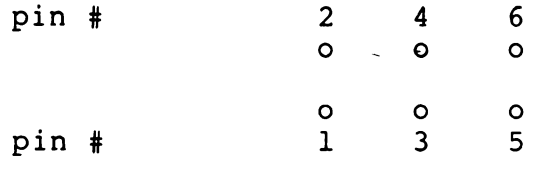
References to ICs look like this: U18/31. The number following the / is the pin number.

References to schematics look like this: p4/1F. The term 1F is the referenced coordinate on page 4.

Bringing Up the System

The SLICER waits after power up for a carriage return from its console port (port #1). It uses this to determine the baud rate of the terminal. A sign on message will be send once the baud rate has been found. At this time no RAM is reqired by the system.

No sign on message can be caused by an incorrectly connected terminal or by a problem with the SLICER. First let us check out the terminal and let us define for this purpose the pin numbers of the terminal jumpers JB3..JB8 (p4/1..4F):



Pins 1 and 2 are connected to a RS232 input to the SLICER, pins 5 and 6 are connected to a RS232 output from the SLICER and pins 3 and 4 are connected to the terminal.

RS232 signal levels are up to +/- 12 volt. Keep this in mind when you use a logic probe.

The nature of a RS232 line can be determined easily: A line at a level of < -4 volt or > +4 volt is an output. If the level is within +/- 2 volt it is an input. The RS232 standard requires that lines can be shorted to ground or each other without damage.

Both ports are programmed to support hand shaking via the RTS and CTS lines. These lines are handled by JB4 (JB7 for port 2). A port will not transmit if /CTS (U18/7 or U18/4) is high and it will put /RTS (U18/29 or U18/12) high if its input FIFO is full.

/DTR (U18/28 or U18/13) is programmed to be permamently low and /CD (U18/36 or U18/2) is ignored.

Terminal Checks: No Signon Message

1. Does the terminal transmit data?
   Press a key on the terminal repeatedly and check JB3 pins 3 and 4 for data. Jumper the pin with the data to an input, eg 2-4 or 1-3. Jumper the other pin to an output, eg 3-5 or 4-6.
   Trace received data: U19/4 (+/-12v) (inverter) U19/6..U18/31.

2. Terminal does not transmit data.
   See if the hand shake outputs of the SLICER are high: pins 1,2 on JB4 and JB5. If not, trace JB4/1..U20/3 (inverter) U20/2..U18/29 and JB5,1..U20/8 (inverter) U20/9,10..U18/28. A high level on U18 pin 28 or 29 points to a systems fault.
   Test hand shake signals of the terminal on JB4/3,4 and JB5/3,4. Determine which line is input, which is output. Install the proper jumpers and make sure again that the hand shake inputs to the terminal are high. Then see if the hand shake outputs from the terminal are high too. A low here points to a problem with the terminal. The terminal should generate data if the hand shake lines are all high.

3. Terminal transmits but no signon message.
   Check U18/30 for data after reset and carriage return. No data indicates a system problem. Trace data (repeating reset and CR) via U20/4,5 (inverter) U20/6..JB3/1,2..jumper to terminal.
   A short burst of data after a reset and CR indicates that the SLICER did recognize a CR and is sending the sign on message.
   A continuous stream indicates that the SLICER saw something but did not recognize a CR. It is now transmitting continuously what it thinks it saw at a rate of 1200 baud. Check the baud rate of your terminal in this case, it must be one of the following: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400.

Basic Systems Check

1. CPU clock (p1/1A)
   U14/8 16 MHz/62.5 nsec, duty cycle 50%.
   If no signal or a bad signal check polarity of diode D1 and the values of C7, C8 (must be 100pF).

2. CPU clock output (p1/2B)
   U15/13 8 MHz/125 nsec, duty cycle 50%.

3. Serial clock (p4/4D)
   U18/33 3.6864 MHz/271 nsec, duty cycle aprox. 40 low 60 high.
   Check value of C9 (220pf) and C10 (100pf).

4. ROM select line /UCS (p1/3E..F)
   U7/20 or U6/20. Note that the pin numbering applies to a 28 pin socket. This line should show low pulses of 450 nsec with irregular intervals of 60 nsec and more. A short burst of pulses after reset could be caused by swapped ROMs.

5. Address latch strobe ALE (p1/1,2,3C)
   U1Ø,11,12/11 A short high pulse of 125 nsec and an irregular period of 5ØØ nsec and more.

6. Address and data bus check
   Apply a low on J3/31 (/WAIT), then reset the SLICER.
   The 8Ø186 is now halted while reading the first instruction at *FFFF*Øh. It should read (high byte first) AØBAh.
   Check if address and data are correct. The address appears on J3/1 (AØ)...J3/2Ø (A19) (see p1/1,2H). The data appears on J23/3 (DB15)..J2/33(DBØ) (odd numbered pins only, see p1/3,4H) when J2/37 (p1/3G) is jumpered to ground (on J2/38).

7. Single step circuit.
   The single stepper shown below allows you to examine more than only the first 2 bytes after reset. We actually used this circuit together with an address and data display to get the SLICER going and still use it for debugging new SLICERs.
   I don't think it is worthwile to build the display part only one SLICER.

8. Addresses and data after reset.
   Address and data are shown with the most significant nibble first. The program begins with changing the address range of the UCS line to cover the full possible ROM. This part of the program is unlikely to change in the future.

address	data
FFFF0	A0BA
FFFF2	B8FF
FFFF4	F838
FFFF6	EAEF
FFFF8	0000
FFFFA	F800
FFFFC	CFC3

   The UCS line has been reprogrammed and the program now jumps to the begin of the ROM. The program from here on is more likely to change in future revisions of the monitor.

F8000	F8B8
F8002	BA3F
F8004	FFA2
F8006	B8EF
F8008	41F9
F800A	A6BA
F800C	EFFF
F800E	3BB8

9. Serial interface (page 4)
   Check chip select line U18/35 (/SIOCE) (p4/3E). A low pulse of 800 nsec duration and a 4 microsec period after reset.

RAM Check Out

The SLICER monitor has a special section for the checking and debugging of RAM. Its great strength is that it needs no RAM to run. It is invoked automatically at power up and reset if the system does not find at least 128 kBytes of RAM or it can be accessed by the MT command from the monitor. The memory test signs on by displaying its menu.

The basic memory test (the T command) is quite good: it does not use a short repetitive pattern, instead it fills a 65 kByte block of RAM with a repeated image of the monitor code. It then proceeds to read this data and compare it to the code in ROM until it is stopped by a <CR>.The next test will now invert the data from ROM. Thus all bits can be checked as Ø or 1. It is
also possible to suspend the test by pressing ctrl S. The test will resume when a key is pressed. This feature allows to test the refresh function.

The location of the memory block to be tested can be selected by entering a segment address (the S command). The default segment on entering the memory test is Ø.

The error display is in the form

RAM address RAM data ROM data .

The address is relative to the segment address, the data shows words with the most significant nibble first. The error display can be halted by hitting any key, it will resume if a key is hit again.

The assembly instructions show how to use the memory test for the installing of the RAM. Two more functions help in the analysis of more persistent problems: the read (R) command and the write (W) command. Both functions will continuously access memory and they allow signal tracing with an oscilloscope. The address can be selected with the S and the A command, the effective address is then 16*S+A. The data to be written can be entered by the D command. The R function first writes the data once before going into its read cycle.

If an even address is entered a full word is accessed, an odd address will access two bytes in sequence.

The signals associated with the DRAM will vary in length. This is due to refresh cycles coinciding with the access cycle. The times given below are basic times without refresh.

The following signals are of interest:

1. RAM select /LCS (U25/4).
   A low pulse of 450 nsec and a period of 5 usec.
   This signal should be used as a trigger for viewing the other signals.

2. /RAS (row address strobe)
   /RAS0 (U25/6) for an effective address < 20000h
   /RAS1 (U25/7) for an effective address beween 20000h and 3FFFFh
   Appears 75 nsec after /LCS.

3. MA0..MA7 (multiplexed address)
   On U25/13,14,19,21,24,27,32,33.
   Set effective address to AB54 for change on all lines.
   Row address appeares 50 nsec after /LCS,
   Column address appears 200 nsec after /LCS.

4. /CAS (column address strobe) (U25/10)
   Appears 225 nsec after /LCS
   Trace via U26/2 or gate U26/3..R6 (33 Ohm) to pin 15 of the even/low byte RAMs,
   and via U26/4 or gate U26/6..R7 (33 Ohm) to pin 15 of the odd/high byte RAMs.

5. RAM bus enable (U23/19 and U24/19)
   Goes low 75 nsec after /LCS.
   Trace back to U28/11 or gate U28/12 (/DEN).

6. Bus direction DT//R (U23/1 and U24/1)
   Low for read, high for write.

7. /RWAIT (RAM wait, U25/2)
   Indicates that a refresh cycle is in progress during a memory access. Use as trigger for oscilloscope.
   Goes low for 1 to 4 clock periods (125..500 nsec).
   Trace to U27/12 (p1/2A,B), synchronized by rising edge of CPU clock at U27/9..U16/55 (ARDY).

OVERVIEW

The monitor consists of four parts:

1) Power up and Reset
2) Memory Test
3) IO
4) Debugger

Part 1, Power Up and Reset, initializes memory, IO devices, and interrupts. The Memory Test helps find problems in the RAM. The IO part allows programmers to access the SLICER's hardware. It also contains the interrupt service routines, including a type ahead buffer, printer spooling, and clock. The Debugger has routines for both software and hardware debugging.

The source code for the monitor is provided on disk. It can be assembled with Digital Research's ASM-86 Vers. 1.1. (We have been told that version 1.Ø will not work.) The command line

```
A>ASM86 SLICER
```

will assemble the monitor. The monitor source is contained in several files. At this time, (monitor vers. I.2) the additional files are named CODES.A86, DEFIN.A86, RESET.A86, MEMTEST.A86, IO.A86, MON.A86.

POWER UP AND RESET

DEBUGGING TIPS

Since most SLICERs are sold as kits, the monitor was designed to help bring up untested boards.

RESET

On reset the 80186 puts UMCS at FFFF0 and begins to execute code at FFFF0. So reset should cause UMCS to go low. If the CPU can read the EPROMs, (and the EPROMs are inserted correctly,) there should be a lot of activity on UMCS. Otherwise UMCS will be mostly high.

CONSOLE

After some initialization the CPU will go into a loop waiting for input from the serial port. This can be recognized by pulses on PCS1, while TXDA (on the DUART) is steady high. The CPU assumes the input is a CR and tries to determine the console baud rate. If after three tries the baud rate cannot be determined, the CPU goes into an echo loop. Baud rate is set at 1200 and characters are continuously transmitted. You should see activity on TXDA. The CPU is also looking for input, the input becomes the character being transmitted.

If the baud rate is determined a sign on message is sent. You should see a short burst of activity on TXDA. Assuming the baud rate is correct, your terminal should display 'the SLICER'.

RAM

Next the program checks how much RAM is installed. If it finds less than 128K (which is impossible with working RAM) it goes to the memory test. The memory test program does not need RAM to run.

If working RAM is found then control passes to the monitor's debugging program. At this point your have the CPU, EPROMS, console, and RAM working. You can get to the memory test from the debugger by typing 'MT'.

The trouble shooting documentation describes the memory test in more detail.

Most of the remainder of this document is for programmers. For now, skip ahead to the section titled 'DEBUGGER'.

INITIALIZATION

The following tables describe how the SLICER is initialized.

SEGMENTS

CS	F800
DS	F800
ES	F800
SS	F800

Addresses in these descriptions are given in SEGMENT:OFFSET format, most have SEGMENT value F800. Some examples;

NAME	ADDRESS	ABSOLUTE
ROM START	F800:0	F8000
RESET START	F800:7FF0	FFFF0
INTERRUPT VECTORS	F800:8000	00000
MONITOR RAM	F800:8400	00400

Chip select lines are programmed as follows: (Power up and reset)

NAME	ABSOLUTE ADDRESS	USE RDY	WAITS
UMCS	F8000-FFFFF	TRUE	0
LMCS	00000-3FFFF	TRUE	0
MCS0	40000-4FFFF	TRUE	1
MCS1	50000-5FFFF	TRUE	1
MCS2	60000-6FFFF	TRUE	1
MCS3	70000-7FFFF	TRUE	1
PCS0	0- 7F	TRUE	3
PCS1	80- FF	TRUE	3
PCS2	100- 17F	TRUE	3
PCS3	180- 1FF	TRUE	3
PCS4	200- 27F	TRUE	3
PCS5	280- 2FF	TRUE	3
PCS6	300- 37F	TRUE	3

The PCSs are IO mapped. Internal peripherals are IO mapped at FF00-FFFF.

SERIAL PORTS. (Power up and reset)

Both transmit and receive are enabled on both serial ports. All interrupts are masked off except for port A receive. Bit outputs are set as:

OP0	/RTSA	LOW
OP1	/RTSB	LOW
OP2	/DTRA	LOW
OP3	/DTRB	LOW
OP4	NC	HIGH
OP5	NC	HIGH
OP6	NC	HIGH
OP7	(floppy) MOTOR	HIGH

SERIAL PORTS (Power up only)

Both ports are set for 8 bits per character, with no parity, two stops, and RTS/CTS handshaking. Port B is set for 9600 baud. The port A baud rate is acquired by software. Baud rates of 38400, 19200, 9600, 4800, 2400, 1200, 600, and 300 are recognized by the monitor.

INTERRUPT VECTORS (Power up only)

All interrupt vectors not otherwise initialized are set to a bomb routine. The bomb routine saves the CPU registers then enters the debugging monitor. An undefined opcode is the most probable cause of an unexpected interrupt.

TYPE	VECTOR NAME	DESCRIPTION
62	VRESET	RESET VECTOR. Can be made to point to a user's reset handling routine. Initialized to the debugging monitor.
63		Not used as a vector, used to check the validity of the reset vector.

(The following are initialize on both power up and reset)

12	VINTØ	DUART INTERRUPT. Used for console input buffering.
61	VLST	OTHER DUART INTERRUPTS. Indicates printer ready for buffered output.
60	VSPOOL	GENERATED WHEN SPOOLING IS FINISHED. Should be set to a routine which frees up the print buffers. Initialized to do nothing.
19	VTMR2	TIMER #2. Used for software clock, and floppy drive shut off.

Along with these vectors, any peripherals and memory locations needed by the interrupts are initialized.

MEMORY INITIALIZATION (Power up only)

ADDRESS	NAME	VALUE	DESCRIPTION
	DISK PARAMETERS		
F800:8483	DSKSEG	0080	MEMORY SEGMENT FOR DISK TRANSFER
F800:8485	DSKMEM	0000	MEMORY OFFSET FOR DISK TRANSFERS
F800:84F7	DSKNUM	00	DISK NUMBER
F800:84F8	DSKTRACK	0000	TRACK
F800:84FA	DSKSECTOR	01	SECTOR
F800:84FC	TRKSIZE	26	SECTOR/TRACK
F800:84A0	DSKTABLE	A TABLE CONTAINING DISK DRIVE CHARACTERISTICS. Initially shows only disk 0 connected, a single density 8 inch floppy.	
	ENTRY POINTS		
F800:8400		A TABLE OF MONITOR IO ENTRY POINTS.	

(Both power up and reset)

	DUART		
F800:84F1	DUIMR	02	IMAGE OF DUART INTERRUPT MASK REGISTERS.
F800:84F0	DUOUT	F0	IMAGE OF DUART OUTPUTS.
	TIMER		
F800:84E8	COUNT	0000	CURRENT TIMER COUNT.
F800:84EA	SECONDS	0000	SECONDS SINCE RESET
F800:84EC	TCONVRT	0032	TIMER COUNTS PER SECOND
F800:84FD	FDSTOP	0078	TIME TO SHUT OFF FLOPPY

RESET

F800:80F8	VRESET
F800:80FE	CHECKSUM
F800:84EE	MEMSIZE

On reset the monitor checks if RAM has been initialized. Memory is considered valid if F800:80FE contains the sum of F800:80F8, 80FA, and 80FC XORed with the constant 0D0BEH, and F800:84EE is the same as F800:80FC. If memory is valid then F800:80F8 contains the reset vector, and F800:80FC has the memory size (in Ks). On power up the reset vector is set to the debugger. On reset, after the above initialization, control goes to the reset vector. Interrupts are not yet enabled, the reset routine must enable interrupts.

MONITOR INTERFACE

The IO section of the monitor provides access to the SLICER's hardware. At F800:8400 there is a table of entry points to the monitor. Use a far call with these entry points. The address must be specified exactly as shown, both segment and offset. Different segment, offset pairs, even if they address the same absolute location, will not work.

F800:8404 CONST

Return console input status.

OUT: AL=0 If character not ready
AL=-1 If character ready
Zero flag set by the contents of AL

F800:8408 CONIN

If a character is ready, grab it and return, otherwise wait for keypress then return character.

OUT: AL=ASCII character

NOTE: Parity bit is set to zero.

F800:840C CONOUT

Output AL to console.

IN: AL=ASCII character

F800:8410 DUBIN

Get a character from port B of DUART.

OUT: AL=ASCII character

F800:8414 DUBOUT

Output AL to DUART port B. (Probably a printer.)

IN: AL=ASCII character

F800:8418 LSTSPOOL

Buffered printer output. Printer (assumed at DUART port B) is set up for background operation. Up to 64K of text can be buffered.

IN: AX=Text start, offset
BX=Segment
CX=Text end, offset

OUT: Carry set if printer is busy.

Interrupt #60 is generated when spooling is finished. Bit 4 of DUIMR is set while the printer is busy. The foreground program could monitor this bit.

F800:841C HOME

Home disk drive. (Actually a flag is set which causes the drive to recalibrate before the next read or write).

OUT: Carry set if unknown drive has been selected.

F800:8420 READ

Read a disk sector.

OUT: AL=Error code (0=no error)
Z set by the contents of AL.

See 'READ/WRITE SECTOR' below for details.

F800:8424 WRITE

Write disk sector.

OUT: AL=Error code (0=no error)
Z set by the contents of AL.

See 'READ/WRITE SECTOR' below for details.

F800:8428 DUON

Turn on DUART interrupts.

IN: AL=Bit mask.
OUT: AL=New interrupt mask.

F800:842C DUOFF

Turn off duart interrupts.

IN: AL=Bit mask.
OUT: AL=New interrupt mask.

F800:8430 DUHIGH

Set bit outputs high.

IN: AL=Bit mask
OUT: AL=New output states

F800:8434 DULOW

Set bit outputs low

IN: AL=Bit mask
OUT: AL=New output states

F800:8438 LSTOUT

Output AL to the printer. (Printer assumed at DUART port B)

IN: AL=ASCII code.

READ/WRITE DISK SECTOR

Before doing a read or write the following disk parameters must be set

F800:84F3	DSKMEM	2 BYTES
F800:84F5	DSKSEG	2 BYTES
F800:84F7	DSKNUM	1 BYTE
F800:84F8	DSKTRACK	2 BYTES
F800:84FA	DSKSECTOR	1 BYTE

DSKNUM is the logical disk number. A disk table gives the physical characteristics of each disk. Initially this table is set to show that only disk #Ø is connected. and that disk Ø is an eight inch, single density, single sided floppy, with no track/sector translation, mounted in drive #Ø. DSKMEM and DSKSEG hold the offset and segment for the disk memory area. DSKTRACK and DSKSECTOR hold the logical track and sector. Initially there is no translation, so this is the same as the physical track and sector.

READ and WRITE return an error code in AL.

Ø indicates no error. Z is set by AL, so NZ indicates a error.

General error codes, (all types of disks).

FF	Drive does not exist.
FD	Bad DMA count.

Error codes for floppies.

8Ø	Not ready.
4Ø	Write protected.
2Ø	Write fault.
1Ø	Seek error or record not found.
8	CRC error.
4	Lost data.
2	Data request. (Disk expects more.)

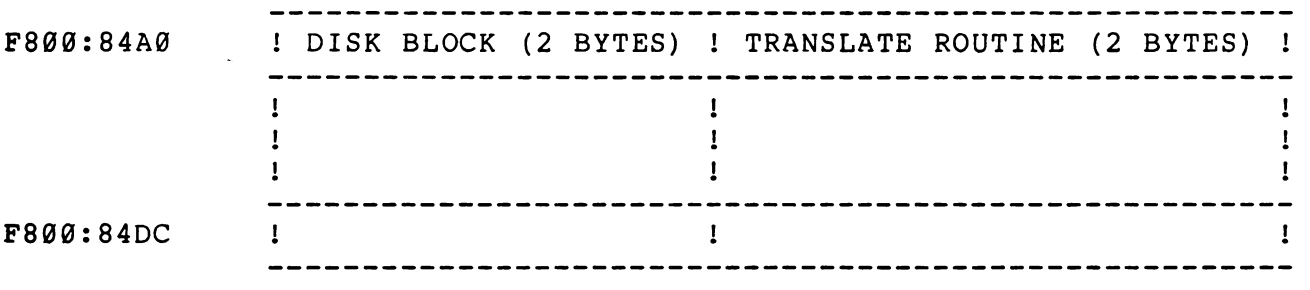
The above floppy error codes are ORed if more than one error condition is present.

DISK DEFINITIONS

Parameters for a disk operation are stored in the following variables.

F800:84F2	DSKCMD	1 BYTE
F800:84F3	DSKSEG	2 BYTES
F800:84F5	DSKMEM	2 BYTES
F800:84F7	DSKNUM	1 BYTE
F800:84F8	DSKTRACK	2 BYTES
F800:84FA	DSKSECTOR	1 BYTE

DSKNUM identifies a entry in the logical to physical disk translation table.



(16 entries, 4 bytes each.)

Each entry in the table consists of a pointer (in segment F800) to a block of physical parameters for that disk, and a logical to physical translation routine. The translation routine gets

```
IN:     AX=Track
        BX=Pointer to disk block
        CH=0
        CL=Sector
```

and should return

```
OUT:    AX=Physical track
        BX=Unchanged
        CH=Side
        CL=Physical sector
        DI=Don't care
```

and leave all other registers unchanged.

The disk block consists of the following:

DISK	1 BYTE
SIDE	1 BYTE
TRACK	2 BYTES
SECTOR	1 BYTE
LSTTRK	2 BYTES
TYPEPTR	2 BYTES

DISK gives the physical drive number, (Ø-3 for floppies, Ø-1 for Winchester). LSTTRK is used to determine whether a seek is needed. If LSTTRK=-1 then the drive is recalibrated. TYPEPTR points (in segment F8ØØ) to a disk type table which consists of:

DTYPE	1 BYTE
STEPRATE	1 BYTE
SECLEN	2 BYTES
DATAADR	4 BYTES
RDCMD	2 BYTES
WRTCMD	2 BYTES

The disk type table does not change, so it may be in EPROM, and several drives may use the same type table. DTYPE gives drive characteristics.

BITØ	MINI
BIT1	DOUBLE DENSITY
BIT3	FLOPPY
BIT4	HARD

SECLEN, DATAADR, RDCMD, WRTCMD are used to set the DMA. SECLEN is the number of bytes per sector. DATAADR is the data address for the disk. (segment:offset, disks are IO mapped, so set segment to Ø. Offset=6 for floppies and 184H for Winchester). RDCMD and WRTCMD are the DMA commands for read and write operations respectivly.

Routines for hard disk are not written yet. Exits to user routines are provided.

USRØ	F8ØØ:84EØ
USR1	F8ØØ:84E4

If bits 3 and 4 of DTYPE are both zero then USRØ and DMAØ are used. If both bits 3 and 4 are one then USR1 and DMA1 are used.

User disk routines get

```
IN:     BX=Pointer to disk block
        DX=DMA control port
        DI=Pointer to disk type table
        DSKCMD=1 for write, Ø for read
```

In addition the disk block has been set up, and the DMA is set (but not enabled). The user routine should do the read or write then return

```
OUT:    AL=Error code (Ø for no error)
        CX=Don't care
        DX=Don't care
        SI=Don't care
        All other registers=Unchanged.
```

Also Z should be set by the contents of AL. (Z set if no error.)

After return is made from the user disk routine, the DMA count is checked. If the user's disk does not use DMA, the DMA count register should be set to zero. The following code could be used:

```
SUB     DX,2    ;Set DX to DMA count register.
MOV     AX,Ø
OUT     DX,AX   ;Set count to Ø
```

On entry to the user routine, the DMA is set up but not enabled. The following code will enable the DMA:

```
IN      AX,DX   ;Get DMA command
OR      AL,6    ;Set start bits
OUT     DX,AX   ;Then start DMA.
```

F800:8030	VINT0
F800:80F4	VLST
F800:80F0	VSPOOL
F800:84F0	DUOUT
F800:84F1	DUIMR
F800:8428	DUON
F800:842C	DUOFF
F800:8430	DUHIGH
F800:8434	DULOW

Serial IO is done with the Signetics SC2681 DUART. The base address for the DUART is at 80H (in IO space). DUART registers are addressed as follows:

	READ	WRITE
DUART	MODE REG. A (MR1A,MR2A)	MODE REG. A (MR1A,MR2A)
DUART+2	STATUS REG. A (SRA)	CLOCK SELECT A (CSRA)
DUART+4		COMMAND REG. A (CRA)
DUART+6	RX HOLDING REG. A (RHRA)	TX HOLDING REG. A (THRA)
DUART+8	INPUT PORT CHG. REG. (IPCR)	AUX. CONTROL REG. (ACR)
DUART+10	INT. STATUS REG. (ISR)	INT. MASK REG. (IMR)
DUART+12	C/T UPPER (CTU)	C/T UPPER REG. (CTUR)
DUART+14	C/T LOWER (CTL)	C/T LOWER REG. (CTLR)
DUART+16	MODE REG. B (MR1B,MR2B)	MODE REG. B (MR1B,MR2B)
DUART+18	STATUS REG. B (SRB)	CLOCK SELECT B (CSRB)
DUART+20		COMMAND REG. B (CRB)
DUART+22	RX HOLDING REG. B (RHRB)	TX HOLDING REG. B (THRB)
DUART+24		
DUART+26	INPUT PORT	OUTPUT CONF. REG. (OPCR)
DUART+28	START COUNTER COMMAND	SET OUTPUT BITS
DUART+30	STOP COUNTER COMMAND	RESET OUTPUT BITS

Serial port A is the console port. Baud rate for the console is acquired by typing a CR on power up. Baud rates of 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 are recognized. Other baud rates are available on the DUART but not supported by the monitor.

The DUART can generate interrupts on eight different conditions. Memory location DUIMR (F800:84F1) keeps track of which interrupts are enabled. A bit set in DUIMR indicates that the interrupt is enabled.

```
BIT 0   TxRDYA
BIT 1   RxRDYA
BIT 2   CHG. BRK A
BIT 3   COUNTER RDY
BIT 4   TxRDY B
BIT 5   RxRDY B
BIT 6   CHG. BRK B
BIT 7   IN. PORT CHG.
```

Monitor routines DUON, DUOFF will enable and disable interrupts, update DUIMR. All DUART interrupts are type 12. This vector is set to the console input routine which generates interrupt #61 if no console character is ready. If another DUART interrupt is used its vector should be put at F800:84F4. (Buffered printer output is an example of such an interrupt.)

Console input has a 32 character type ahead buffer. Besides checking the input buffer, the console status routine, CONST, checks for a character at the console. If a character is present then a software interrupt is generated. This allows console input even while interrupts are disabled.

Routines have been provided for buffered output to port B. To use them put the buffer start location in AX, end in CX, and segment in BX then do a far call to LSTSPOOL at F800:8418. After the last byte has been sent, interrupt #60 is generated, and bit 4 of DUIMR is cleared. The foreground program could monitor this bit or use the interrupt to reclaim the buffer space. Serial port B is initialized to 9600 baud, no parity, 2 stops, CTS handshaking. You will probably have to reprogram it for your printer.

The DUART has two sets of baud rates. They are:

CODE	SET 0	SET 1
00	50	75
11	110	110
22	134	134
33	200	200
44	300	300
55	600	600
66	1200	1200
77	1050	2000
88	2400	2400
99	4800	4800
AA	7200	1800
BB	9600	9600
CC	38400	19200

One of these sets has been selected by the console baud rate program. If your console has a 38400 baud rate then set 0 has been selected. Otherwise set 1 has been selected. If your console baud rate is less than 19200 you can still use set 0. The following instructions select set zero.

```
MOV     AL,0
OUT     88H,AL          ;Select baud rate set 0.
```

Do not switch baud rate sets if your terminal is operating at 19200 or 38400 baud, this will change the baud rate for the terminal port. Port B baud rate is set by outputting the baud rate code to 92H. For example:

```
MOV     AL,0BBH         ;Set port B baud
OUT     92H,AL          ;to 9600.
```

The DUART has eight bit outputs. Memory location DUOUT (F800:84F0) holds the current state of the outputs. To set or reset these outputs put a bit mask in AL and call DUHIGH or DULOW. The bits set in AL will be set (or reset) on the outputs and set (or reset) in DUOUT.

DEBUGGER

The monitor prints a '+' prompt to indicate it is in debugging mode. The following dbugging commands are available:

Dml,t	DUMP HEX/ASCII
DQml,t	DUMP ASCII ONLY
Sml,bl,...,bn	SUBSTITUTE MEMORY
Pml,text	PUT TEXT INTO MEMORY
Fml,t,b	FILL MEMORY
Mml,t,m2	MOVE MEMORY
RMml,t,m2	MOVE MEMORY, REVERSE ORDER
Cml,t,m2	COMPARE MEMORY
Ip	INPUT
Op,w	OUTPUT
Hwl,w2	HEX MATH
X	EXAMINE REGISTERS
Xr	EXAMINE/MODIFY A REGISTER
Tn	TRACE
Un	UNTRACE
Gg,bl,b2	GOTO WITH BREAKS
MT	MEMORY TEST
DMm	SET DISK TRANSFER MEMORY AREA
DSb	SET DISK SECTOR
DNb	SET DISK NUMBER
DTw	SET DISK TRACK
DZ	ZERO DISK (RECALIBRATE)
DP	DISPLAY DISK PARAMETERS
DRbl,b2	DISK READ
DWbl,b2	DISK WRITE
SS	SHOW SEGMENTS
B	BOOT DOS
&	AGAIN
Z	GO TO SLEEP

Commands are given by typing the one or two letter command code, followed by up to three parameters separated by commas. All numbers are assumed to be in hexadecimal, When a number is expected for input you may type as many digits as you want, the computer ignores all but the last two (for bytes) or four (for words). If a parameter is missing a default value is used. For example, the dump command, 'Dml,t', has two parameters, ml for 'memory from' and t for 'up to'. The default for ml is 'where the last command left off' and default for t is 'enough to fill the screen' (ml+13F). Suppose you type

```
i) DØ,2Ø
ii) DØ          (or DØ,)
iii) D,1FF
iv) D           (or D, or D,,)
```

i) will dump memory from Ø to 2Ø. ii) will dump memory from Ø to 13F. iii) will dump memory from 14Ø to 1FF, and iv) will dump memory from 2ØØ to 33F.

TIMER

F800:804C	VTMR2
F800:84E8	COUNT
F800:84EA	SECONDS
F800:84EC	TCONVRT

Timer #2 is set to generate an interrupt 50 times a second. TCONVRT is initialized to 50. If the timer frequency is changed then TCONVRT should be modified to reflect this change. The interrupt service routine will increment seconds after TCONVRT counts. The interrupt routine also checks the time since the floppies were last used. If it has been more than two minutes the drive motors are shut off.

FUTURE

The monitor is copyrighted. We don't want our software used on a competitor's hardware. However you should feel free to incorporate any or all of the monitor into your software, as long as it is distributed only to SLICER owners.

An important monitor function is to make programming easier. Some routines which a program may need are already written and stored in EPROM. We hope this will encourage third party software. The monitor will grow and change for some time yet. We intend to keep the entry points and variables described here fixed, so that software which uses the monitor will work with future versions. Room has been reserved for some additional entry points. If you find an undocumented routine in the monitor that you think is useful, let us know. If we agree we will assign it a permanent entry point. Also let us know if there is something you think should be added to the monitor. We are currently working on Winchester routines.

Address parameters are expected in the form SEGMENT:OFFSET. The absolute location is **16*SEGMENT+OFFSET.** SEGMENT can be a 16 bit number, or a segment register, or it can be omitted. If SEGMENT is omitted then a default segment register is used. The default segment register for ml in the dump command is DS.

Examples:

i) D123
ii) DCS:45
iii) D1234:5678

i) dumps memory starting at offset 123 in the current data segment. ii) dumps memory starting at offset 45 in the current code segment. iii) dumps memory starting at offset 5678 in segment 1234, that is absolute address 12340+5678=17AB8. Some commands use more than one segment. The move command, Mml,t,m2 (move memory from ml up to t to m2) defaults to ml in the data segment and m2 in the extra segment.

More than one command may be typed on a line. The commands should be separated by spaces. (There are situations where the space is not necessary, but the rule 'always use spaces' is easier to state and remember.) Any commands after a command which requires additional input (S and Xr) are ignored.

DETAILED COMMAND DESCRIPTIONS

In the command descriptions, default for m1(m2) is 'where the last command to use m1(m2) left off'. Default segment for m1 is DS. Default segment for m2 is ES.

Dm1,t — DUMP HEX/ASCII

Dump memory from m1 to t in mixed HEX/ASCII format. Default for t is m1+13F. t must be in the same segment as m1, putting a segment prefix on t will cause an error. Use cntl-S to stop scroll, and cntl-C to abort the command.

DQm1,t — DUMP ASCII ONLY

Dump memory from m1 to t in ASCII only format. Default for t is m1+4FF. t must be in the same segment as m1. Use cntl-S to stop scroll, and cntl-C to abort the command.

Sm1,b1,...,bn,. — SUBSTITUTE MEMORY

Substitute memory at m1. b1 through bn are byte values which are put into memory starting at m1. If a byte value is omitted then that memory location is left unchanged. Data may be continued over several lines. When CR is typed the monitor displays the current memory location and its contents. Another string of data may be typed. Replace one of the bytes with a period to exit this command.

Pm1,text — PUT TEXT INTO MEMORY

Put text into memory at m1. Everything after the comma is interpeted as part of the text string. So do not try to put other commands after the P command.

Fm1,t,b — FILL MEMORY

Fill memory from m1 to t with b. b is a byte value which defaults to Ø. t must be in the same segment as m1. t defaults to FFFF, 'the rest of the segment'.

Mm1,t,m2
RMm1,t,m2 — MOVE MEMORY

Move memory from m1 to t to m2. t must be in the same segment as m1. Default for t is FFFF. M starts the move at m1 and goes up, RM starts at t and goes down. Use RM when you move a block to higher memory and the new block overlaps the old.

Cm1,t,m2 COMPARE MEMORY

Compare memory from m1 to t with m2. t must be in the same segment as m1. Default for t is FFFF. Memory locations with differences and their contents are displayed. Use cntl-S to stop scroll, and cntl-C to abort the command.

Ip INPUT

Display word input from port p. Default for p is the last port used by the I or O commands.

Op,w OUTPUT

Output word value w to port p. Default for p is the last port used by the I or O command. Default for w is the last word used by the O command.

Hw1,w2 HEX MATH

Display the sum and difference of w1 and w2. Default for w1 and w2 is Ø.

X EXAMINE REGISTERS

Display the current contents of the CPU registers.

Xr EXAMINE/MODIFY REGISTER

Display the contents of register r. A word value may be typed to change r. A CR leaves r unchanged.

Gg,b1,b2 GOTO WITH BREAKS

Put breaks at b1 and b2 then start executing code at g. Default for g is the current contents of CS and IP. CS and IP are changed when g is read. Default segment for b1 and b2 is CS after it has been changed by g. Type a period in place of a break point to set the same break as the last G command. If a break point is omitted then it is not set. Suppose the last G command was G5,3F,67. Then G8 sets no breaks. G8,. sets one break at 3F. G8,,. sets one break at 6F. G8,.,. sets two

Tw TRACE

Trace program execution starting at CS:IP for w steps. Default for w is 1. Instructions which modify segment registers are not counted as steps. After each step the CPU registers are displayed. Use control-S to stop scroll, or control-C to abort command. Note that the starting point cannot be entered as part of the command. Use XCS and XIP to set the starting point.

Uw UNTRACE (STEP)

Execute w steps starting at CS:IP then return to the monitor. Default for W is 1. Instructions which modify segment registers are not counted as steps.

MT MEMORY TEST

Go to memory test subprogram. The memory test does not return, it exits to the power up routine. If lower memory has been changed by the memory test you will have to type CR to get started again.

DMm SET DISK MEMORY AREA

Set memory area for disk transfers (read or write) to m. Default segment and offset for m is the current disk memory area.

DSb SET DISK SECTOR

Set sector for disk transfers to b.

DNb SET DISK NUMBER

Set disk number for disk operations.

DTw SET DISK TRACK

Set disk track for disk operations.

DZ ZERO DISK (RECALIBRATE)

Set flag to recalibrate before next disk operation. Should be done after disk errors.

DP DISPLAY DISK PARAMETERS

Display the current setting of the disk operation variables.

DRb1,b2 READ DISK

Read the next b1 disk sectors, assume the disk has b2 sectors per track. Default for b1 is 1. Default for b2 is 'what was used for the last disk operation', (26 initially).

DWb1,b2 WRITE DISK

Write the next b1 disk sectors, assume the disk has b2 sectors per track. Default for b1 is 1, for b2 whatever was used for the last disk operation.

SS SHOW SEGMENTS

Display the contents of the segment registers. IP and the last used IO port are also displayed.

B BOOT DOS

Boot CPM86 on the single density eight inch floppy in drive Ø. We hope to add other disks and other operating systems later.

Z SLEEP

Put processor to sleep. (Don't worry, any console input will wake it up again.)

& AGAIN

Repeat command line. Use control-S to pause, and control-C to abort.

COMMA
SPACE NOTHING

Can be used as separators on the command line.

NOTE: An error code is displayed when a disk error occurs. See the section 'READ/WRITE DISK SECTOR' under 'MONITOR INTERFACE' for a description of disk errors.