Sol-20 Restoration

Power Supply

The power supply regulator PCB is Rev B with the Rev C modified crowbar circuit implemented by the original owner or the factory. The transformer is not one of the problem transformers mentioned in Change Notice #9.

The power switch was broken (acted like a momentary switch) . I replaced with it with an OSLO SLA5A6V1M9 switch. This is a lighted version of the power switch, but I did not connect the light. It also has both normally open and normally closed contact sets. The N.O. contacts are used. The switch has a red button very similar to the original switch. The original wiring had a 3/16" quick-disconnect for the short cable that connects the fuse to the switch. In the original, the 1/4" tab on the switch was cut to make the 3/16" push-on fit. Instead, I replaced the 3/16" push-on with a 1/4" push-on as it should be. Note: The SLA7A12V1M9 can also be found (e.g., onlinecompoments.com) and works well.

The ground prong on the C14 AC power inlet would not stay in place and the other two blades had flattened over time and power cords would not connect reliably. Standard C14 inlets were too big to fit into the rear panel cutout from the inside. I purchased a European brand (Schurter 6100.3100) that was very close to fitting, though it did take just a bit of filing to make it fit (top to bottom dimension)

While performing these power supply modifications, the nylon screw that clamps the TIP41 transistor to the heat sink broke off. This part's tab must be isolated from the heat sink (chassis), so I installed a nylon shoulder washer and metal screw from a heat sink mounting kit.

The output of the regulated 5 volt supply is a bit low under load of the main board and keyboard (just under 4.7v). *Should I adjust this?*

Follow up: About a month after putting everything back together, I decided I should work on the low 5v supply. The resistor divider used to measure the 5v output and control the inverting input of an op-amp uses the -12 volt supply for the low side of the divider instead of ground. This makes the 5v output dependent on the -12v supply. The -12v supply is at -11.65v (within spec), but the 0.35v error contributes substantially to the 5v supply being low.

I was able to remove the large, horizontal filter capacitor (8v supply filter for the S100 bus) and gain access to the lower resistor in the 5v sense divider (R12, 4020 ohm, 1%) without having to remove the power supply from the computer. I soldered a 100K resistor in parallel with R12 which put the 5v supply back up to 5v +/- .05v at the main board connector. The supply is down to about 4.85 at the keyboard.

Main Board

The main board is Rev "E" from the factory. ECN's from E to F and from F to G have been implemented by the original owner or the factory. The ECN from G to H has not been implemented. This last ECN “enhances the reliability of the opto-isolator which couple a current loop device to the Sol.” I do not plan on applying this update.

When running step 40 on page III-29 of the assembly manual, the screen should have 16 lines of 64 characters of an alternating 90909090… sequence. This test should be run with the personality card removed and no external RAM in the system. Upon reset, the CPU fetches a RST 7 instruction (0xff) since nothing is responding to the memory read cycle. The RST 7 pushes the return address (0x0001) on the stack and then jumps to location 0x38. Here, another RST 7 instruction is fetched. This pushes the return address (0x0039) onto the stack and loops right back to 0x38 and repeats. This loop quickly writes 0x00, 0x39 to all of memory, including video memory from 0xCC00 through 0xCFFF. This generates the “test pattern” that should be seen if the CPU, the video section, and video RAM are working properly.

The Sol-20 displayed garbage in the 1st 32 characters of each line and the proper “909090” pattern in the second 32 characters of each line. This was due to buffered A5 from the CPU being stuck high (lower 32 bytes of each line in RAM were not being initialized). U68 is a 8T97 that buffers this signal. The output pin for A5 had broken off. I replaced the 8T97 with a 74367. Though not identical (slightly less current source/sink), it’s a very close replacement.

Two 2102 RAM chips were corroded enough to have broken legs. They have been replaced. On occasion, video problems occur coincident with cursor flash. The problem included treating the entire video RAM as a “cursor” at times (i.e., the entire screen flashes white and black). Bit 7 of video memory in U21 is the cursor flag. This bit was erroneously reading out as a one sometimes. Pulling this RAM, cleaning the pins and reseating the RAM fixed this problem. Most likely the RAM chip itself is fine and the issue was a pin and socket contact problem.

Update: About 9 months after restoration, the computer would not power up into the SOLOS monitor. Typically a single line of garbage would display on the screen. It appeared that the CPU wasn’t able to fetch and run instructions from the personality module. The problem ended up being U68 once again, however this time, it wasn’t the chip that failed – it was poor contact between the IC legs and the socket. A few insert/remove cycles cleaned the contact points enough to for the computer to then boot reliably.

Keyboard

The keyboard did not work at all. Upon removing the keyboard mechanism from the keyboard PCB, disintegrated foam pad debris immediately fell out. New pads will have to be purchased, made, or pulled from a Sun type 4 keyboard. The keyboard works when “typed” on with bare fingers directly onto the PCB.

As of 2014, Sun type 4 keyboards are still available. Not many show up on eBay, but an Internet search shows results. Many are listed under the following part numbers: 320-1005 and 320-1018. This is not the official model number, but may be a Sun internal part number. The number is present on the product label, but it’s meaning is not identified. The 1018 number is a “French Canadian” keyboard. It seems to be the most readily available. These show up on numerous websites for $19.98. I think they're all actually all coming from the same place. The one I purchased through ebay came from "Memory Ten." The keyboard I received was in perfect, brand-new like condition. The pads looked great and user a newer style of foam that does not break down so quickly.

After replacing all pads, some keys still did not work. Multiple attempts at fixing the problem positions still did not fix every bad key. I then took fine grit sandpaper (600) and smoothed all PCB "capacitor" pads. Though I had not noticed or felt any sort build-up on these pads, the sanding fixed the remaining keyboard problems.

Serial Port

The DB25 connector is wired as DTE. The S4 switch settings for word size in the manual are incorrect. Use the following instead:

Word Size	S4-2 (WLS 1)	S4-3 (WLS 2)
8 bits	OFF	OFF
7 bits	ON	OFF
6 bits	OFF	ON
5 bits	ON	ON

Cassette Interface

Saving to cassette returned to the SOLOS prompt after about 4-5 seconds regardless of the number of bytes specified to save. Nothing was written to tape, though the idle marking tone was recorded. I suspected that I was seeing the 4 second startup delay followed by writing all of the memory range specified very quickly because the transmit data ready bit from the UART was stuck high due to a PCB or UART problem. A quick test routine proved this to be the case.

Using a meter, I didn't find continuity from the TDRE pin on the UART to the line it connects to on the input bus. This explains why the signal always appears asserted. I pulled the UART, cleaned up numerous ugly pins, reinserted it, and then the TDRE pin showed continuity to its input bus line. Thought I had an easy fix, but now the save operation hangs forever.

The hanging forever problem was U111, the 4019 selector used to route the proper baud rate clocks to the UART (based on the 1200 or 300 baud selection). It had a pin bent up under the chip. The bent up pin carries the clock used for 1200 baud to the UART. The original owner must have only used 300 baud for cassette and wondered why 1200 baud never worked.

At this point, the UART was properly accepting and clocking out data, however, the cassette audio output still remained just the marking idle tone. This ended up being still more corroded pins on the UART that were preventing transmit data out from the UART from making it to the modulation flip-flops.

Audio now seemed to record properly, but the cassette still does not load properly. I have not been able to load .wav files saved by other Sol owners either. The operation does not work even if I save and play back to/from a computer.

As received, the Sol's record out level was jumpered for line level. I changed the output to mic level in order to use the Sol with the tape recorder I have for the Altair. However, when later reading trouble shooting tips for the cassette interface, it was recommended that mic level be avoided. I have since restored the record output to line level. However, this has not fixed any of the symptoms described.

An audio input problem still exists as I cannot load any audio file successfully. While the recorded output looks reasonable in a sound file editor, until I can test the recorded output by loading it back in, I can't be sure the audio output is correct either. *The next step will be to use the test routines I've written for the Altair and use the Sol cassette hardware at 300 baud to try and isolate the problem(s).*

Follow up: I updated my Altair cassette test routines to run on the Sol-20. These routines write a test pattern of U's (55h) to the tape with a 1.5 character idle time in between each U (300 baud assumed). The routines also provide the option to read from tape and display received data on the screen. On a side note, it took a while to get the routines to work as expected because the "SET TAPE x" command in SOLOS does not actually OUT to the hardware when the speed command is issued. Instead, it simply updates a variable that is looked at by the SOLOS cassette routines. I had to include the OUT to 0FAh with bit 5 set in the test routines in order to set the hardware for 300 baud.

I recorded 300 baud audio out from the Sol into a .wav file on a PC. Using a sound file editor, I discovered that even though 1's and 0's were being properly modulated, the sequence of 1's and 0's was incorrect for the pattern of U's that was being written. Further experiments, looking directly at the transmit output of the cassette UART, proved that the transmit side of the UART was defective. Very reasonable looking 300 baud data was coming out of the UART, but it was not the correct data. When swapping the serial port and cassette UARTs, the problem followed the UART.

Additional tests proved that the receive side of the cassette UART was operating properly. So unfortunately, a new UART still won't fix the problem that the Sol can't read tapes (.wav files) written from a good machine.

Looking at the cassette demodulator circuit, I found the AGC op-amp was railed to the -12v supply with audio data appearing as a slight ripple near -12v. The 2N4360 JFET that acts as the gain control shunt on the inverting input of the AGC op-amp was providing a small DC bias into the input divider and causing the problem. Looking more closely at this JFET on the main board, it was clear this transistor had been replaced by the end-user at some point. The soldering job was mediocre and a 2N5460 had been substituted for the original 2N4360 (a reasonable replacement, however). I cut the JFET off the board and the AGC op-amp now behaved at expected.

After removing the defective JFET (I didn't have a replacement available yet), I attempted to load a .wav file of "TARGET" into the Sol. The AGC op-amp limits the input to the op-amp comparator stage to about a 2v positive peak. Since I had disabled the AGC by pulling the JFET, I manually adjusted the audio output from the PC to keep the signal level at the op-amp comparator stage to about 2v PP (1v positive swing). I was then able to successfully load a pre-recorded 1200 baud audio file for the first time. I successfully repeated the experiment with several other 1200 baud program recordings.

The next step is to install a new JFET and cassette UART and repeat much of this process.

Follow up: I found a NTE326 JFET at Fry's which is a reasonable substitute for the 2N4360. I installed the JEFT and with the new part, the AGC op-amp stage works as designed. I was able to load several pre-recorded cassette .wav files (1200 baud) at substantially different volume levels. The receive side of the cassette interface now appears to be fixed!

I replaced the defective UART with an AMI S1883 part (alternate for the TMS6011 shown in the schematic) that matches the serial port UART in my Sol-20. Now I am able to both save to and read from cassette at both 300 and 1200 baud. Compatible UARTS include AY-5-1012, AY-5-1013A, TMS6011, AMI S1883, COM2502, COM2502H, COM2017, COM8502, COM8017, Western Digital TR1402/1602/1863/1865, Intersil 6402.

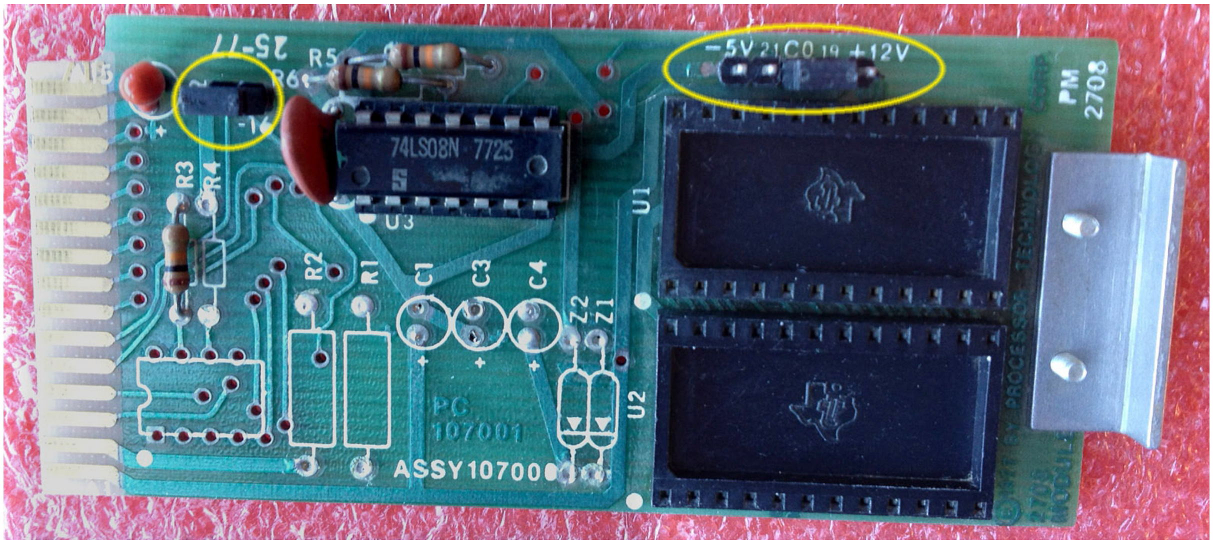
Personality Module

The personality module is the PM-2708. This module can be configured to work with a 2K 9216 ROM, a 2K 8316 ROM, a 2K 2716 EPROM (and therefore ½ of a 2732), or two 1K 2708 EPROMs. The board was configured for, and populated with, a 2K 9216 ROM containing SOLOS 4.1.

These chips are not directly interchangeable without changing jumpers on the board to accommodate the differences in pin assignments and power supply requirements as shown in the table below.

	9216	**8316**	**2716**	**2708**
Pin 19	+12v	A10	A10	+12v
Pin 21	A10	CS3	Vpp (+5v normally)	-5v
Jumpers	-16, C0 to 21, 19 to +12v	-16, C0 to 19	-16, C0 to 19	-5v to 21, 19 to +12v

To make the board easy to use for any of the above combinations, I cut the default jumper traces on the PCB (-16, C0 to 21, 19 to +12v) and installed header pins in each jumper position. This allows use of standard push-on jumpers to select different options. Note that using 2708's requires installation of the parts missing on the PCB shown below in order to generate –5v for the the 2708's.



Personality Module Jumper Modification

Sense Switches

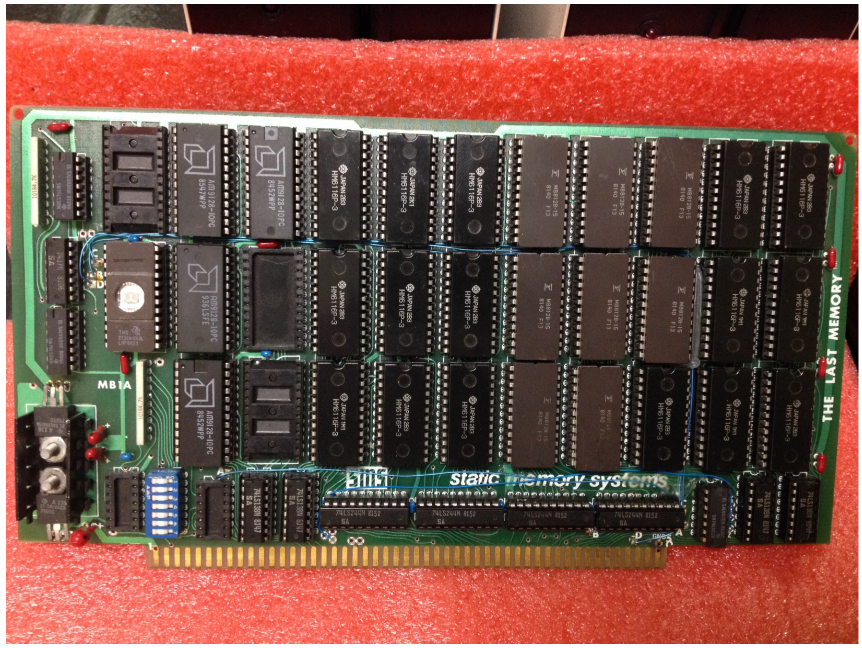
In trying to get the “Music System” up and running, I needed to set the sense switches to all zero. Until I did some digging in the schematic, I didn’t realize the Sol-20 even had sense switches! Like the Altair, they are read at I/O address 0xFF. On the Sol-20, the sense switches are an 8 pole DIP switch near the serial port. The switches were all in the “1” position (switch open). I moved them all to the “0” position (closed) and turn the computer on. No prompt from Solos. I tried a few combinations and found the computer would run if bit 7 or bit 6 were turned set to zero, but fail to run if any of bit 5-0 were set to zero.

Testing reveals there is no power on U57, the 7406 that is responsible for enabling the read of the sense switches. Because of this, the sense switch IN operation is constantly "on." When a switch is closed, it is driving its data line on the internal bus low 100% of the time. This interferes with all other data fetches on the internal bus. An open switch results in a “1” on the data bus by going hi-z and letting the bus pull-ups drive the bus. This is why setting all sense switches to “1” (open) doesn’t cause a problem.

The power problem for the chip ended up being in pin 14 of the socket. Unfortunately, proper repair means disassembly of the machine to get the main board out to replace that socket. As a temporary fix, I tied pin 14 of the 7406 directly to +5v. This fixed the sense switch problem.

SMS 64K RAM Board

Several 64K RAM boards using 2K x 8 static RAMs were made for S-100 computers. Some, like the SMS 64K (“The Last Memory” board from Static Memory Systems), make it easy to leave 2K holes in RAM and substitute 2716 EPROMs in 2K blocks as needed. For this particular board, removing a chip is all that is required to make the board not respond to address range of the removed chip(s). This makes these boards ideal for working with the Sol-20.



As shown in the picture, the sockets at C000-C7FF and C800-CFFF are empty to leave room for SOLOS at C000-C7FF and for RAM and video RAM at C800-CFFF. The socket for F800-FFFF is also empty as I have a Micropolis FDC in this address space. Finally, an EPROM is installed at F000-F8FF in which I have burned an Intel hex file loader. RAM is installed from D000-EFFF, but most software won't know about this RAM or be able to use it.

The board does not generate a wait state for the RAMs, so the EPROM used must not require a wait state either. This means a 250ns EPROM should be used. It is much easier to find a 250ns 2732 than a 2716, and a 2732 can be used without problem in this application. The simplest approach is to program the same code into both the lower half and upper half of the 2732.