The Radio Shack Model 100 and Y2k

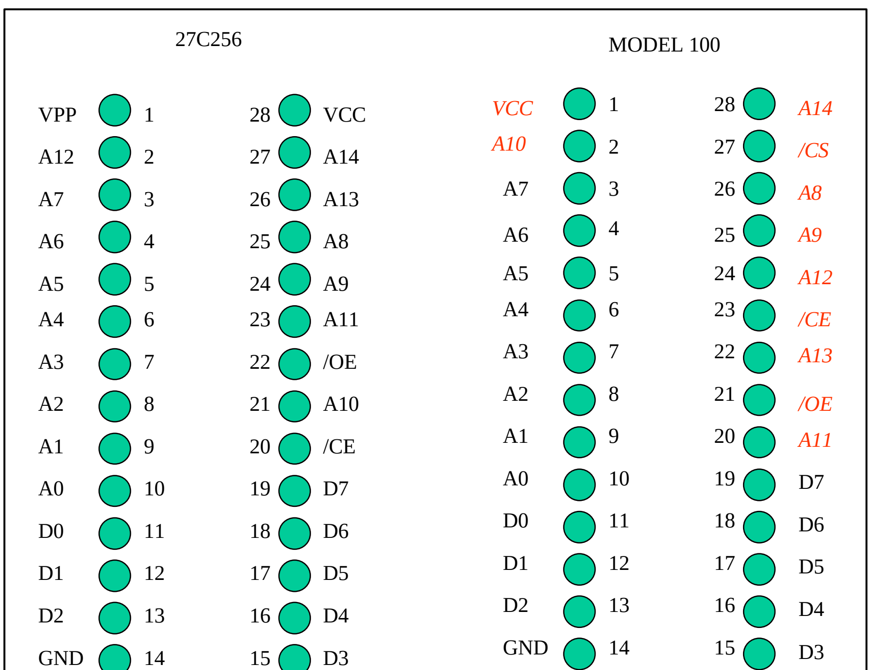
The Radio Shack Model 100 works internally with only the last two digits of the year such as 99 for 1999 or 00 for 2000. Therefore, applications written in its Basic interpreter must always work with just the last two digits for the year. The century mark is programmed into a mask programmed ROM chip. This century mark is only displayed when the Main Menu is on the LCD screen. The Y2K leap year problem is nothing new because the M100 does not deal with ANY leap year, pre Y2K or post.

There are two solutions to the century mark in ROM problem. Solution one is to ignore the 19xx and life will go on. Solution two, for those of us who like a challenge, is to patch the ROM chip. This is the approach I took, just to see if it could be done.

I had a copy of the Technical Reference Manual for the Model 100 and this was a good starting point. I also have a Disk/Video interface sitting in a closet on which I had stored a disassembler that I wrote many years ago. The first step was to see if I could find where the 19 in the date came from.

I started with the list of service routine entry points. The Main Menu entry point was listed as 5797 Hex and I started there with the disassembler. The disassembly ran down the 19 in the routine to display the date. Starting at address 5A52 is a sequence of instructions 'MVI A,31' with the 31 Hex (ASCII 1) being at 5A53. The 31 Hex needs to change to 32 Hex (ASCII 2). There was also a sequence of instructions 'MVI A,39' at 5A55. With the 39 Hex (ASCII 9) being at 5A56. The 39 Hex needs to change to a 30 Hex (ASCII 0). Those are the only changes needed to the ROM.

I then looked at the schematic of the Model 100 and the ROM chip used. A quick inspection revealed a problem. The part used in the Model 100 and a 27C256 32Kx8 Low Power CMOS EPROM have pin-outs that differ on several pins. The pin-outs of the two devices are shown below:



Non one-to-one pins shown in red on Model 100 Socket

The connection list is as follows:

27C256	M100 Socket
1	N/C
2	24
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	27
21	2
22	21
23	20
24	25
25	26
26	22
27	28
28	1
N/C	23

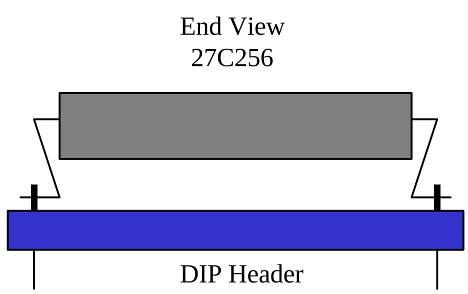
I obtained the parts from JDR Microdevices (www.jdr.com). The parts were as follows:

Quant.	Part No.	Desc.	Price.
1	27C256-250	32kx8, 250nS Low power EPROM	$3.09
1	ICC28	28 pin DIP header	$1.99

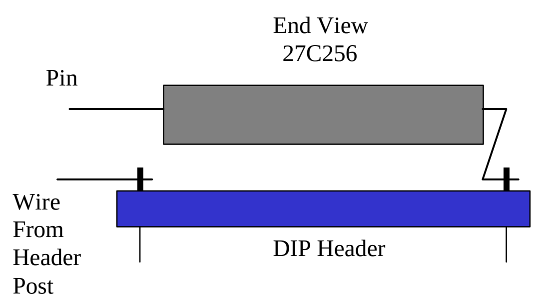
Before building the assembly, I programmed the 27C256. I first needed the code from the Model 100 ROM chip. For this, I wrote a quick and dirty program in M100 Basic to peek each byte of ROM, and send it out the COM port. I had a PC receiving the bytes over RS-232 and writing them to a binary file. I then loaded the binary image into an S4 Dataman programmer and used its editor to change the required two bytes. I then wrote the new image to a disk file and programmed a 27C256 with the new image.

The hardware plan was to use the DIP header to re-map the M100 ROM socket to fit the 27C256 part. The DIP header would then plug into the M100 ROM socket. The overall module would have to be kept as thin as possible to avoid contact with the back of the keyboard assembly.

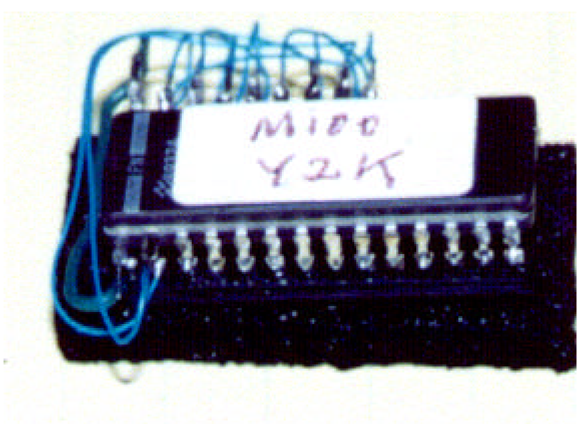
To make the assembly, I put the DIP header in a piece of stiff anti-static foam and clamped it in a small vice. I then installed wires on the posts of the DIP header on the pins that were not a one-to-one correspondence with the 27C256 making sure to allow enough wire to reach the appropriate destination. I then bent the pins of the 27C256 to mate with the DIP header. The pins with one-to-one correspondence were bent in to connect to the DIP header, and the remaining pins were bent straight out to avoid contact with the DIP header (use caution when bending the pins!). The rest of the assembly was just soldering the one-to-one pins from the 27C256 and then making the appropriate interconnects with the wires from the pins of the DIP header to the bent out pins of the 27C256. The Vcc connection was made with 22Ga. wire and all the other connections were made with 30Ga. wire wrap wire. The assembly is pictured below from several different angles:

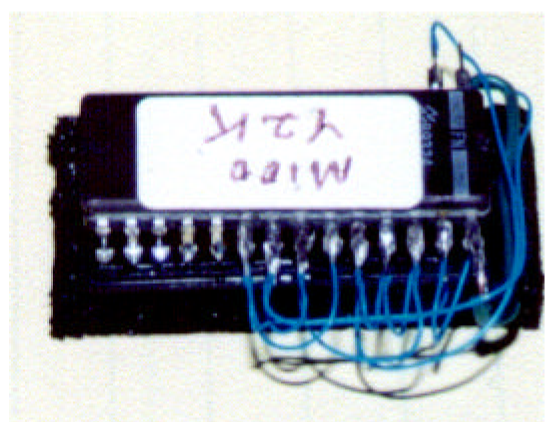


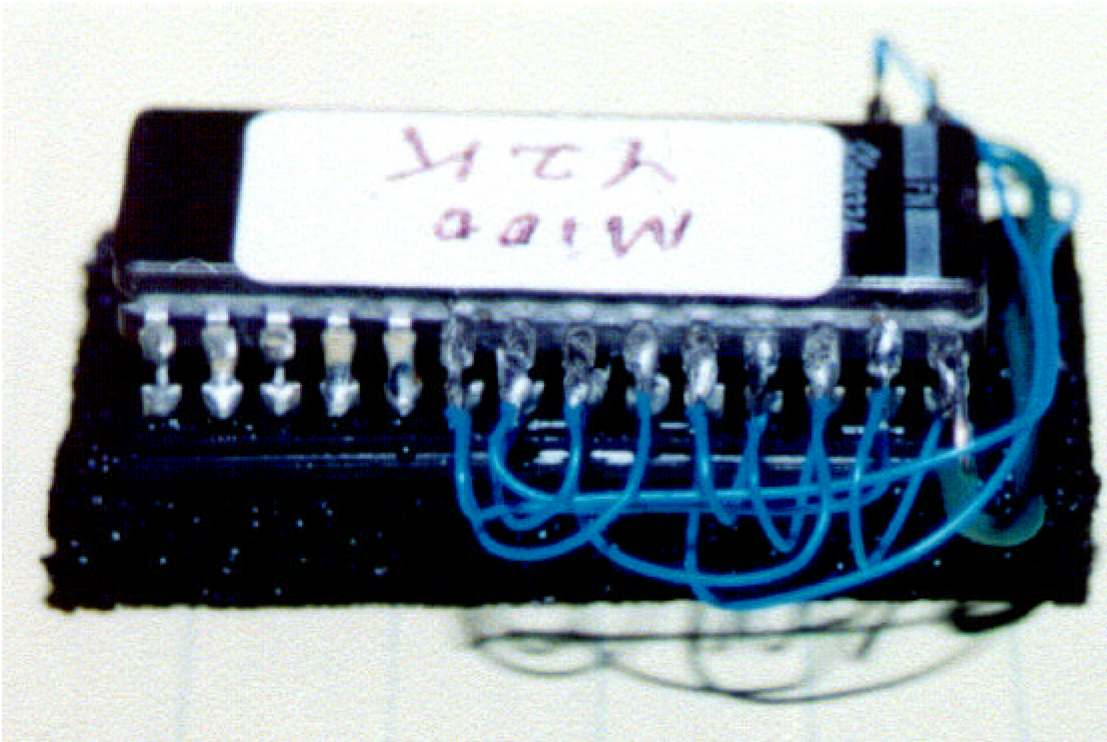
One-to-one PROM to Socket Connections



Non one-to-one PROM to Socket Connections (Left Side)







I then pulled out the original ROM and plugged in the adapter for the Y2K patch. I had to be careful plugging in the Y2K module as the pin 28 corner of the assembly is not as rigid as the other corners. I just used a small screw driver to put pressure on the DIP header below the chip.

After insertion, the menu came up in the 20xx century. A successful patch!

Why go to all this trouble? 1) It is a fun project if you like this sort of project. 2) If anyone looks at your menu, they won't freak about Y2K when they see the wrong century!

Anyhow, if you set out on this path, you will need an PROM burner that can program a 27C256 part. I can provide the image. A good soldering iron is also a must. Since you are soldering right on a CMOS part, you want a grounded tip and good temperature control to get on, make the connection, and get off before things get too hot. I used a Hakko 928 60W iron with a very small tip. This iron has a temperature dial and holds the tip temperature within ± 2 Deg. F.

Good luck if you choose this path and may your Model 100 provide you with many more years of service!

As one further note, This applies to a Model 100 only and not Model 102's. I suppose the 102 patch would be similar, but I don't know if the address would be the same but I would bet they would be close. The 102 does not roll the years over from 12/31 to 1/1, so I know their code is different somewhere. They might have a more standard ROM part which would help the assembly process, but I have never seen a Technical Reference Manual for a Model 102 so I don't know.