

Some parallel ASCII keyboards

[Post Reply](#)

First unread post • 8 posts • Page 1 of 1

MMcM

21 Feb 2021, 05:37

Trying to fill the Covid lockdown with interesting vintage boards to find, restore, and convert means a steady stream of ones that use a simple parallel ASCII protocol. The idea is this: when a key is pressed, it is converted to US-ASCII by the keyboard and output as a 7-bit (sometimes wider) signal together with a strobe pulse.

This was quite common in the late '70s to early '80s inside / alongside terminals and similar equipment. Common enough that even back in those days there was a steady supply from surplus / overstock. So they were picked up for use by homebrew computers, where they were ideal because very little processing power or extra electronics is needed to get a character at a time into the program encoded the same way it would be in a file.

In fact, the very first issue of *Byte* magazine had an article, "[Deciphering Mystery Keyboards](#)". It pretty much only discusses this kind of keyboard. There is no mention of bit-serial encoding, raw scan codes, or of make and break codes.

These keyboards don't convert to a full modern USB keyboard, with their limited character repertoire. I make them CDC serial devices, which allows for a typing test where you can see what you typed. Even with all these limitations, there is a lot interesting going on from an engineering perspective and aesthetically.

I have posted about some of these before:

- [Micro Switch SW-11234 / 55SW5-2](#)
- [SD-16234 / 99SD24-3](#)
- [Micro Switch SD-16534 / 91SD30-3](#)
- [Consul 262.3](#)

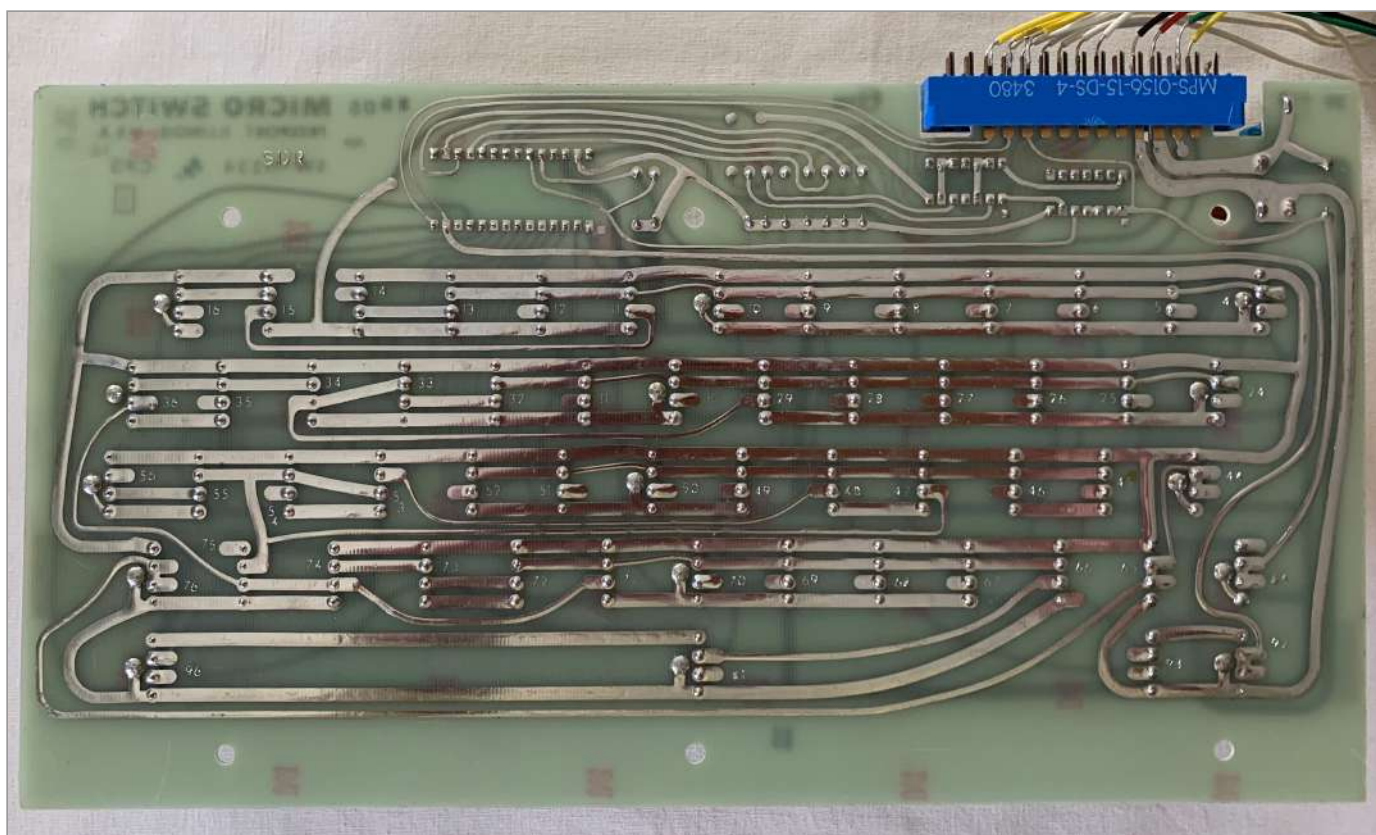
- [DEC LK01](#)
- [Beehive B100](#)
- [Amkey SNK-58](#)
- [Maxi-Switch 216004](#)

And in the meantime have worked with some more and made handling them more consistent.

The first two of those old posts, on the Micro Switch SW-11234 and SD-16234 did not have pictures or have lost them. So here are some of those boards here:



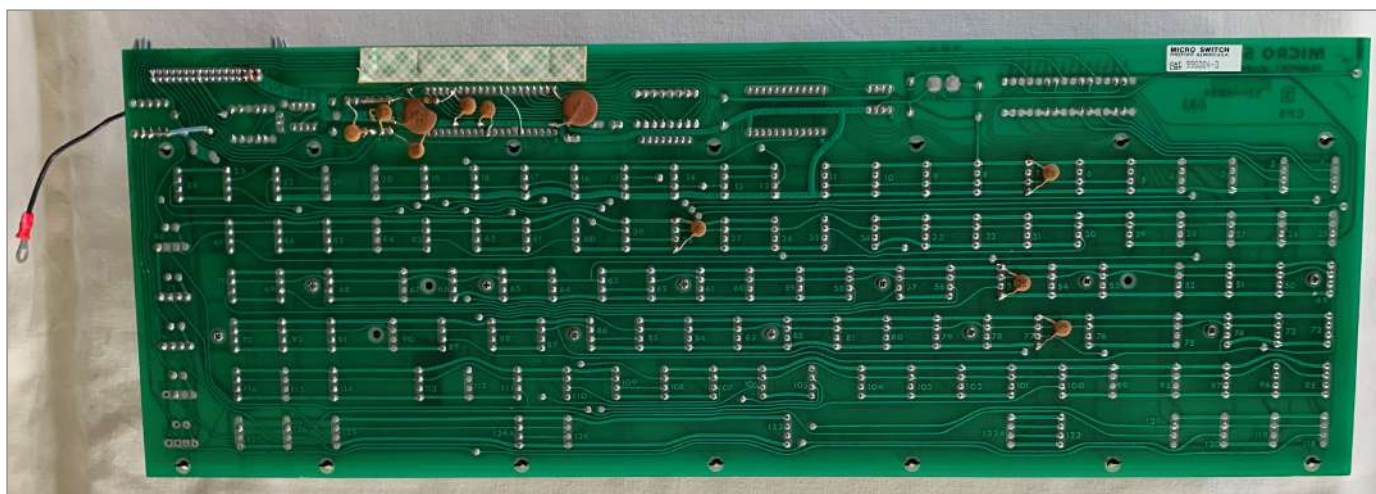
55SW5-2-top.jpg (1.77 MiB) Viewed 5651 times



SW-11234-bottom.jpg (1.61 MiB) Viewed 5651 times



99SD24-3-top.jpg (1.08 MiB) Viewed 5651 times



99SD24-3-bottom.jpg (1.26 MiB) Viewed 5651 times

As I said, by the time 55SW5-2 got to me it had passed through Godbout. So it isn't too surprising that the last of those posts, the Maxi-Switch 216004, was being sold these days as part of somebody's hobbyist system (and hence historically valuable). In response to its wood-grain vinyl contact paper, mr_a500 posted a picture of a TEC terminal also with a keyboard in a metal box with that. I have managed to get hold of a TEC EKA-9542, which looks to be substantially the same, but without the wood-grain option.



TEC-EKA-9542-case.jpg (332.25 KiB) Viewed 5651 times

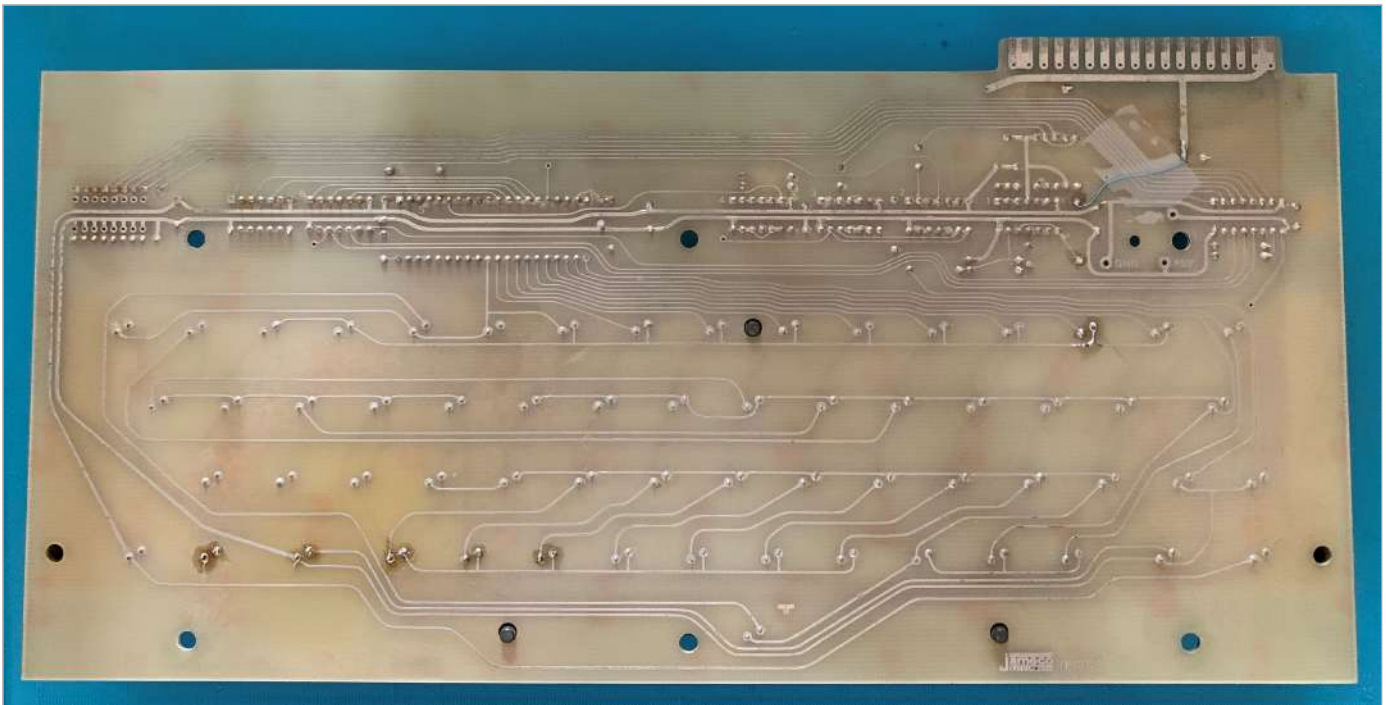


TEC-EKA-9542-back.jpg (152.83 KiB) Viewed 5651 times

And for a keyboard in a metal box that really was sold to hobbyists (with grid switches that I believe were OEM rather than surplus), a Jameco JE610.



JE610-case.jpg (277.8 KiB) Viewed 5651 times



JE610-bottom.jpg (1.28 MiB) Viewed 5651 times

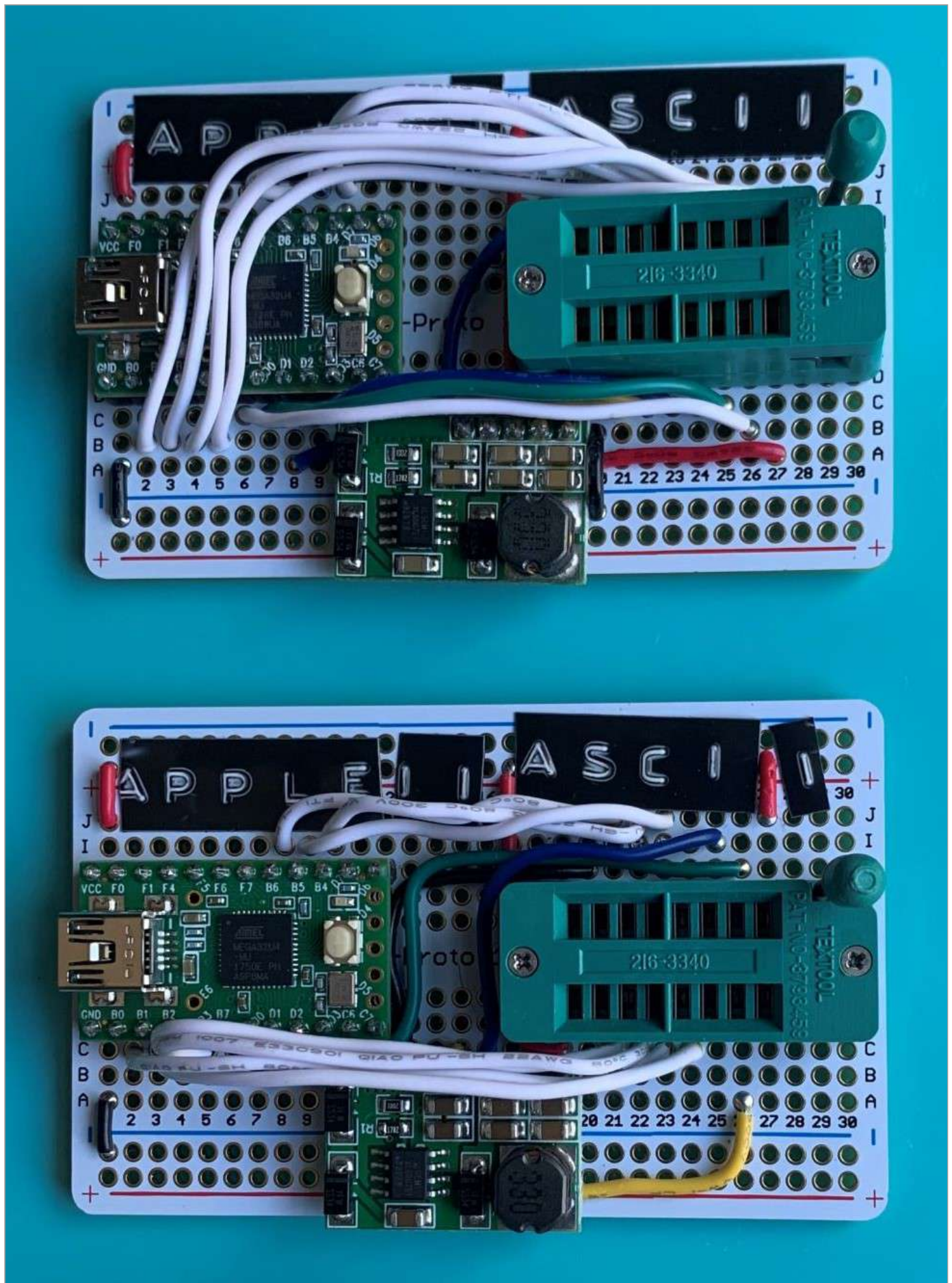
Jameco is still in business; I buy parts from them all the time because they are close enough for USPS to have them here in a couple days.

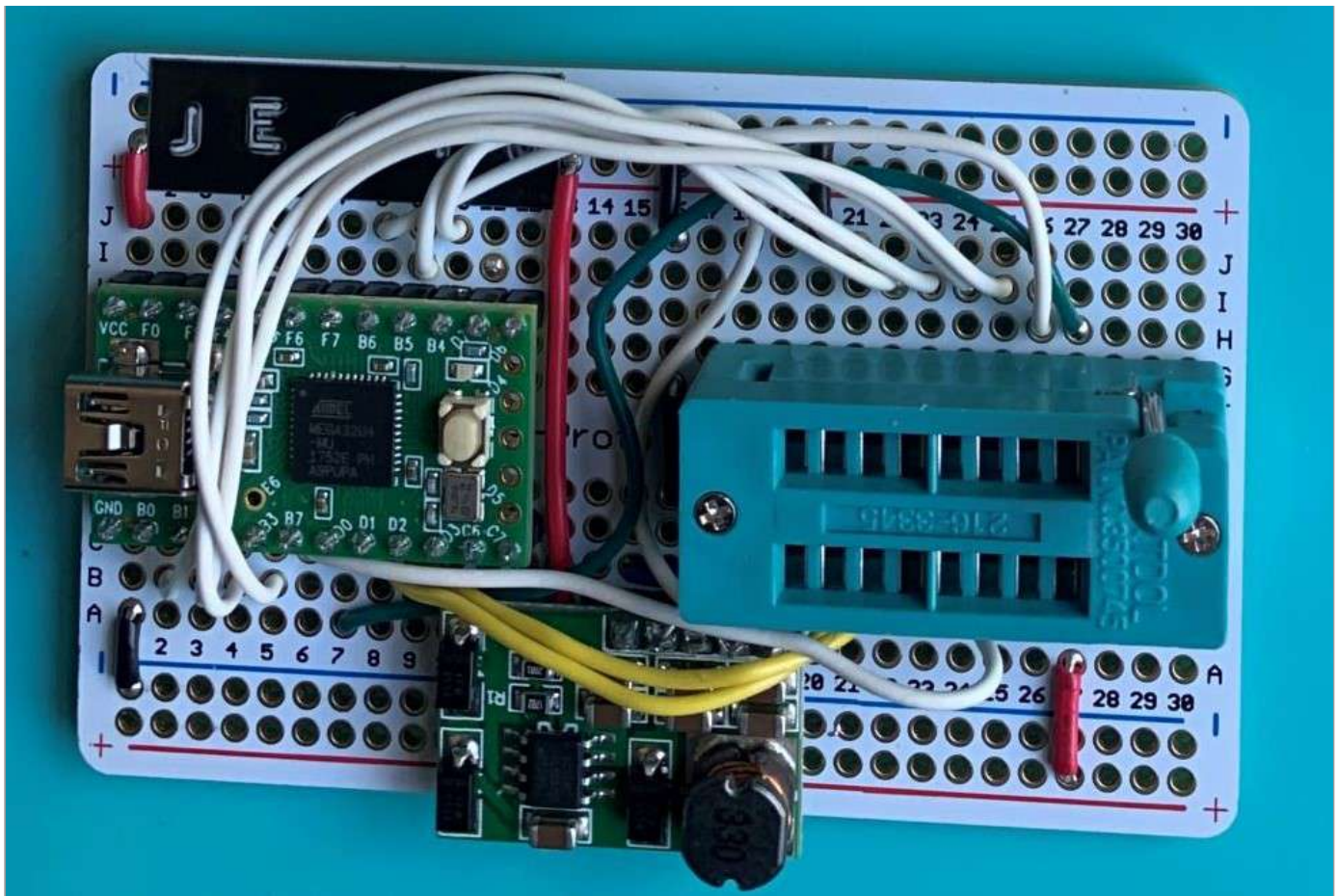
Connectors

7 bits of ASCII + strobe + power + ground = 10 signals at least, which rules out some smaller connectors.

Some hobbyist interfaces shared a 2x8 DIP with a ribbon cable. But no standard layout of the pins there.

Which leads to this situation.



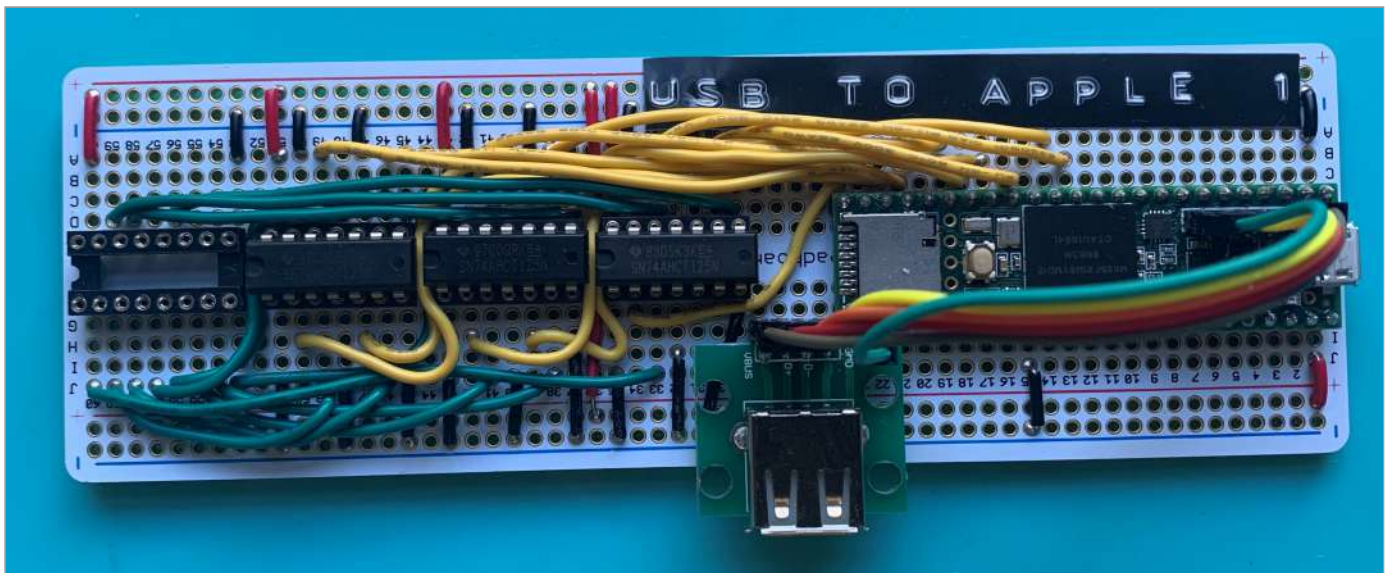


DIP16-Parallel-ASCII-converters.jpg (521.66 KiB) Viewed 5651 times

These are essentially all the same converter, just with the signals at different places on the DIP socket.

As these hobbyist keyboards have become separated from their computers and as revival versions of the computers have been made, there is a market for a converter from a modern keyboard to this classic interface. But since the pinout is variable, there needs to be a place to put in jumpers as in [this](#), which can still be bought [here](#) and on eBay, in two different configurations.

To go from a USB keyboard to parallel ASCII is even easier these days, if you use one of the Teensy's that has USB host support like this.



Apple-1-from-USB.jpg (872.09 KiB) Viewed 5651 times

Some level shifters are still needed since those have 3V GPIOs, though.

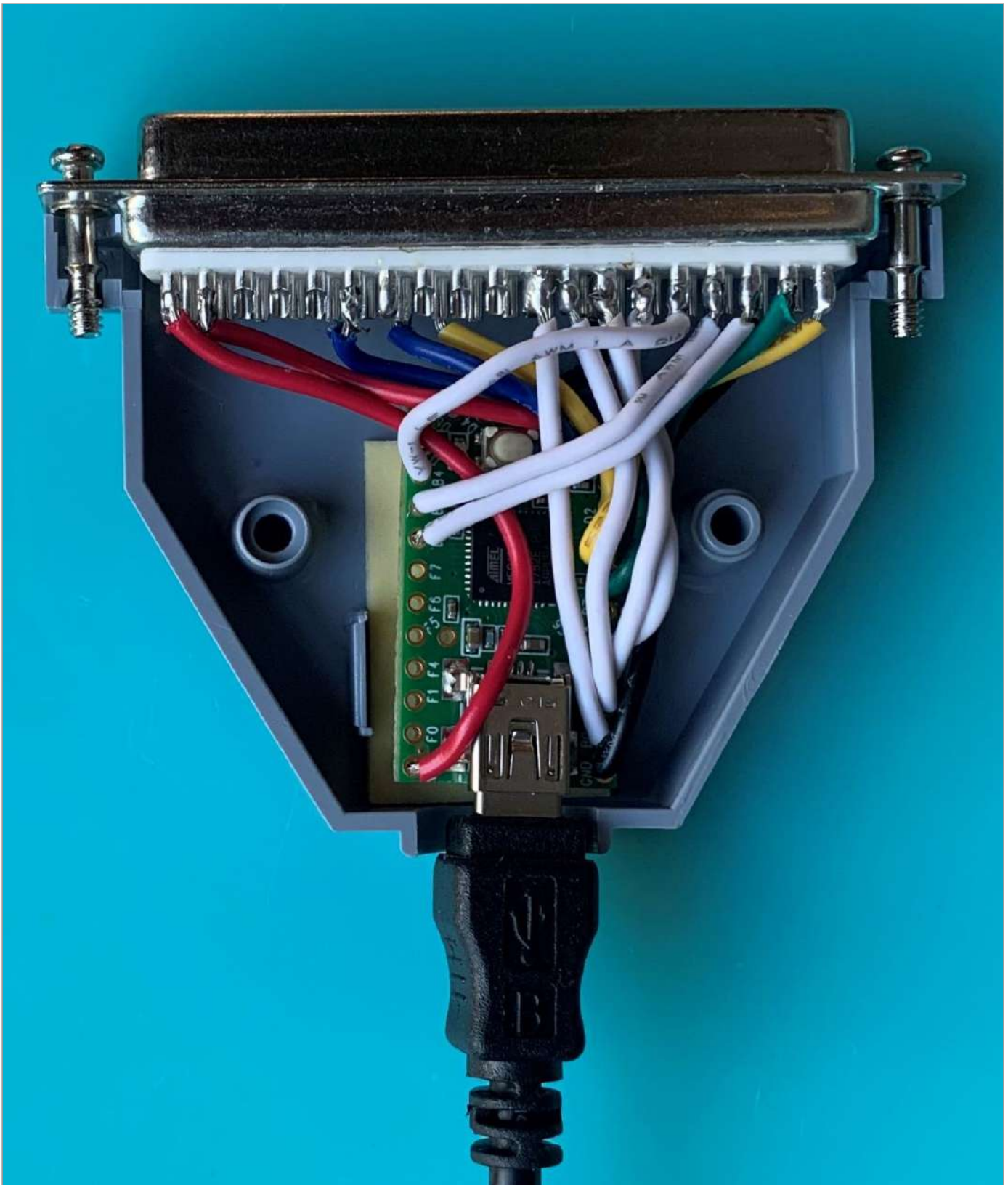
The Beehive B100 had a similar 2x9 DIP.

Other IDC ribbon cables with more standard headers had 20, 24, and 34 pins. The LK01 has a 40-pin header, but only 20 of the pins are connected, and those are in pairs, because it really only has the minimum 10 signals.

Sometimes a d-sub is used. The Honeywell BKBD has a female DB-25; having spent money on that connector, it is saved on the number of wires actually inside the cable that connected it to the terminal. The TEC EKA has a DC-37, but with only 17 connections for 15 distinct signals. That connector is so big that the Teensy converter fits inside the housing.



TEC-EKA-converter.jpg (332.39 KiB) Viewed 5651 times



TEC-EKA-converter-inside.jpg (274.62 KiB) Viewed 5651 times

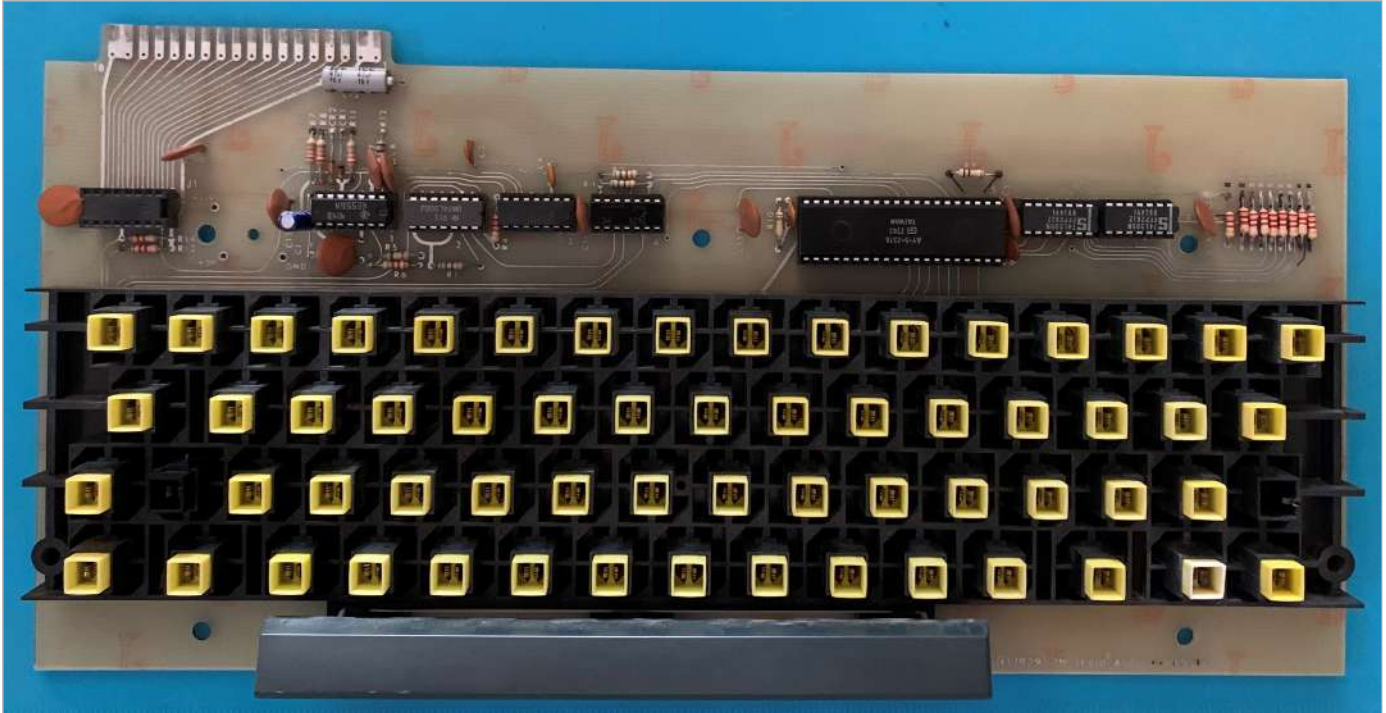
Strobe Direction

There is no firm convention for whether the strobe line is idle high or low, that is, whether the trigger is rising or falling. Of course, most keyboards would pick whichever seemed easier. But the JE610, which might be put to use inside a variety of computers, exposes both directions on different pins, saving the other end of the interface an inverter if they needed it the other way.

Encoder

The simplest way to scan the key matrix for key presses, encode an ASCII character taking shifts into account, and strobe the output was using a custom MOS encoder for just this purpose.

The JE610, for instance, has one:



JE610-top.jpg (1.4 MiB) Viewed 5651 times

Some examples of such MOS chips were:

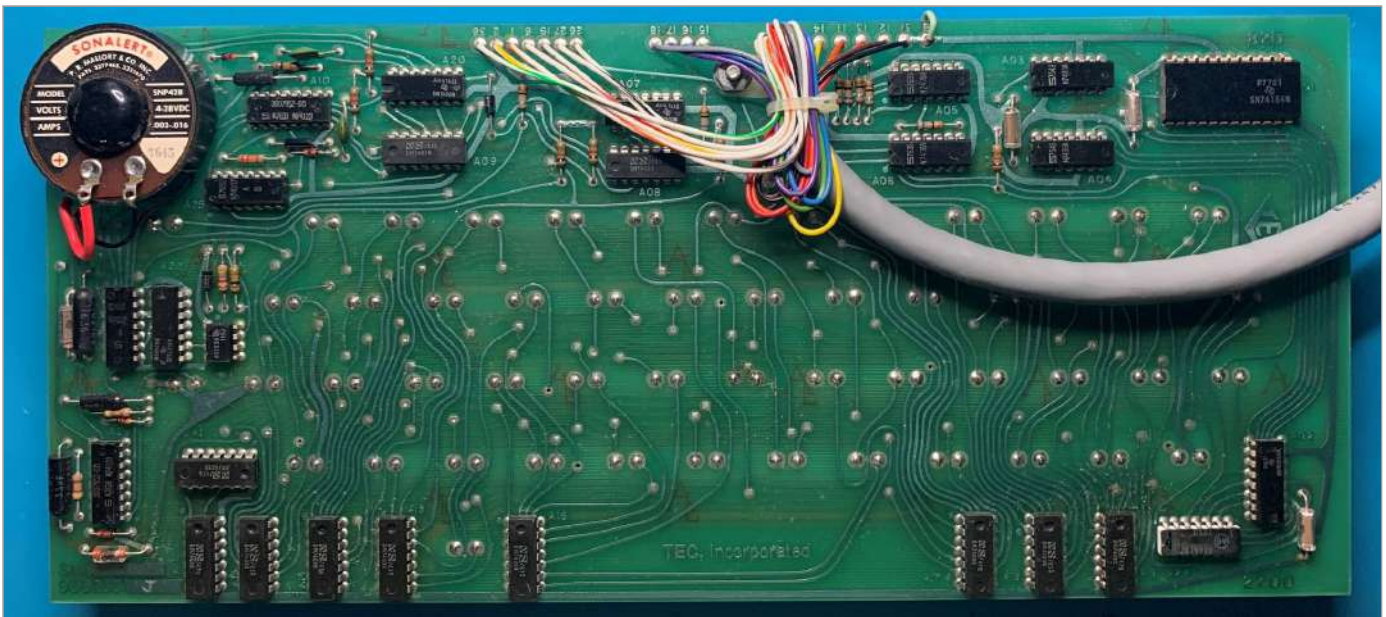
- National MM5740/AAx, with /AAC, /AAD, and /AAE layout variants (at least that I have -- there was also a MM57499 with the same matrix inputs and TTL serial output).
- GI AY-5-2376 and SMC KR2376 for 88 keys.
- GI AY-5-3600 and SMC KR3600 for 90 keys. 3600 is the number of bits in the ROM. Plus there are similar (same pinout, different ROM contents) models like an SMC 3603.
- AMI S9201 Dynamic Keyboard Encoder, whose [datasheet](#) includes the format of the 91 punchcards (90 keys plus end card) used to customize the ROM.
- Tesla MH103, inside the Consul, a clone of some AMI chip, I believe.

OEM keyboard specialist companies like Micro Switch, that work with customers to design the protocol, would tend to use a general purpose microcontroller, such as an Intel MCS-48 series.

The TEC EKA board does not have any programmable elements. All the encoding is done with 74-series TTL and a 555 timer to control the scan cycle.



TEC-EKA-9542-top.jpg (528.42 KiB) Viewed 5651 times



TEC-EKA-9542-bottom.jpg (534.97 KiB) Viewed 5651 times

The Beehive B100 is similar, but with 256x4-bit of SRAM instead of a bunch of gates.

The Sanders Assoc 722 keyboard that was reverse engineered in the *Byte* article referred to above uses diodes to do the ASCII encoding, together with DTL logic for the strobe control. (And shift masking, I believe. I don't think diodes alone are sufficient to do that. The [Cardco Cardkey](#) external numpad only had diodes, which gave it a unique code for each key and a single any-key-pressed trigger.)

LEDs

Most of these keyboards do not have any lights. For one thing, that would be even more signals to pass, if sticking to a parallel protocol.

A few have a power LED or some kind of Carrier Detect / Data Set Ready indicator. These were terminals, or part of one. But today the PC is the terminal. So the ASCII keyboard converter is a modem. So I actually have

these display Data Terminal Ready (i.e., does a program have `/dev/ttyACM0` open) instead.

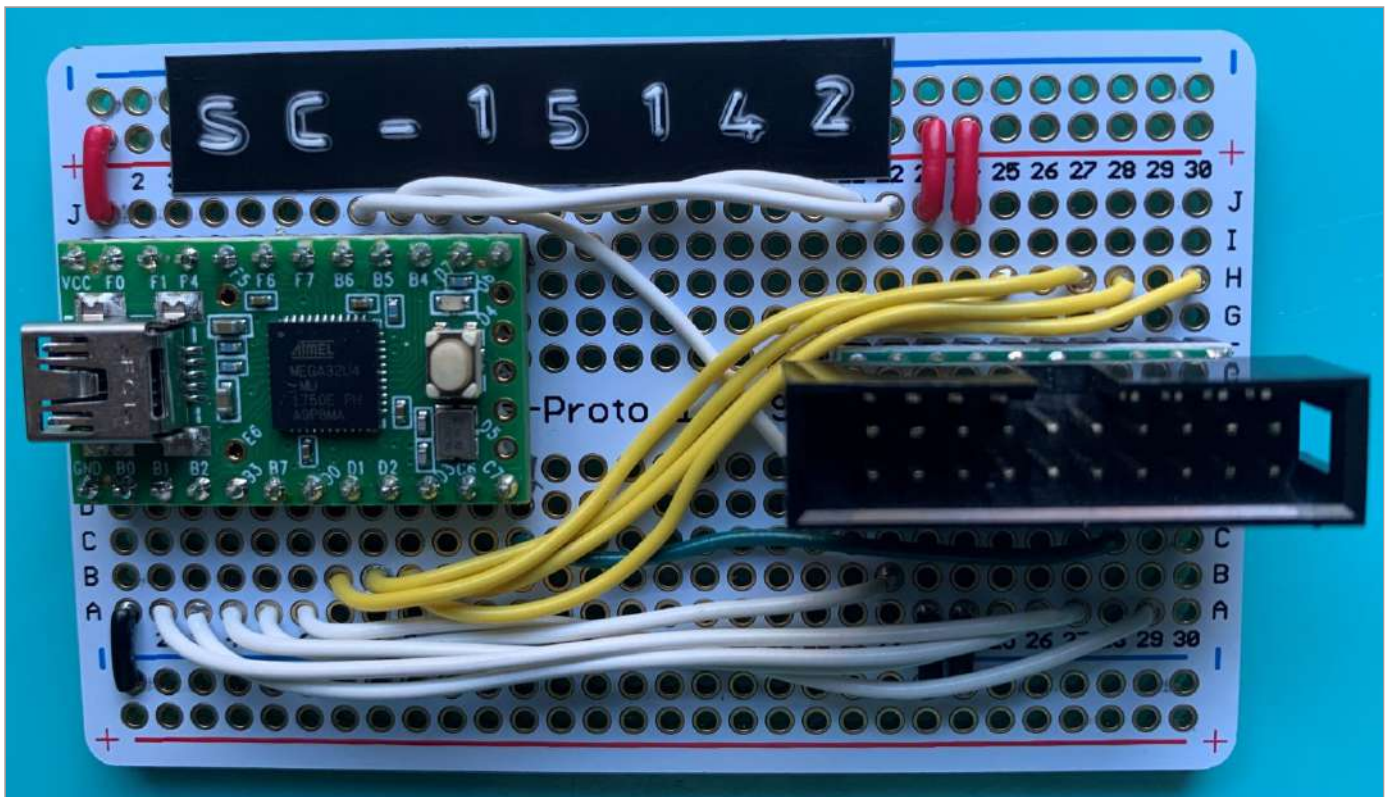
Many have an LED inside / under the **CAPS LOCK** key, but entirely under the control of the keyboard itself. This Micro Switch SC-15142 /63ST13-1 has a **SHIFT LOCK** key, that is one that shifts non-letters as well, which is just wired along with the **SHIFT**, as a result of which the LED lights when **SHIFT** is pressed, too. Which turns out to be very distracting (to me).



63ST13-1-top.jpg (436.5 KiB) Viewed 5651 times



63ST13-1-bottom.jpg (387.04 KiB) Viewed 5651 times



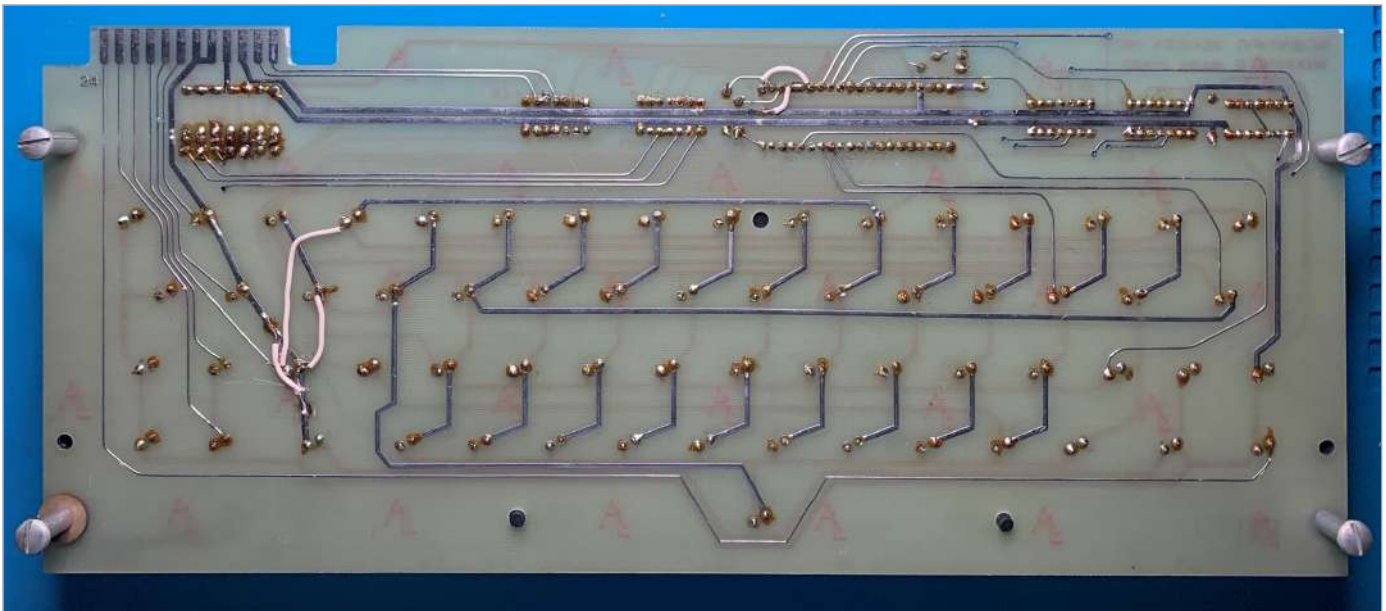
SC-15142-converter.jpg (436.74 KiB) Viewed 5651 times

The Console 262.4 is, in fact, the same PCB as the 262.3, but populating the UART and LEDs slots. The strobed parallel output from the encoder fills the UART's parallel-to-serial input. And the LEDs are lit directly from its serial-to-parallel output.

This Scientific Devices board has seven little LEDs right on the parallel output, so that they show the (LSB left, inverted) ASCII character while the key is held down.



Scientific-Devices-top.jpg (1.2 MiB) Viewed 5651 times



Scientific-Devices-bottom.jpg (380.43 KiB) Viewed 5651 times



Scientific-Devices-converter.jpg (382.17 KiB) Viewed 5651 times

Bell / Buzzer

The TEC EKA keyboard has a signal whose falling edge triggers a one-shot to drive an alarm-style buzzer.

I can see no reason that a parallel ASCII boards could not add a pin connected directly to a speaker / piezo, though I don't seem to have one like that. It would be easy enough to PWM from the Teensy, as is done for

some converters for other protocols.

Additional Outputs

Some keyboards have 8 strobed outputs instead of just 7 and encode some keys with the high bit set. For instance, the Amkey SNK-58.

The Micro Switch SC-15142 has a parity bit for the ASCII character. The JE610 has 8 bits of output and a 9th parity bit.

(Back a long time ago, like when these keyboards were new, we had specially [modified DM2500](#)'s that set what the terminal thought was the parity bit directly from a **meta** key near space. This was supported by a few programs, including PDP-10 EMACS.)

Direct Keys

It is common for a few keys to be output directly, rather than scanned and ASCII encoded. Generally this is done by the switch connecting the signal to ground and the terminal having a pull-up. But the Maxi-Switch woodie actually has two signals for each of the switches it does directly.

Popular keys to do this was include the shifts, **Shift**, **Ctrl**, **Caps Lock**, which is easy, since the encoder tends to want these signals separate from the matrix to do the bit logic for shifting anyway.

Some others include **Break**, **Clear**, **Here Is**, and **Reset**, all of which would be special functions handled by the terminal, rather than just passed on to the host over RS-232. The **Break** and **Reset** keys sometimes have stronger springs in their switches.

The two direct keys on the JE610 do not have legends and are just called **UD1** and **UD2** in the documentation.

The Scientific Devices keyboard has 10 direct outputs, not all of which have legends, at least on the board I have worked with.

Break

The **Break** key on a terminal would be for sending a break signal over the serial line.

- This might send a unique code and the terminal does the serial work.
- The **Break** key can just be another direct key and the terminal asserts break while it is pressed.
- A timer inside the keyboard can turn a **Break** press into a pulse of a few hundred milliseconds.
- The **BREAK** key on the JE610 just sends a null character, since it isn't really a terminal.

Repeat

Many of these keyboards have some kind of repeat function, either a **Repeat** key or auto-repeat (typematic).

- This can be done entirely inside the keyboard, with the strobe pulse repeating periodically so long as a key is pressed, and enabled by the **Repeat** key if there is one.

- The keyboard can have an any-key-down output signal and the terminal can repeat the last-seen character so long as that is on.
- The **Repeat** key can then just be another direct key that the terminal uses it to enable this.
- The keyboard can have an input signal that acts as a repeat clock, triggering another strobe on its edge if a key is still pressed.
- Or the terminal can send a delayed "ack" of a key press, triggering the same action.

Last edited by [MMcM](#) on 21 Feb 2021, 06:12, edited 3 times in total.



MMcM

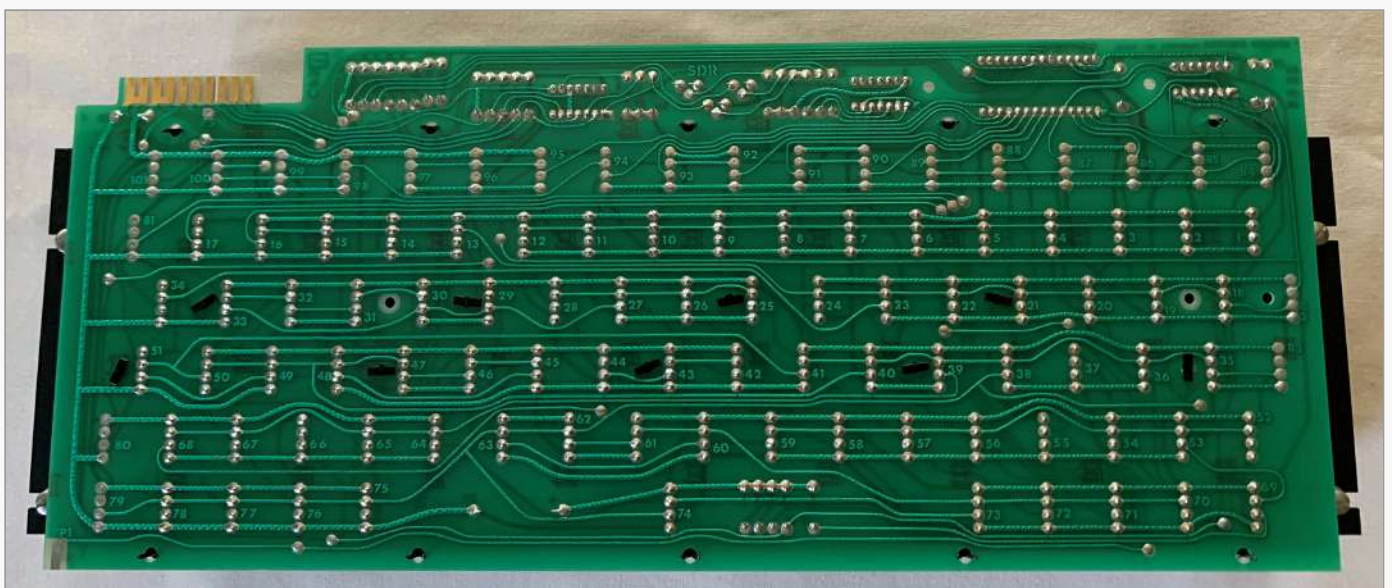


21 Feb 2021, 05:48

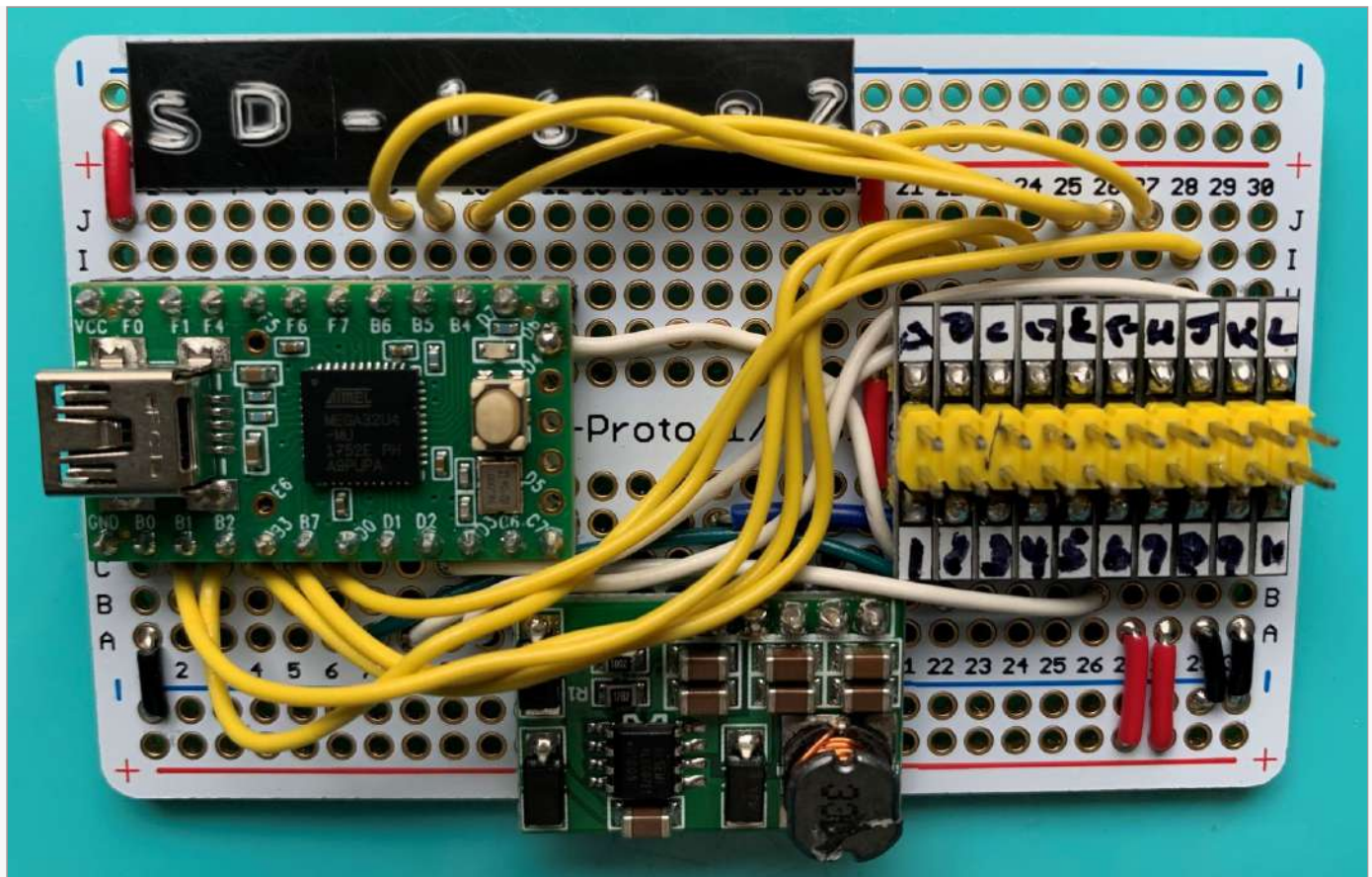
This Micro Switch SD-16192 / 78SD12-6 board has an AML encoder.



SD-16192-top.jpg (1.18 MiB) Viewed 5641 times



SD-16192-bottom.jpg (1.41 MiB) Viewed 5641 times



SD-16192-converter.jpg (469.01 KiB) Viewed 5641 times

Most switches are 4B3B, 2.5 oz. sink pulse. The key in the lower-left is 4B8B, 8.0 oz., but still a pulse, which is close to synchronized with the strobe pulse. The three keys in the upper-right are 4B3K, timed repeat, so those signals remain on while the key is still pressed. These four keys are output directly. They also send unique character codes, so they can be tracked either way.

The encoding is mostly ASCII in mostly a standard typewriter layout, but with some oddities, like line feed where you'd expect slash and ^A for space. Many shifted keys send a character with the eighth bit set.

CODE: **SELECT ALL**

	02	2F	24	31	32	33	34	35	36	37	38	39	30	29	22	7C	11
	13	14	04	71	77	65	72	74	79	75	69	6F	70	2D	20	15	10
	06	07		61	73	64	66	67	68	6A	6B	6C	3B	0D	18	19	05
03	09	12		7A	78	63	76	62	6E	6D	2C	2E	0A		1B	1F	
1D	08	1A	17				01						0B	1C	60	1E	
	23	25	21	0C	0E	0F	5B	5C	5D	7B	3D	26	3F	28	27	2B	5E
	5F	7D	7E	51	57	45	52	54	59	55	49	4F	50	40	7F	80	81
	82	83		41	53	44	46	47	48	4A	4B	4C	3A	84	85	86	87
88	89	8A		5A	58	43	56	42	4E	4D	3C	3E	2A		8B	8C	
8D	8E	8F	90				92						93	94	95	96	

But then there are the actual keycaps that it comes with.

Spoiler:

Show

Outside of the letters and numbers, the legends don't seem to line up much at all with the ASCII character sent. So it is almost as though they were just treated like scan codes by the system.



MMcM



21 Feb 2021, 05:50

I am fairly certain that the AES board KB71SD12 from [here](#) works like these others, with an AMI encoder whose only difference is the exact matrix and so the mapping ROM.

But, sadly, it seems that chip is shot somehow. The keys themselves work. But the encoder never generates a strobe. The pulse generator works as best I can tell. I have accounted for every signal on the header, so I don't think there is anything that might be misconfigured, as opposed to broken.



JPI



21 Feb 2021, 06:03

Wow, it looks like you've been busy. Glad to see your projects as always.



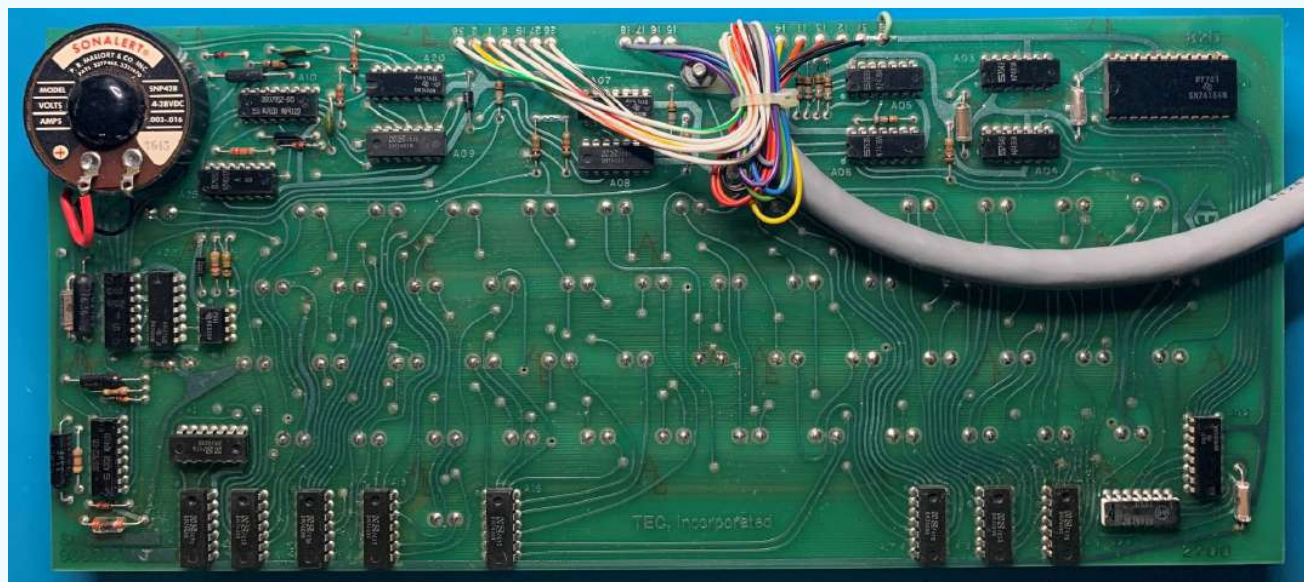
mr_a500



21 Feb 2021, 15:14

MMcM wrote: ↑





I've got the TEC EKA 9100-054 (a.k.a Harris 8620). As you can see from the date code on the SONALERT, mine is a couple years older than yours. According to the TEC pamphlet, the case has "blue Armohide paint". Notice how there are mostly white sliders with some randomly placed black sliders. (I assume yours are Clare Pendar switches, like mine.)

8620 1-5.jpg

8620 2-5.jpg

8620 3-5.jpg



eekee

“

21 Feb 2021, 23:42

“



That layout is a beautiful thing! Showing all the ASCII control codes is cool, but the separate LF and CR keys are something else! 😊 I would have loved that on the Linux console in the 90s. Not so much now though. 😊

About the same goes for [bit-paired](#) keyboards. In the 90s, I got a lot of old stuff, and basically all the keyboards which weren't IBM-compatible were bit-paired. This was in the UK. But I got used to IBM's weirdness despite partially modifying my layout to bit-pairing for an unshifted colon when I was writing Python.

“ **MMcM** wrote: ↑

Trying to fill the Covid lockdown with interesting vintage boards to find, restore, and convert means a steady stream of ones that use a simple parallel ASCII protocol. The idea is this: when a key is pressed, it is converted to US-ASCII by the keyboard and output as a 7-bit (sometimes wider) signal together with a strobe pulse.

This sounds like almost what I want for the microcontroller-based computers I'm planning. I'd want them a bit simpler: partial ASCII encoding with all the modifiers as bucky bits: like direct keys but sent to the computer. I also want 2 strobe lines, key down and key up. Simple to use and powerful.

MMcM

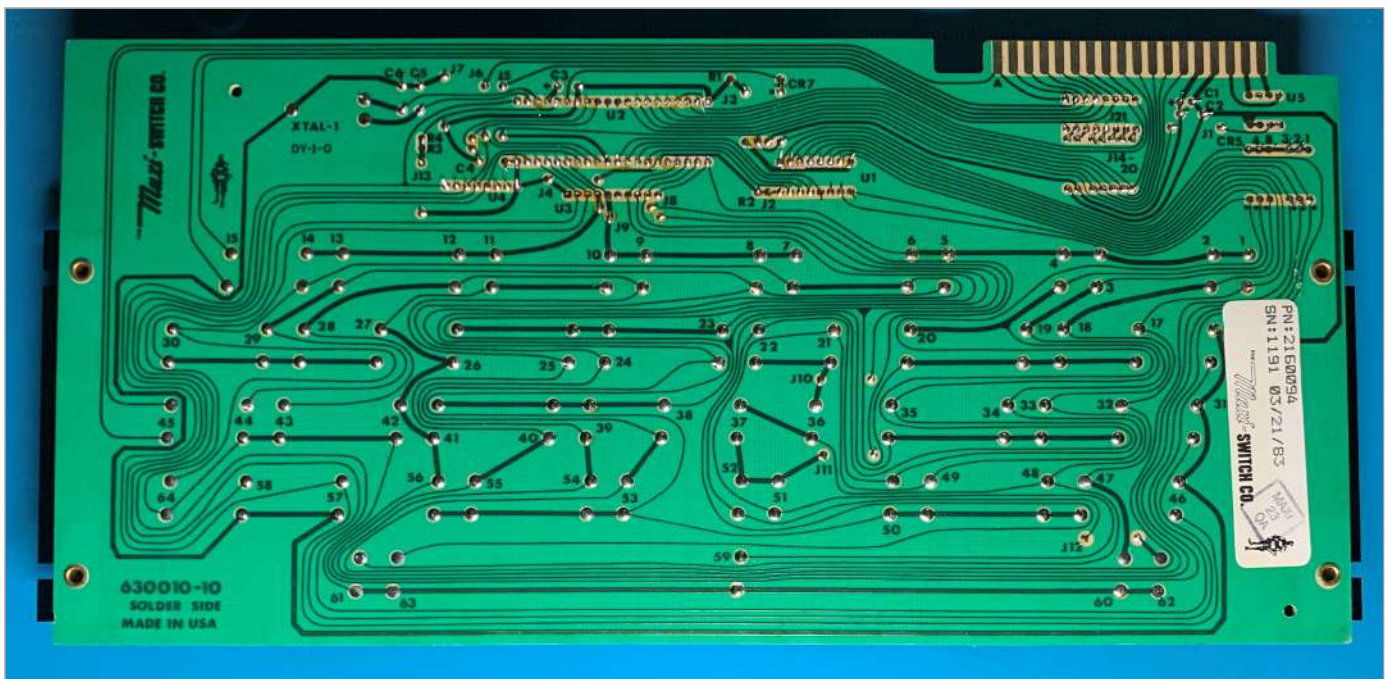


25 Oct 2021, 08:09

Another Maxi-Switch linear. This one is 2160094, with PCB 630010-10.



Maxi-Switch-2160094-top.jpg (1.23 MiB) Viewed 4647 times



Maxi-Switch-2160094-bottom.jpg (1.54 MiB) Viewed 4647 times

A general purpose microcontroller is used in place of just a converter, but there isn't anything interesting going on with the output encoding.

It has both an edge connector and a DIP-16, which suggests multiple intended OEM channels. The signals include two direct-output switches. But those locations aren't populated on this board. I am guessing they would be BREAK and HERE IS for a terminal.

ENTER ID, ENTER DATE, and ENTER TEXT seem special purpose to me, but I don't have a specific idea of where. They send 01, 02, and 03, respectively, and not brackets, so pretty definitely go with the firmware.



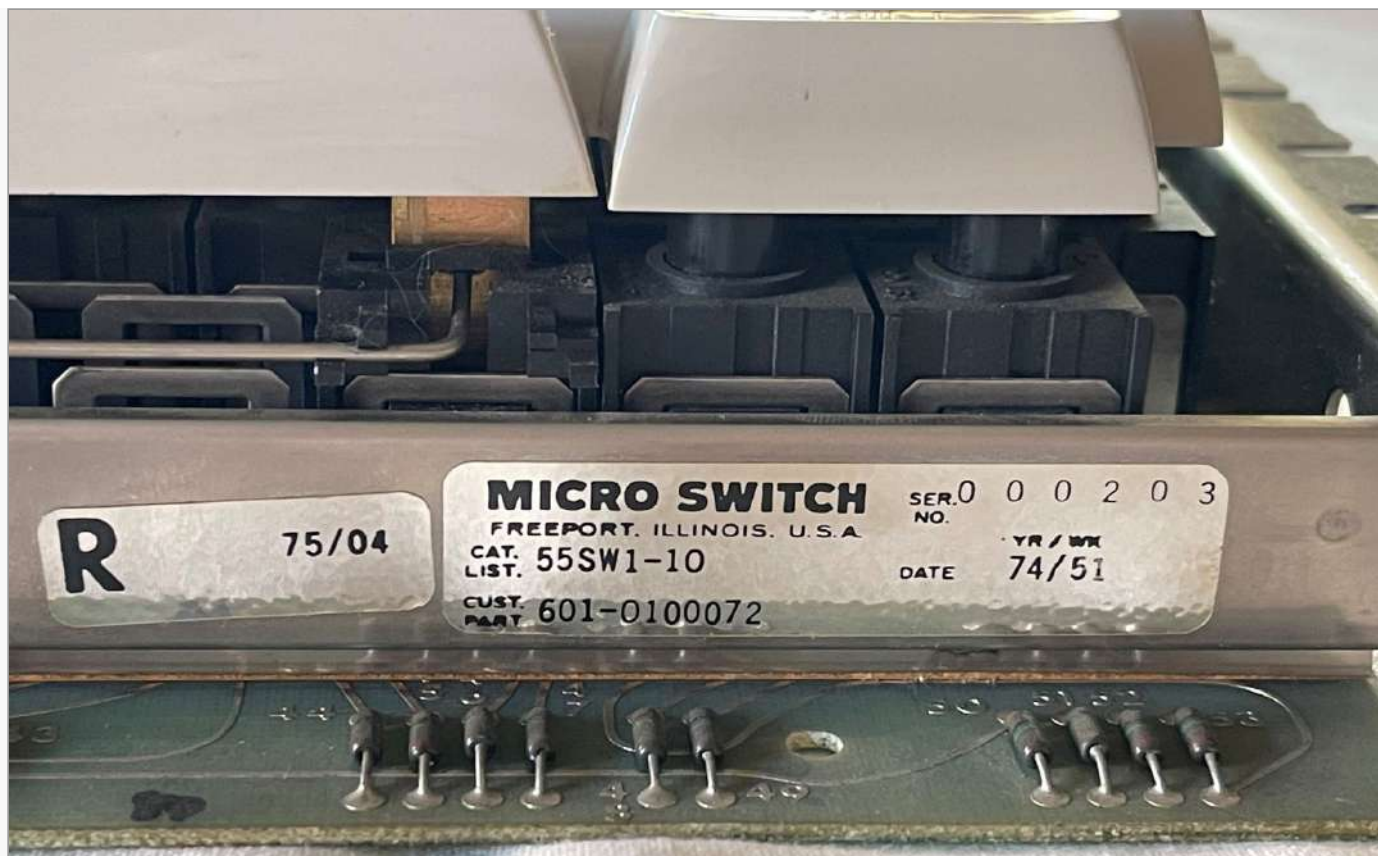
22 Nov 2021, 21:18

Another 55SW board with the same thick doubleshot keycaps with ASCII control legends.



55SW1-10-top.jpg (1.65 MiB) Viewed 4392 times

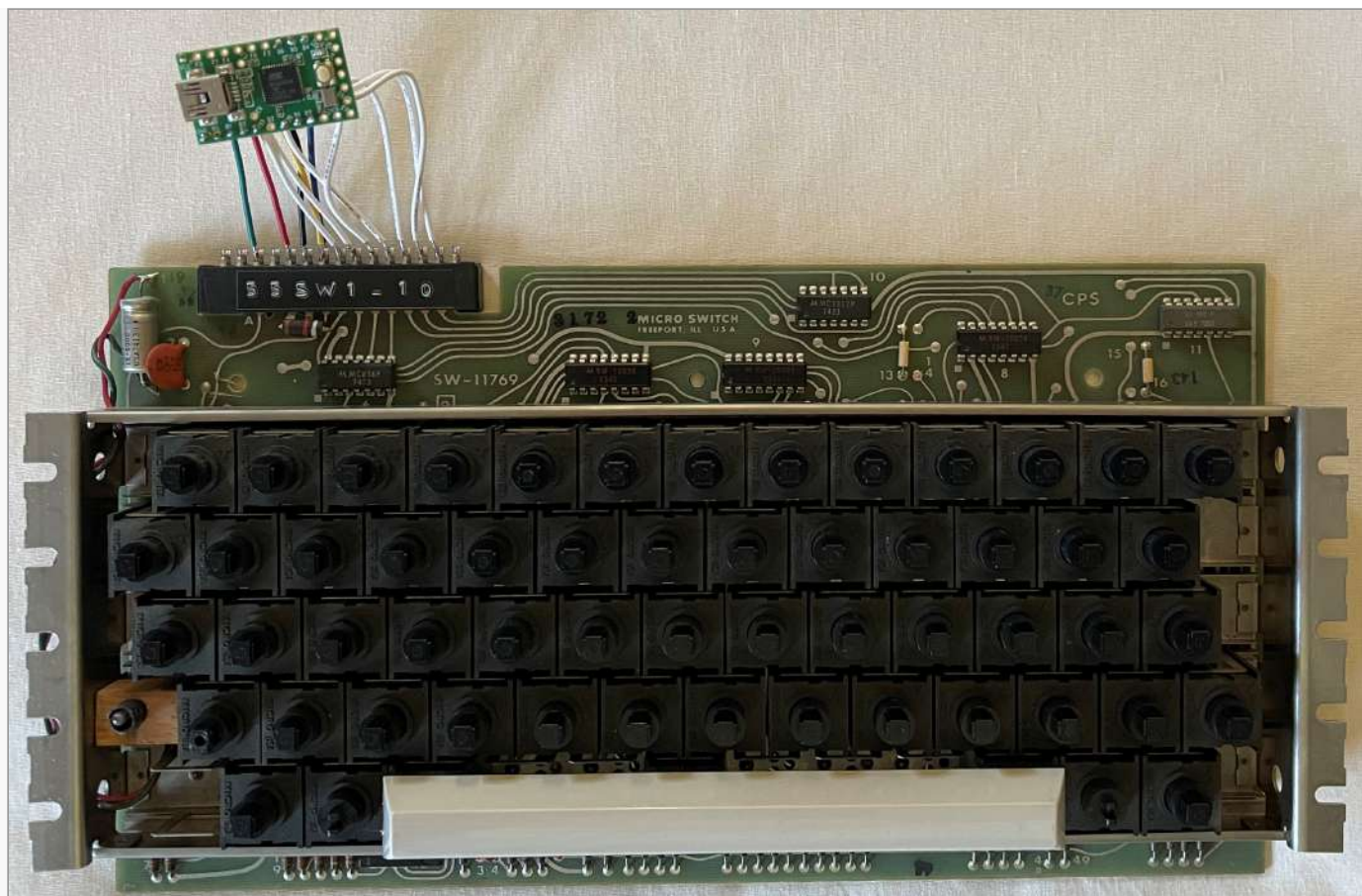
This is SW-11769 / 55SW1-10.



55SW1-10-side.jpg (381.89 KiB) Viewed 4388 times

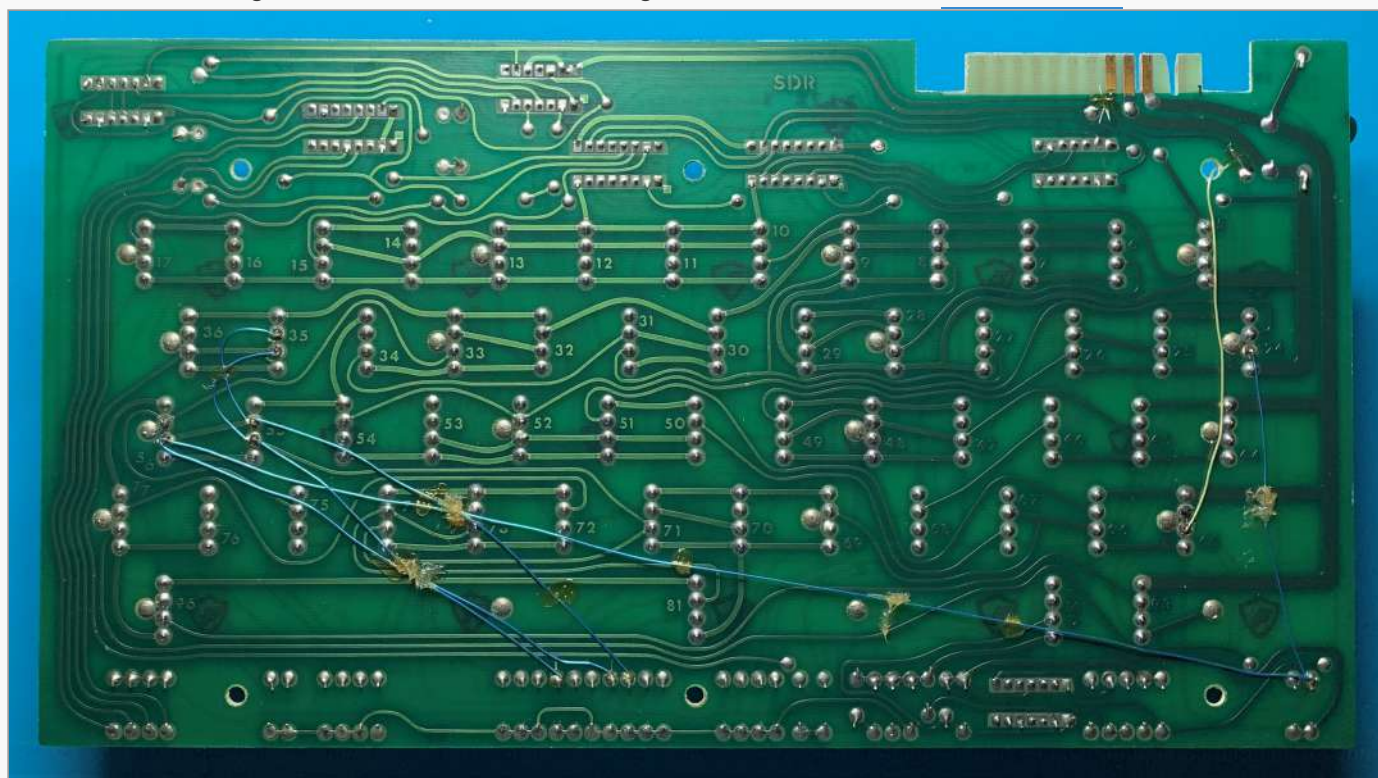
The parallel output on the edge connector is substantially the same as SW-11234 / 55SW5-2 above, though the STROBE signal is not inverted (so rising trigger rather than falling). The SHIFT and CTRL direct outputs are also not inverted as well as being on different pins.

It only needs 5V; there is no MOS encoder.



55SW1-10-switches.jpg (1.29 MiB) Viewed 4392 times

Instead it uses RW-10038 and RW-10039 2-of-N encoders, which were originally developed for the [RW series](#). The shift bit handling is done with XOR and NAND gates. It is similar to the [MiniBee circuit](#) mentioned there.



55SW1-10-bottom.jpg (1.4 MiB) Viewed 4392 times

Equally interesting are the hand-wired modifications that were done, I believe, to adapt it to some homebrew computer.

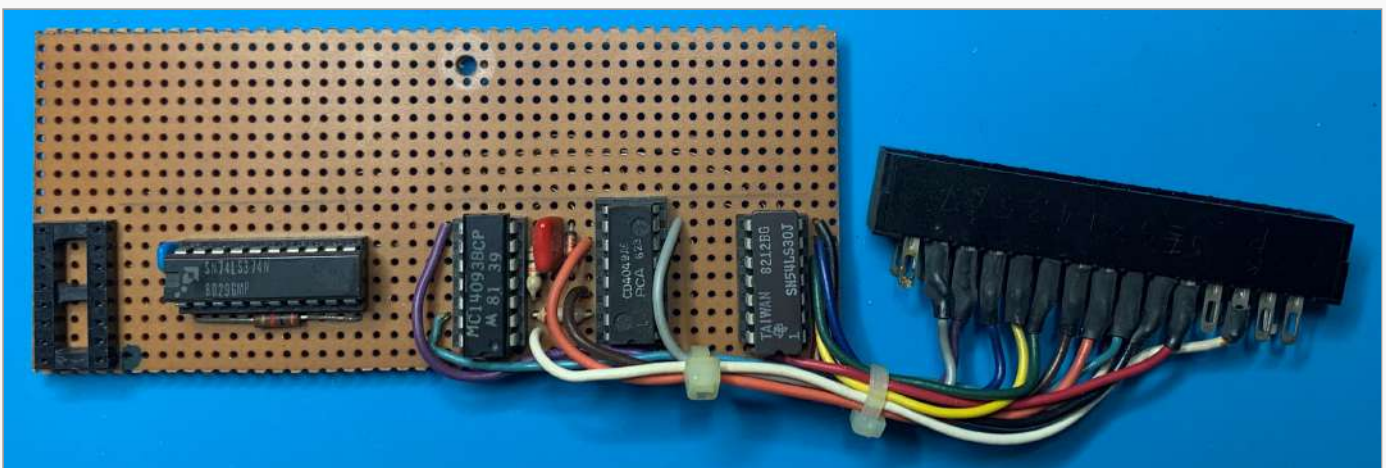
- { [was patched to always have CTRL, that is to always generate ESC, at the expense of then only being able to enter } and].
- SHIFT and SHIFT LOCK were decoupled, with SHIFT affecting the encoding and SHIFT LOCK output as a direct signal.
- SHIFT LOCK was relegated CAPS LOCK using White-out and a pen.
- a power light bulb was added, stuck into a piece of balsa wood.



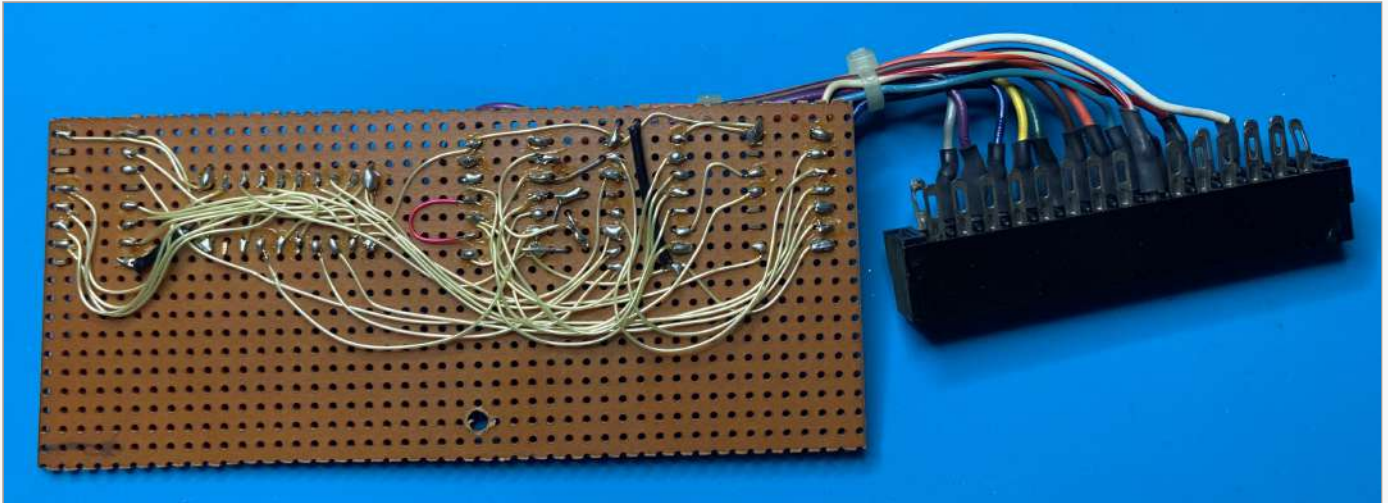
55SW1-10-lamp-caps.jpg (82.52 KiB) Viewed 4392 times

The board came with an external perfboard that

- adapt the 30-pin edge connector to 16-pin DIP, a common choice for hobbyist systems.
- NAND gates to do the bit shifting; one is 54-series (mil-spec), confirming a one-off build.
- a tri-state register to store the last character: inputs are connected to the parallel output from the keyboard, clock to its strobe, outputs would be connected to a common bus, output-control to an address decoder.



55SW1-10-adapter-top.jpg (818.14 KiB) Viewed 4392 times



55SW1-10-adapter-bottom.jpg (991.22 KiB) Viewed 4392 times

I wasn't able to keep the new legend and clean the caps, which were filthy.

I didn't undo the circuit changes, though.



Post Reply ↩



8 posts • Page 1 of 1

< [Return to "Workshop"](#)

Jump to ▼

[Board index](#)

[Contact us](#) [Delete cookies](#) All times are UTC+02:00

Powered by phpBB® Forum Software © phpBB Limited

[Privacy](#) | [Terms](#)