THE TAYLOR-WILSON PET COMPUTER UNIVERSAL PRINTER INTERFACE

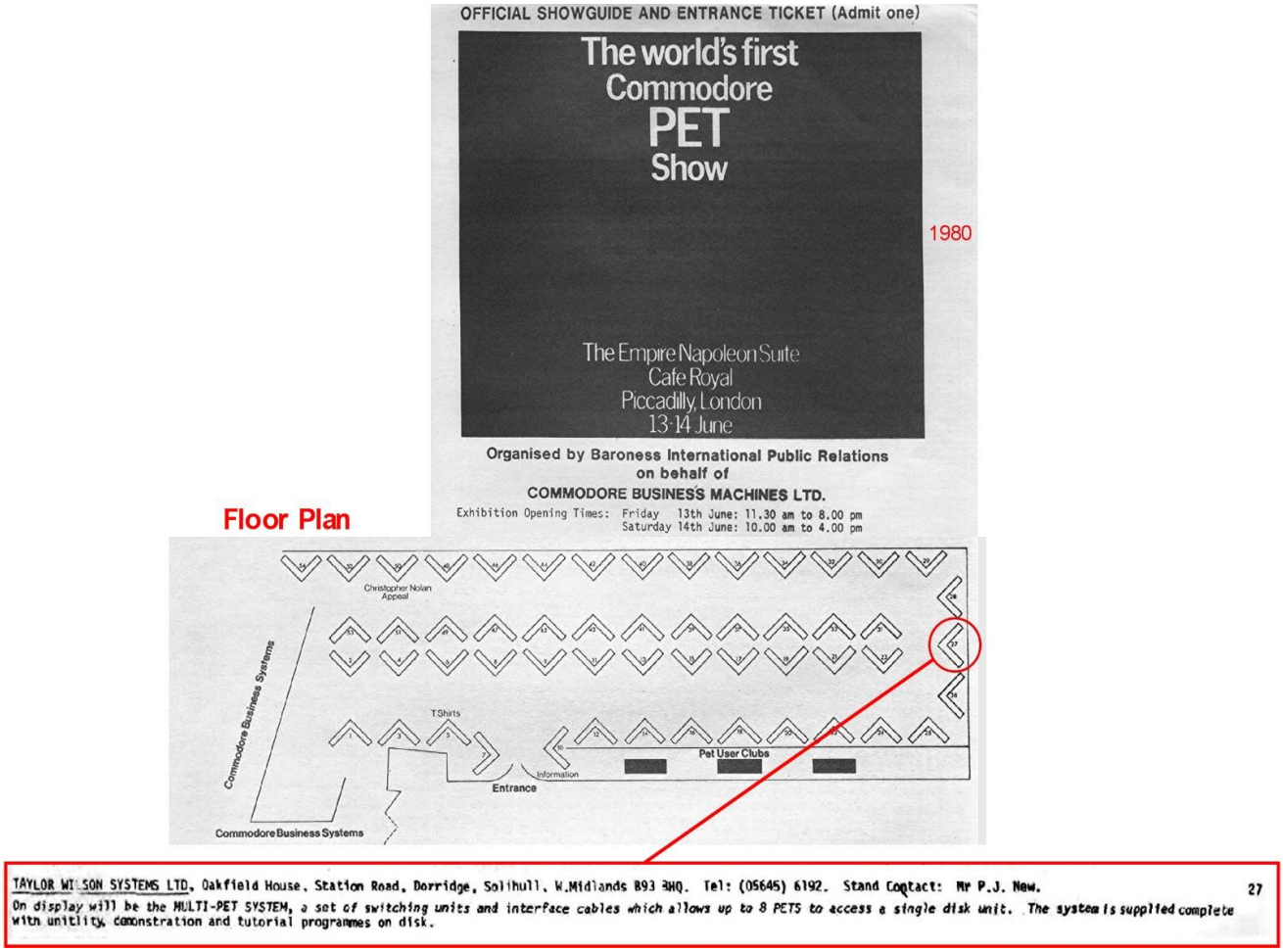
A Reverse Engineering & Restoration Project.

Dr. H. Holden. Oct. 2022

Sections in this article:

1) Background: Who were Taylor-Wilson Systems?

I was searching for a suitable printer interface for my PET vintage computer. This would be a device which accepts the output from the PET's GPIB (general purpose interface bus) and converts it to a Serial or Parallel output, to drive a Printer. I came across a very interesting vintage unit; The Taylor-Wilson Systems Universal Printer Interface. I attempted to find data on this company & the unit, but very little on the internet. However, I did manage to find a ticket to a "Big Meeting" or "PET Show", where Taylor-Wilson was one of the exhibitors in stall 27, of a 1980 meeting:



Clearly, by what they were offering, T-W had become experts at interfacing to the PET's GPIB. They had interfaced eight PET's to a single Disk unit. No mention of the Universal Printer Interface here though. I took a risk and bought the T-W unit from a UK based eBay seller, with no data on it, except for a label on its base indicating how to set the multiple DIP switches.

This interface was UK manufactured in the late 1979 or 1980 era it would appear. One of the voltage regulators has a ’79 date code. All the other IC’s have had the original labels covered or cleaned off and have been re-labelled with the RS logo, from RS Components of the UK, but without date codes.

I knew without the schematic, if I wanted to repair it, I would have to reverse engineer the entire unit & figure how it worked, so as to effectively create a service manual. This is now done and presented in this document. This information will help others repair the units too, if they have a faulty one. I expect though, that the T-W unit is fairly rare and my one may now be one of a few surviving units left from a bygone era.

I noticed right away that the T-W unit is much more elaborate than the Connecticut Microcomputer (C-M) printer adapters; the ADA1200, the ADA1450 and the ADA1800 units also designed for use with the PET.

The T-W unit contains 28 logic IC’s. This initially had me wondering, what are all those extra logic IC’s for, compared to the C-M units ?

One reason there are more IC’s is that the T-W unit has a precision Baud rate generator based on a quartz crystal and digital divider chain, unlike the ADA units which simply use an R-C based oscillator and a 555 timer IC. Also it has an output data storage latch.

In addition, the interesting thing about the T-W unit is that it was clearly designed for use with the PET computer. It contains a clever, PETSCII to ASCII code converter. This and some other features account for some of the extra IC’s, compared to the C-M printer adapters, or just generic GPIB to serial & parallel adapters, not specifically intended for use with the PET.

The PETSCII to ASCII conversion function in the T-W unit is controlled by software and/or hardware (DIP switch) and is software activated (or deactivated) by applying a secondary address in the OPEN command statement after the primary IEEE-488 address.

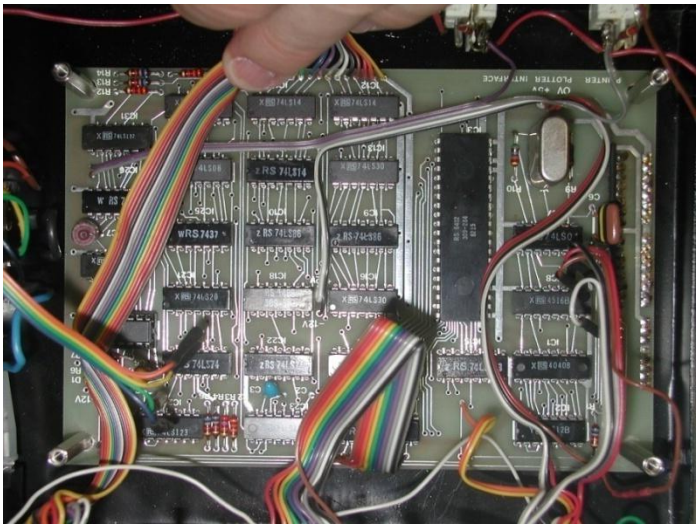
Also, except for a couple of small (and perhaps unusual) aspects of the physical design of the T-W unit, which will be discussed in this document, the T-W unit appeared very well theoretically designed, so I considered that it was worthy repairing, restoring & and creating a manual for it.

This reverse engineering effort presented here includes the PCB foil design and the full schematic derived from that. Also a circuit analysis to explain how the unit works.

The circuit analysis only became possible after the schematic was documented. This provides other T-W unit owners (if there are any) with the required data for repairs and PET owners with enough data on the unit to build a replica, if it was required. In this instance I did not require to manufacture a replica, because I have the original unit that I have restored to working condition.

2) INITIAL CHALLENGES:

One thing initially, which complicated the study of the PCB, was that it was hard wired into the housing. The ribbon cables joining the PCB to the rear panel connectors were soldered directly into the PCB. It was not possible to unplug anything to remove the pcb from the housing.



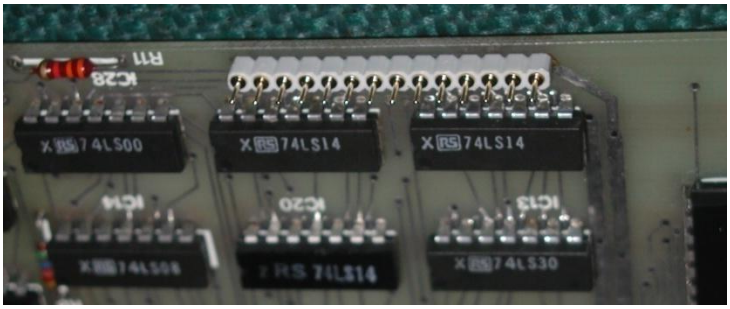
Also, the same applied to the power supply PCB, which sits above the main PCB. It was hard wired in making access nearly impossible. This was modified by adding 0.9mm gold plug-socket connectors for individual wires and “connector arrays” for the ribbon cables.

Since the plated through holes in the main board were only 0.6mm diameter where the ribbon cables were connected, they could not receive standard sized header pins for a connector. This required the use of some 0.55mm Gold header pins and a “custom connector” manufactured from some IC sockets and some PCB material.

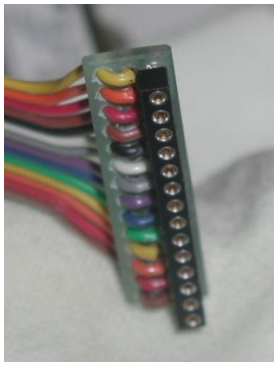
Once the unit was in a condition where the boards could be unplugged, the study could begin.

On the other end of the ribbon cables, these are soldered to the rear panel Serial connector, the rear panel GPIB connector and the rear panel Printer connector. Therefore, there was no help in unscrewing those to free the PCB.

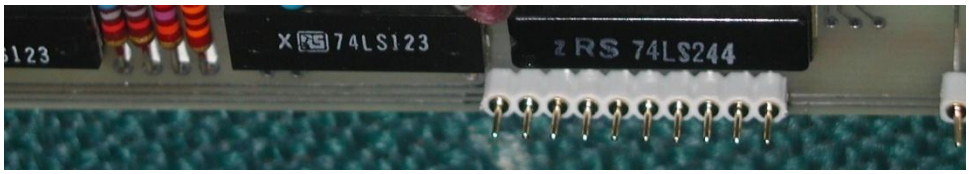
The images below show how the plugs/pins were added so that the PCB could be taken in & out of the unit for study:

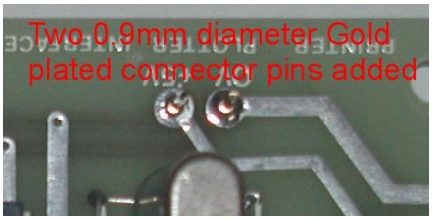


0.55mm diameter pin array added.

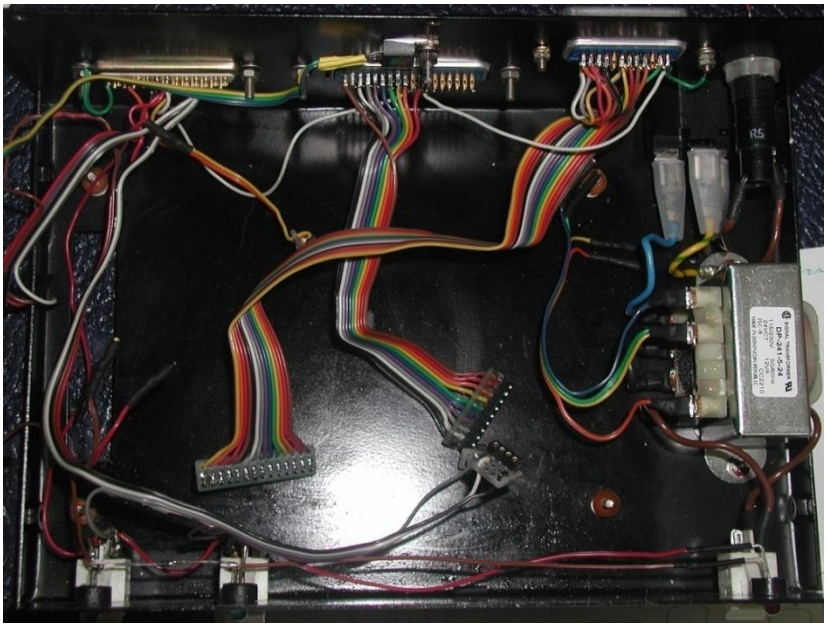


Connectors made from IC Sockets





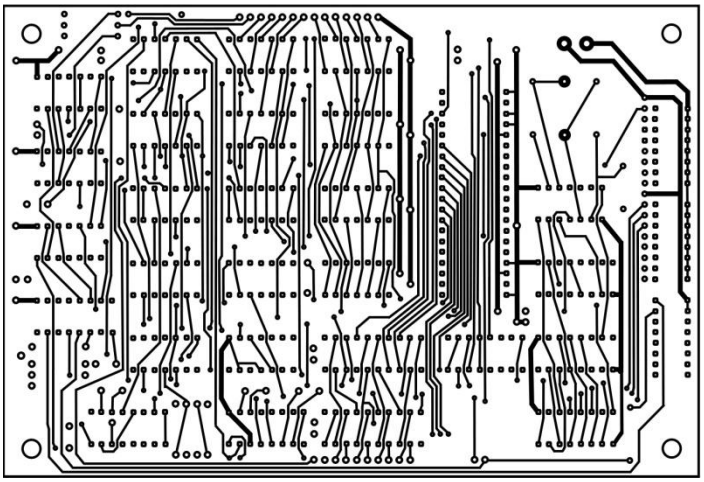
The image below shows all of the plug fittings added and the PCB's removed. Of note, the power transformer in this unit has been replaced by me. The original was, in fact, inadequate, but not faulty. This is explained in the section on the power supply nearer the end of this article. Later I plan to re-solder and sleeve the connections of the wires to the rear panel connectors, which were not sleeved by the manufacturer and appear messy. Also note the exposed wires on the LED's and the somewhat messy wiring (See article end).



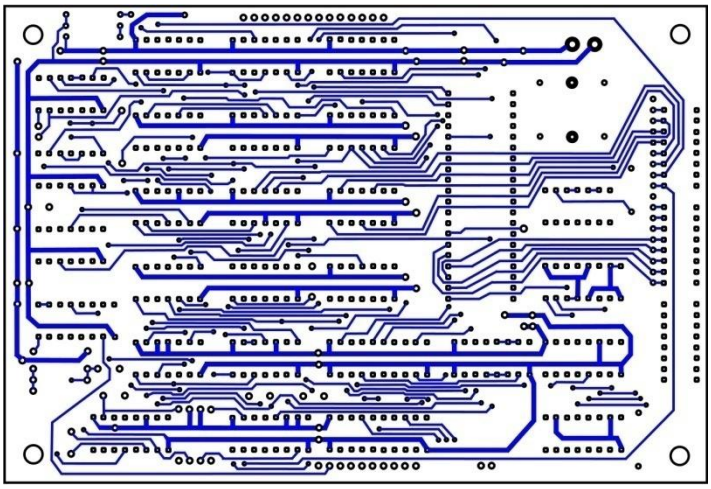
The first order of business was to reverse engineer the PCB foils:

3) MAIN PCB FOIL PATTERNS AND LAYOUT:

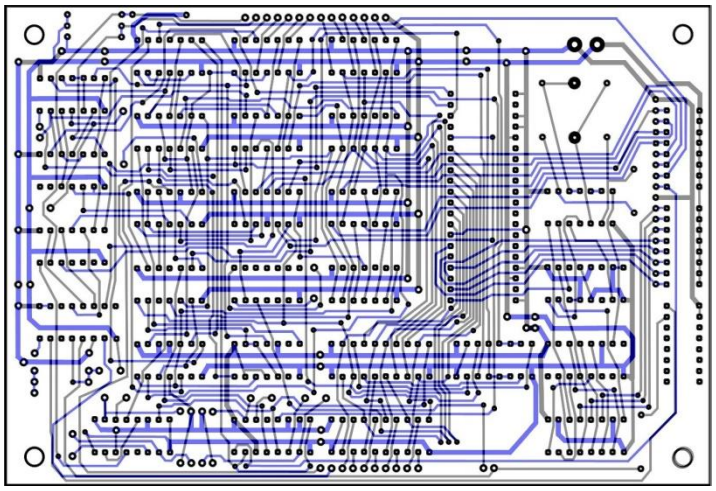
Top



Bottom

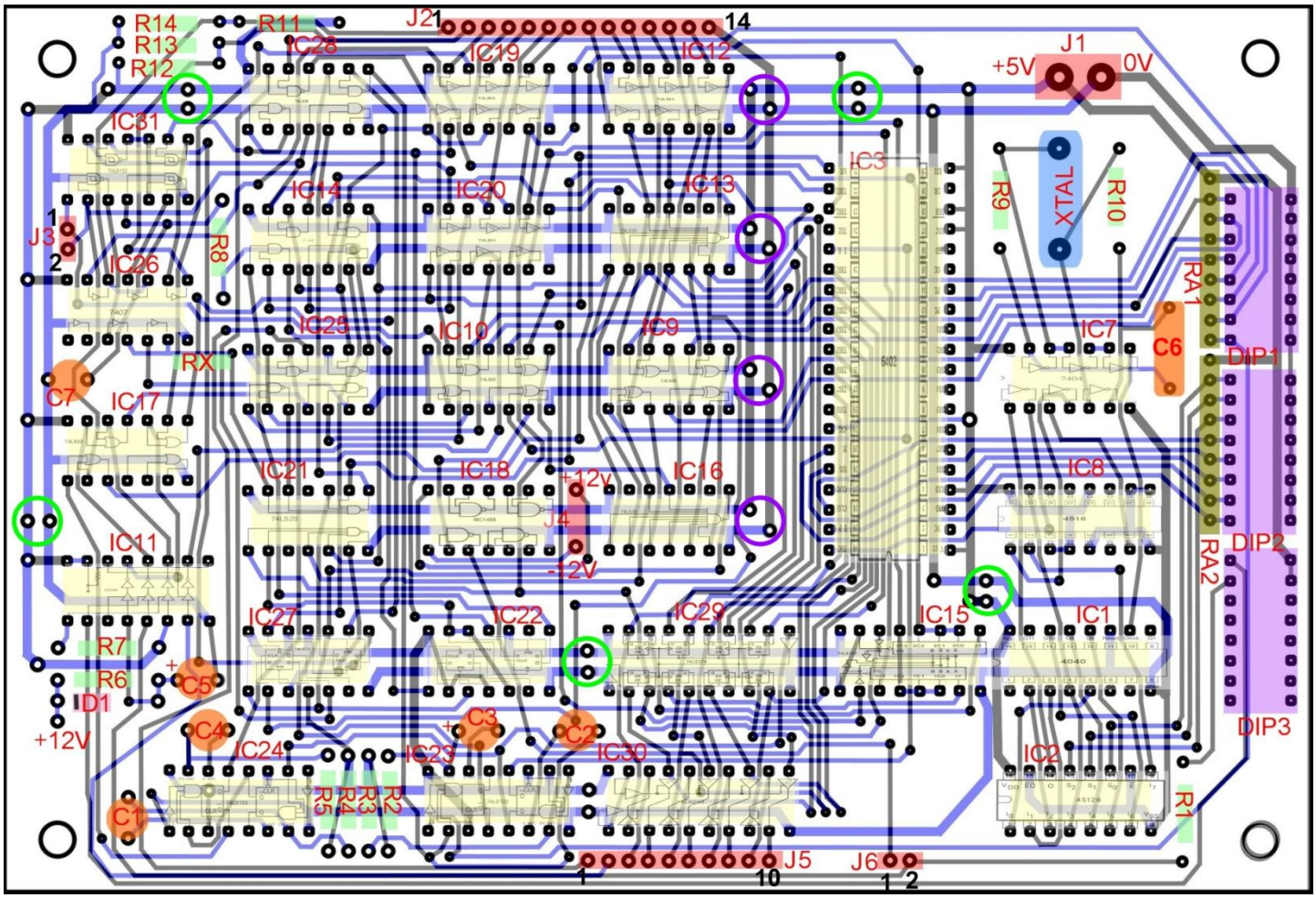


All



To enable rendering of the schematic from the PCB foil pattern, I used a modified type of component overlay image. I pasted in the IC's internal logic diagrams where possible. Then I had it printed out (by commercial printing company) to a size about 4 times the surface area of a piece of A4 paper. This made it easier to initially draw out the schematic by hand on paper and then later draw it as a tidy image on the computer.

Of course, if you have designed a circuit yourself the flow and the logic of the schematic is easy to draw well. If somebody else has designed the circuit and you don't fully understand how it works, yet, the schematic could be drawn in 101 ways. So the final result for the schematics shown later, were about my 3rd iteration, after I figured out how the circuit actually worked.



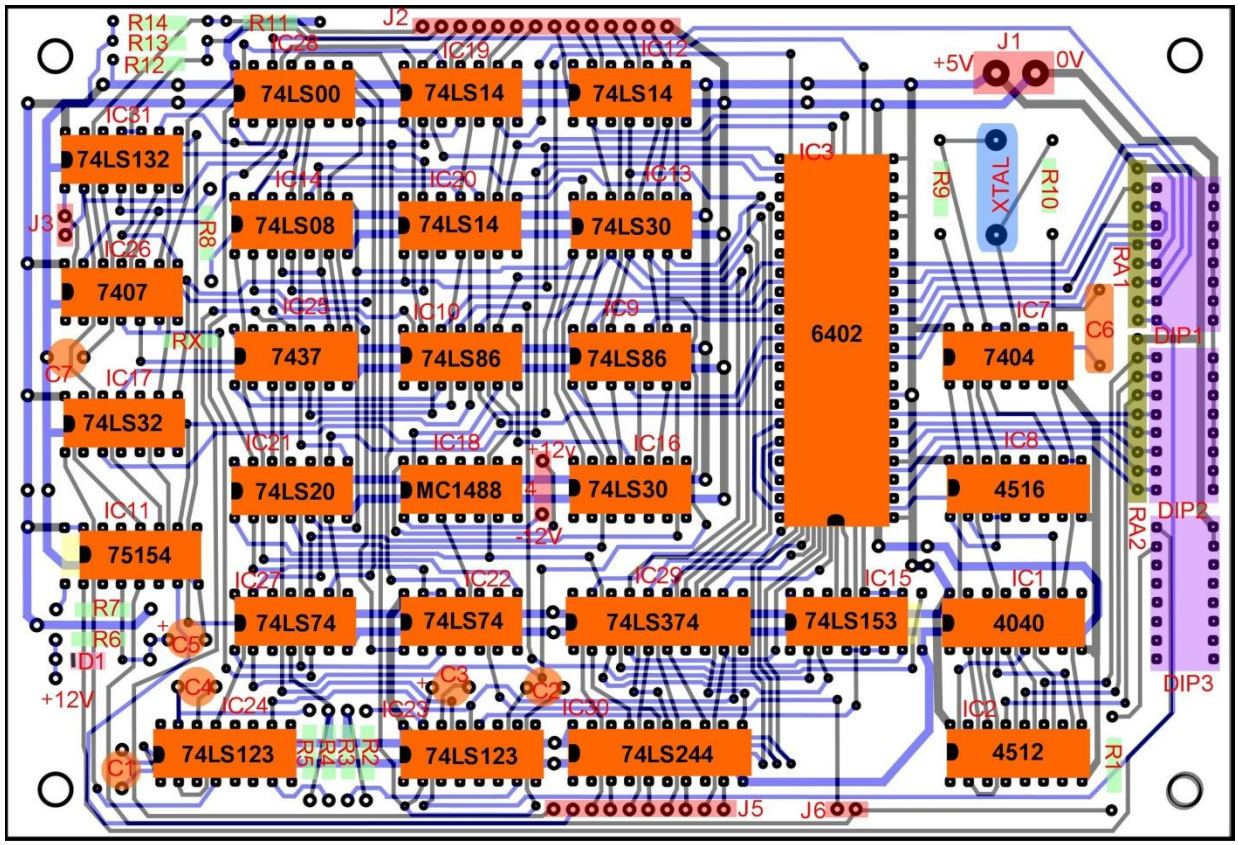
One peculiar oddity with this pcb, it does not contain a single 0.1uF bypass capacitor on its 5V IC power rails ***not a single one*** despite 28 IC's ! This is very unusual in my opinion.

Four capacitors could be added to the original board, shown where the existing PCB via holes are, within the purple rings on the diagram above. I decided (some nerve) to modify the PCB design and I added the holes & pads shown inside the green rings, which are not there on the original PCB, where 0.1uF ceramic capacitors could (probably should) be added. So that there are at least 5 bypass capacitors for the 28 logic IC's, but even that doesn't sound very generous.

I addition I had to name what were now "PCB Connectors" with numbers J1 to J6 and assign them pin numbers. The connector J1 is the 5V power input and the holes here fortunately were large enough to fit the standard 0.9mm diameter Gold pins and single socket connections (from Jaycar Electronics). The same type of pin & connector were added to the power supply PCB.

Also of note, not shown, there is a link wire on the rear of the PCB connecting the +12V connection of J4 with the +12V connection on cathode of D1 and R6.

The following diagram makes the IC type numbers easy to identify:

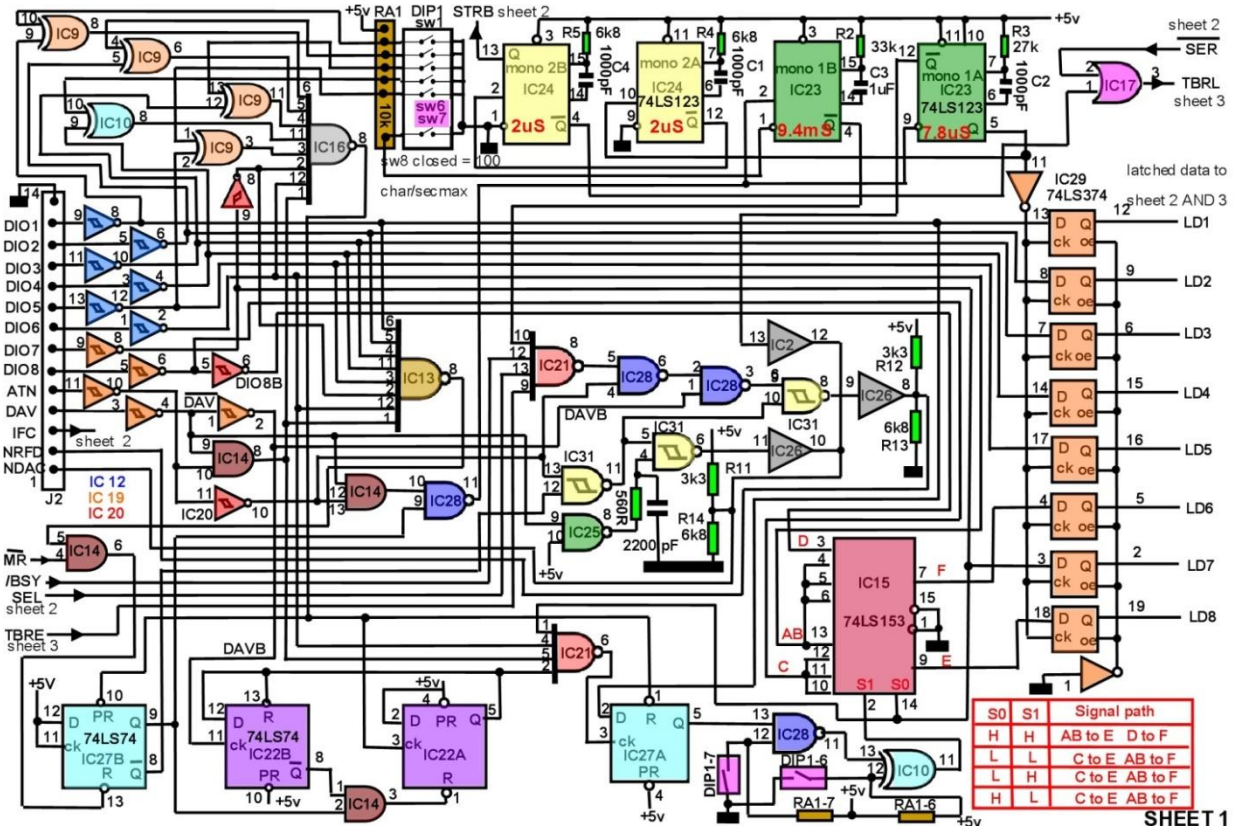


The whole schematic (excluding the power supply) is too large to easily draw on one standard sheet of paper. Therefore I broke it down to three sheets with the minimum number of interconnects.

4) SCHEMATICS:

Sheet 1:

This shows the main circuitry. The circuit operations are discussed in the section on operating theory. I added a small table here in Red, lower right on the diagram of sheet 1, to indicate the signal flow via the 74LS153, IC15. This part of the circuit is in charge of the PETSCII to ASCII conversion.



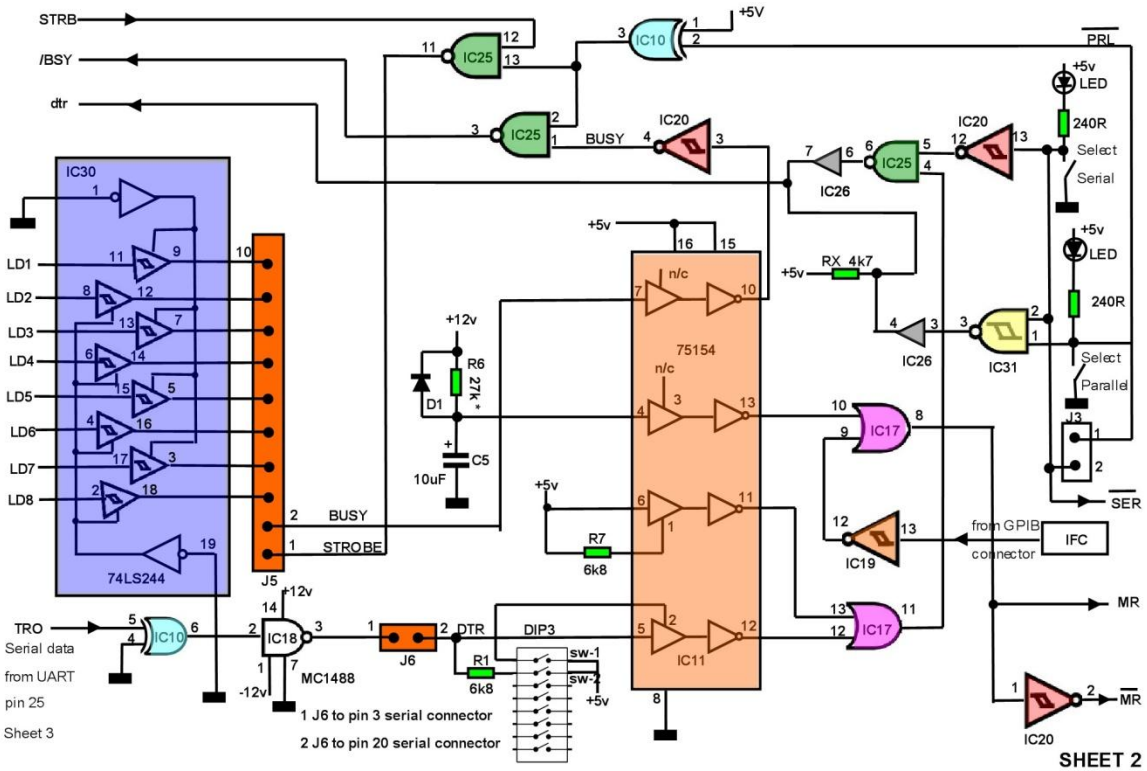
The 14pin array (now a connector now labelled J2) leads to the GPIB panel connector on the rear of the unit. A later diagram shows which pins are which in that connector. The switches sw-6 and sw-7 in the DIP switch block DIP-1 are located in the circuitry below IC15.

Essentially there is a nest of gates in the central part of the diagram which have three basic modes of operation depending on; 1) receiving addresses 2) when data bytes are transported to the output latch and 3) when the unit is in a "not ready" state. This "gate array" is supported in its operations by four Flip Flops and four Monostables or "one shots" the functions of all of these will be explained in the theory section.

Sheet 2:

This schematic contains the power up reset circuitry, the logic for the parallel and serial front panel select switches and the connection to J6. This connector connects to the RS-232 serial connector on the rear panel of the unit.

The connector J5 leads to the Printer connector on the rear panel of the unit.



Of note there is a significant issue with the value of R6 (27k), hence the * near its label, which was evident in conjunction with the issue involving the inadequate power transformer. This is discussed on the section about the power supply. When I first received the T-W unit it was not working due to an interesting combination of circumstances, explained later.

Three types of Reset:

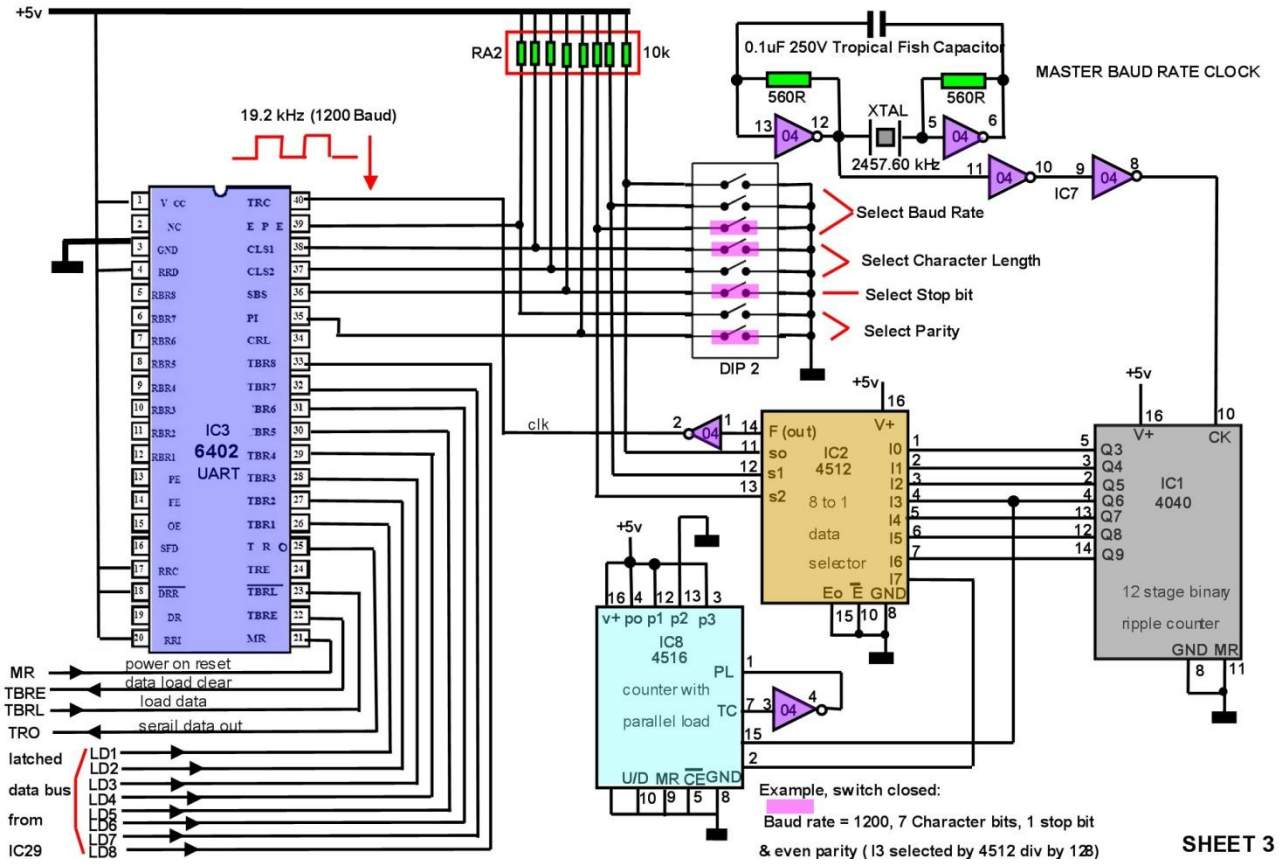
One thing worth mentioning, initially, before the circuit analysis, there are three types of Reset which put the T-W unit from a Listen to the Un-listen mode on the GPIB bus. Referring to sheet 2 above, the first one is a Power On reset due to R6 and C5. The second one is a reset issued by the IFC control line on the GPIB bus, and the third is the detection of byte 3Fh which signifies the end of the transmission of the file. The control line EOI is not used by the T-W unit. The second two reset types are discussed in the circuit analysis section.

Sheet 3:

This shows the UART and its support circuitry.

The UART's Baud Rate control circuitry has been very nicely done and is far superior to an RC based oscillator.

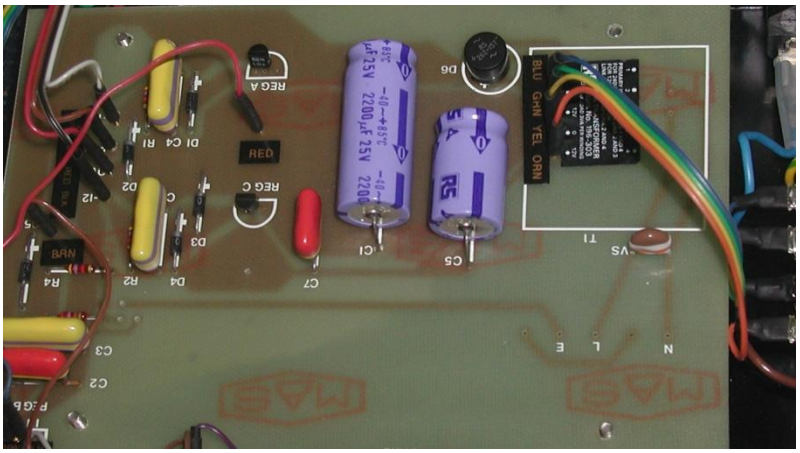
"Tropical Fish" capacitors (if anyone is unfamiliar with them) are a type of decorative looking striped film capacitor made by Mullard in the UK and have very brightly coloured stripes and resemble Tropical Fish, so they acquired this "nickname" in popular electronics culture.



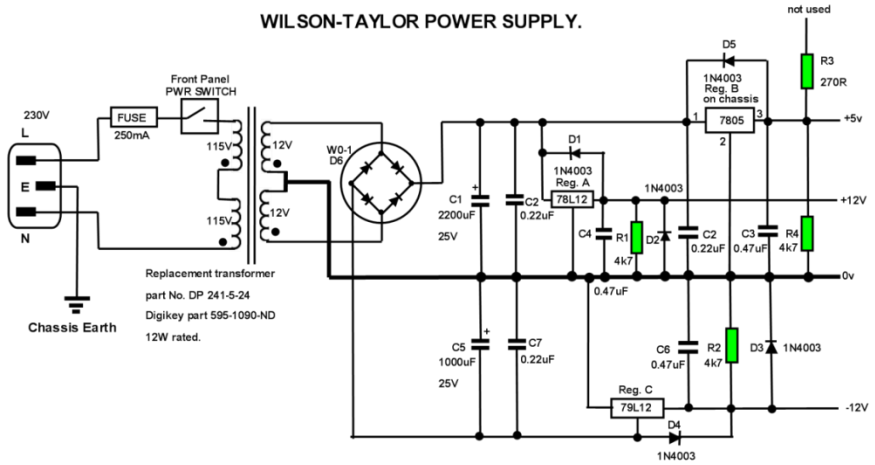
5) POWER SUPPLY:

The power supply board in the W-T unit sits on top of the main PCB, spaced away from it by 1" long spacers. Also, there is a 1/8" thick grey PVC plate separating it from the main PCB. The PVC plate is screwed to the spacers and the power supply board is attached to extra screws on the PVC plate, which are separated from the spacers. Probably one reason for this was that in some cases, instead of the line power transformer being screwed to the chassis, it was a 6W rated PCB mounted transformer, RS part number 196-303 seen on a label on the PCB.

Previously, all the wires were soldered directly into the PCB, before I fitted the single pin 0.9mm Gold connectors to allow the wires to be unplugged and free the PCB.

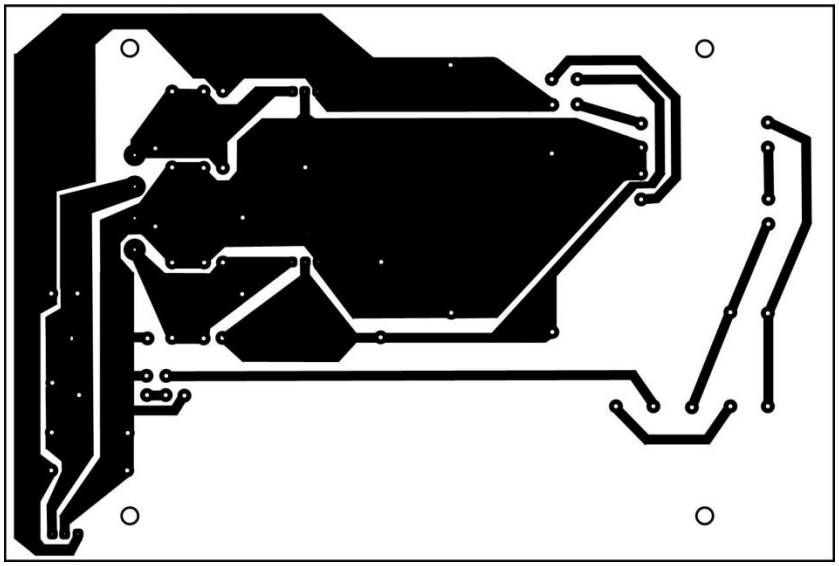


Of note, the electrolytic capacitors have RS logos too (like the IC's) and there are high voltage rated Tropical Fish capacitors, much higher than required. Perhaps there was a surplus of these capacitors at the time. The circuitry is very straightforward to generate the +5V and the +12V and -12V supplies. The +/- 12V supplies are required for the serial interface only.

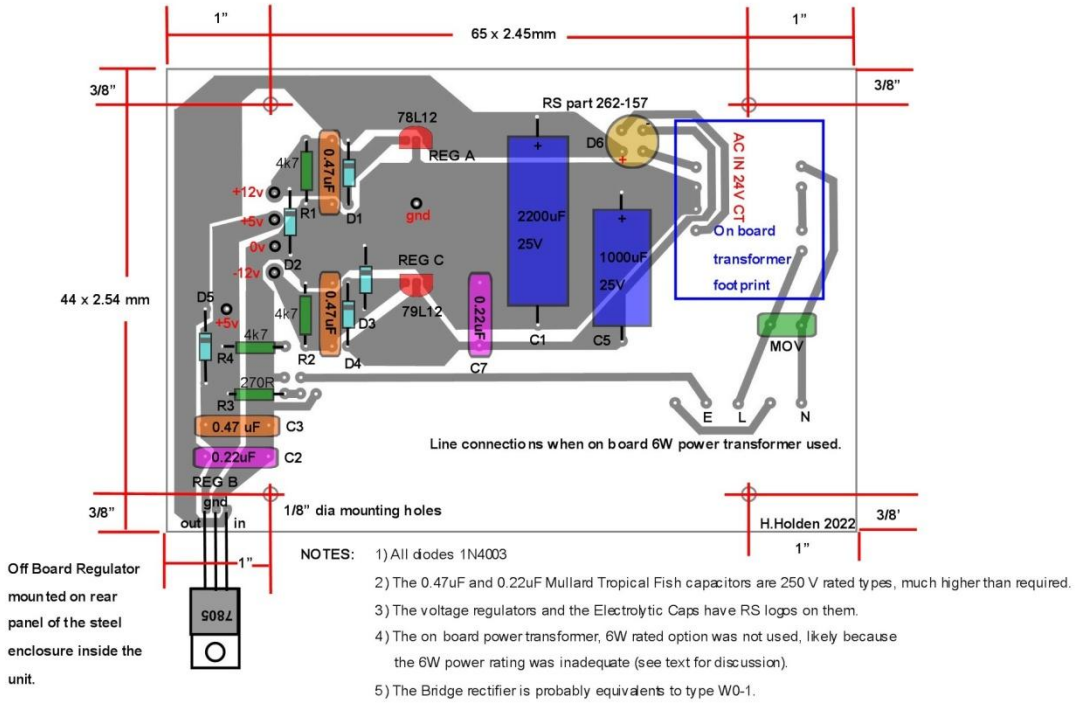


Of note: The 0v connection of the power supply (common) was not connected to the metal housing (chassis and line power Earth) . I changed this by removing the mica insulator under the 7805 regulator and screwing the regulator's tab directly to the chassis. The reason for doing this is explained later.

As can be seen, the power supply board is single sided. This image shows the track-work as if one were looking through the top of the board:



The diagram below shows details of the PCB's geometry and component overlay and some additional notes:



Power supply issues:

I found when I received the T-W unit, it was not working and no data was coming out the serial port. But the fellow in the UK who sold it to me said it was working and I believed him, he seemed like a very genuine person and a straight shooter.

I found, that running the unit here in AU, that there was a problem with the +12V supply. The reason was that the DC input to the +12V regulator was too low to allow the 78L12 regulator to regulate properly.

It was quickly determined that with a 230V line power input (that I what have here) and with the loading on the transformer via the 5V regulator, supplying the 28 IC's, the transformer's secondary voltage was too low for the 12V regulator's input. The output of the 78L12 being about 11V and the regulation had failed.

The DC output voltage was still satisfactory of course for the 5V regulator. And the rectifiers and capacitors were checked, without any issues.

If the load from the 5V supply was disconnected, the voltage into the 12V regulators' input came up to near 15V and the 12V regulator functioned normally.

Raising the line voltage up with a Variac to 243V, just allowed the 12V regulator to start regulating and put out the correct 12v.

The original chassis mounted transformer appeared to be about a 6W rated type by looking at its core size. The current drain on the main 5V rail was in the order of 300mA. Ideally, for the 78L12 12V regulator it would have a minimum of 14V at its input.

However, the transformer is a 24V CT type, or 12V a side and the peak off load DC voltage at best, per winding, would be about (12 x 1.41) – 0.7v or about 16V. The situation is not helped by the common line power sine wave distortion these days with modern appliances drawing current on peaks and flattening the waveform tops.

Under load, one could only afford about a 2v drop in output, prior to the 12v regulator which isn't much of a range. A quick test on the original transformer indicated that is secondary winding's DC resistance was about 6 ohms for each 12v winding and the primary resistance about 954 Ohms. Transforming this primary resistance by the impedance ratio and adding it into the secondary resistance indicated that the total internal resistance of the transformer as a whole was about 8.3 Ohms minimum, per 12V winding.

This means that with the 300mA current load for the 5V system, the transformer's output voltage would drop in the order of about 2.5V under load, possibly a little more. This combined with the low range line voltage here (although standard for AU at 230V), meant that the original

transformer could not give a satisfactory output for the 12V regulator's input voltage requirement, being at least 14v.

I replaced the power transformer in the T-W unit with the Digikey 12W rated part and this solved the problem for two reasons. One being that it has a much lower internal resistance (about 3 Ohms per 12V winding) and its voltage drops less under load and two, that its primary windings (in series) are designed for 230V, being 115V each. Probably the original transformer had a primary designed for 240V I suspect. This solved the voltage regulation issue, however:

One interesting thing happened; the low voltage (11V) on the 12V supply rail "unmasked" a borderline condition in the Reset circuitry, where one might not have expected that a change from 12V to 11V should have made any difference at all. As noted in the power on reset circuitry, on sheet 2, the value of R6 is 27k (as T-W fitted to the pcb). This must either be an error of design, or the wrong resistor fitted at the factory by mistake. There are no values marked on the board, so this is quite plausible. Possibly the original value might have been 2.7k and a 27k got fitted in error.

With the +12V rail slipping down to 11V, the logic level at pin 4 of IC 11, the 75154, did not quite reach a logic high for the IC. It was determined that the IC was normal (replaced with another) and the 10uF capacitor not leaking. The +11v & 27k combination was simply not providing enough current to get the IC's input pin to threshold, if the supply voltage was 11V and not 12V.

Checking the data sheet on the 75154, its input resistance is typically 6k, though it can range from 3k to 8k. Also, regardless of its operating mode involving the threshold pin, which affects the hysteresis and the low level threshold voltage, its high level threshold voltage is typically 2.2v. A quick calculation shows that in the worst case, of a 3k input resistance, a resistor required to pull up the input voltage from a +12V source could not be anything higher than about 13.5k. A 27k might just barely make it to threshold, if the IC specimen has an input resistance of 6k or higher. So, they were lucky it worked at all with the 27k fitted. Normally the correct pull up resistor is 6.8k, which would guarantee a pull-up to over logic high threshold, in all IC specimens.

Because the UART remained in reset, this explained why there was no serial data coming out of the UART. I regarded this condition as a little "marginal", even with the 12V regulator issue fixed, so I reduced the value of R6 to around 13.5k, by simply paralleling another 27k resistor with R6. Although lowering the 27k to half its initial value halves the time constant of the power on reset pulse, it is still long enough to perform a solid power on reset. If there is any concern about this the 10uF capacitor, C5, can simply be increased to 22uF. In this location it is better to use a Tantalum capacitor than an Electrolytic type too, as they have lower leakage.

In the topic of Tantalum capacitors, in my T-W unit I replaced the original blue 1uF Tantalum capacitor C3, a timing capacitor for Monostable one shot IC23, with a 1uF MKT 63v film

capacitor, because these old blue Tantalum capacitors are notorious for spontaneously shorting out.

Also, on this original unit I replaced the original DIP switches. These original pale blue CTS types are not sealed to the ingress of moisture and fumes. After they were removed, I cut the CTS switches open and the corrosion on the contact surfaces was fairly severe. I replaced them with 16 pin (machined round pin sockets) and into those I lugged in high quality new OMROM brand DIP switches. The overall height profile matches the original CTS switch and it has the advantage that later a defective switch is easily replaced.

(As an aside, if you have vintage computer gear that uses the pale blue CTS switches, I would recommend replace them all. Or you might have a very long road tracking down intermittent behaviour of your vintage computer pcb)

6) CIRCUIT ANALYSIS & OPERATING THEORY:

In this discussion I have started first with the way in which the primary and secondary addresses transmitted by the PET, are received by the T-W unit.

The primary address gains the attention of the T-W unit as a whole, the secondary address, in this machine, controls the PETSCII to ASCII converter. The secondary address uses a "second time around circuit" so that it does not respond to the initial primary address commands from the PET to the T-W unit.

There is some very interesting circuitry here involving data reorganisation & switching. The data bytes making up the contents of the text message and how these bytes are transferred from the unit to its output connectors is explained. Unlike the C-M printer adapter units, the T-W unit has an 8 bit output data latch. This 74LS374 latch is used to drive a 74LS244 buffer for the Printer output data and is used to feed the UART's data input pins too.

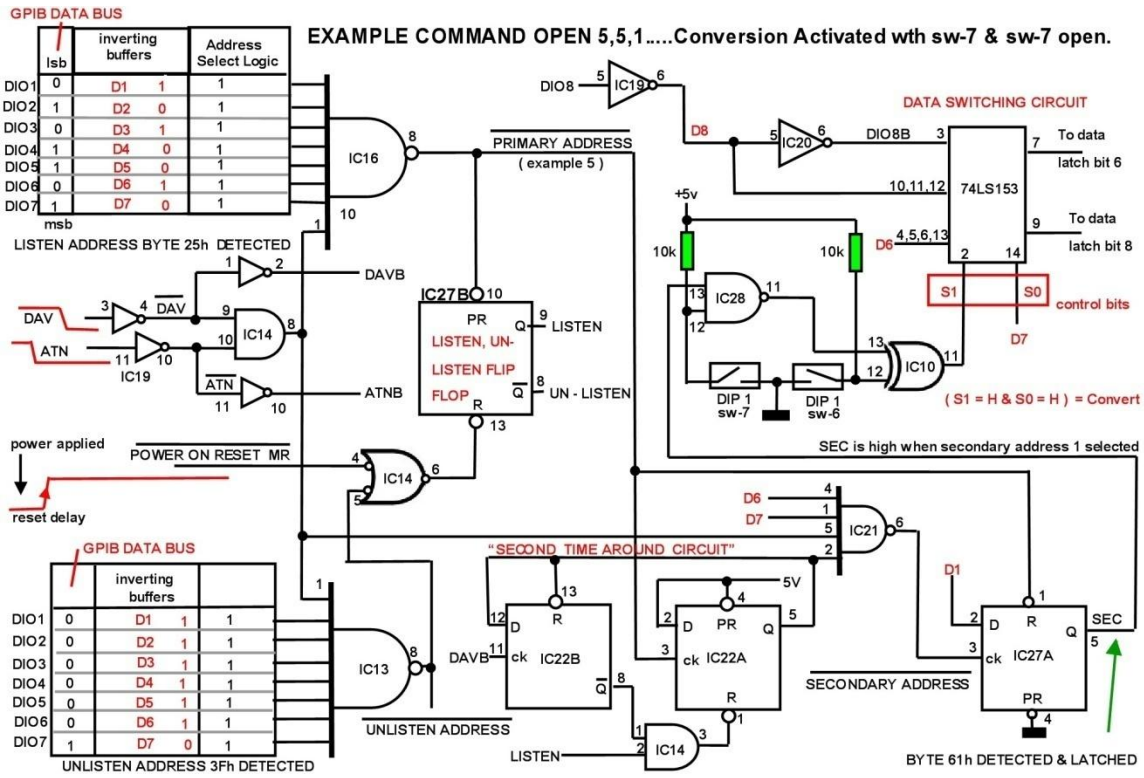


FIGURE 1.

Figure 1 above is a start to look at the circuitry involved in the primary and secondary address decoding.

The T-W unit behaves as a “Receiver” on the GPIB bus. The actual voltage values on the GPIB bus lines are inverted logic where a low represents a logic high and the high represents a logic low. Initially the 8 bit data is past to 74LS14 inverting buffers which are Schmitt trigger types to help noise immunity and these flip the data bits back into standard H = True logic.

The actual GPIB data bus lines are labelled DIO1, DIO2 etc, but in their inverted form I have simply labelled them D1, D2 etc. Signals that are double inverted, back to their original polarity they had on the bus, I have labelled DAVB (buffered DAV) or DIO8B (buffered DIO8) ATNB (buffered ATN).

When the unit is initially powered, or when the computer deploys the IFC line on the GPIB interface, a master reset pulse is generated (see circuit on sheet 2). This is passed to the “Listen Un-listen” flip flop via IC 14’s pin 4 and pin 6 to reset the flip flop IC27B so its pin 8 is high and pin 9 low and it is in the “un-listen” state. See Figure 1 above. Note that IC 14, pins 4,5,6, is an AND gate but drawn in its De Morgan equivalent form to help show its function, because all of its terminals are active low.

To gain the attention of a peripheral device on the GPIB bus, the computer deploys the ATN line low. (Of note, if both the ATN line is high, and the T-W unit not listening yet, Un-listen =high, the output of IC31 pin 11 is low and therefore NDAC and NRFD are forced to a high and cannot respond).

Shortly after ATN goes low, the computer deploys the DAV line low, to signify that the data to be sent is stable & available on the GPIB bus. In the diagram Figure 1 above, this indicates that the output of IC14 pin 8 will go high because /DAV and /ATN both are now high.

The primary address byte is on the GPIB data bus and is stable there before the DAV line went low. This byte is the address that the T-W responds to. It is byte 25h for an IEEE-488 device set for primary address 5. Most peripheral devices, like the T-W unit have DIP switches that allow its primary address to be changed. This is done with the help of XOR gates. See sheet 1 for the arrangement.

If the primary address placed on the bus, in this case byte 25h matches, then the output of IC8 pin 8 goes low, this is the /PRIMARY ADDRESS signal. It sets the flip flop IC27B pin 9 to logic high into the “LISTEN” condition. And pin 8 of this flip flop is goes low.

(The flip flop IC27B stays in this Listen state until all the data bytes are received , passed through the unit to the Printer or terminal and until the Unlisten byte, 3Fh, is transmitted by the PET and is received and detected by the T-W unit. Unlike the Print# 5, “some message” example for sending data to the T-W unit, the CMD command does not transmit the 3Fh, so the unit remains in the LISTEN state. This is useful for LISTing BASIC programs. IC 13 is configured to recognise the byte 3Fh and this causes pin 8 of IC 13 to go low, resetting the Listen-Unlisten flip flop IC27B, via IC 14’s pin 5 input & 6 output).

To complete the transaction involving the primary address, while the ATN line is low, the T-W unit must notify the computer that it has received the primary address and responded to it. When the DAV (data valid on the bus) line went low, the T-W unit responded to the PET by setting the NRFD line (not ready for data line low), signifying to the PET that it is ready to receive a byte of data.

After the address byte is received by the T-W unit and after a time delay (created by an RC network in the T-W unit) the T-W unit asserts NDAC (not data accepted) line high, to inform the PET that the data byte was accepted. The PET then responds by setting the DAV line high. This is a typical GPIB handshake.

The circuitry in this initial **addressing mode** is fairly simple, when the ATN line low, shown as the red logic levels on the diagram of Figure 2 below.

When ATN is low (and signal ATNB low), during the recognition of the primary and or secondary address, some of the gating that is operational when data bytes (message content) are being sent is disabled. In this "address receive mode", the DAV signal passes to become the NRFD signal in a non-inverted form and to the NDAC in an inverted form. However, in the NDAC signal case, there is an added small time delay for the NDAC signal, acquired from the 560R & 2200pF RC network shown in Figure 2 below on pin 8 of IC28. In this "address receive mode", none of the monostable one shot generators are active (triggered) because pin 11 of IC 28 is stuck high due to the low ATNB signal:

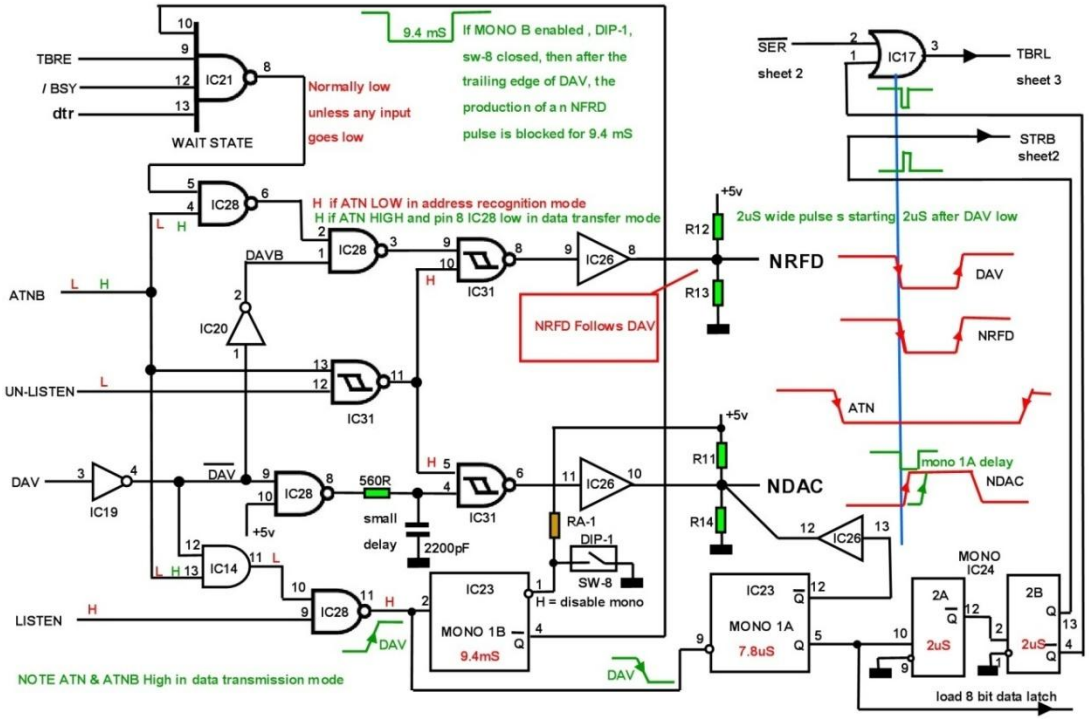


FIGURE 2.

In the case that the primary (and or secondary) addresses are specified and the ATN (and ATNB) signal goes high (shown in green H in figure 2), then the T-W unit is ready to receive the message byte data. The LISTEN line is high. The DAV signal passes via IC 19 pins 3 & 4, IC14 pins 12 & 11, IC28 pin 10 &11, to trigger the Monostables. The Monostables are one shot pulse generators. IC23 pin 2 is strobed and responsive to the DAV line going high (IC23 mono 1B responds if enabled by DIP-1 sw-8). When DAV goes low (mono 1A responds).

Re- presenting Figure 1 to avoid scrolling back to it:

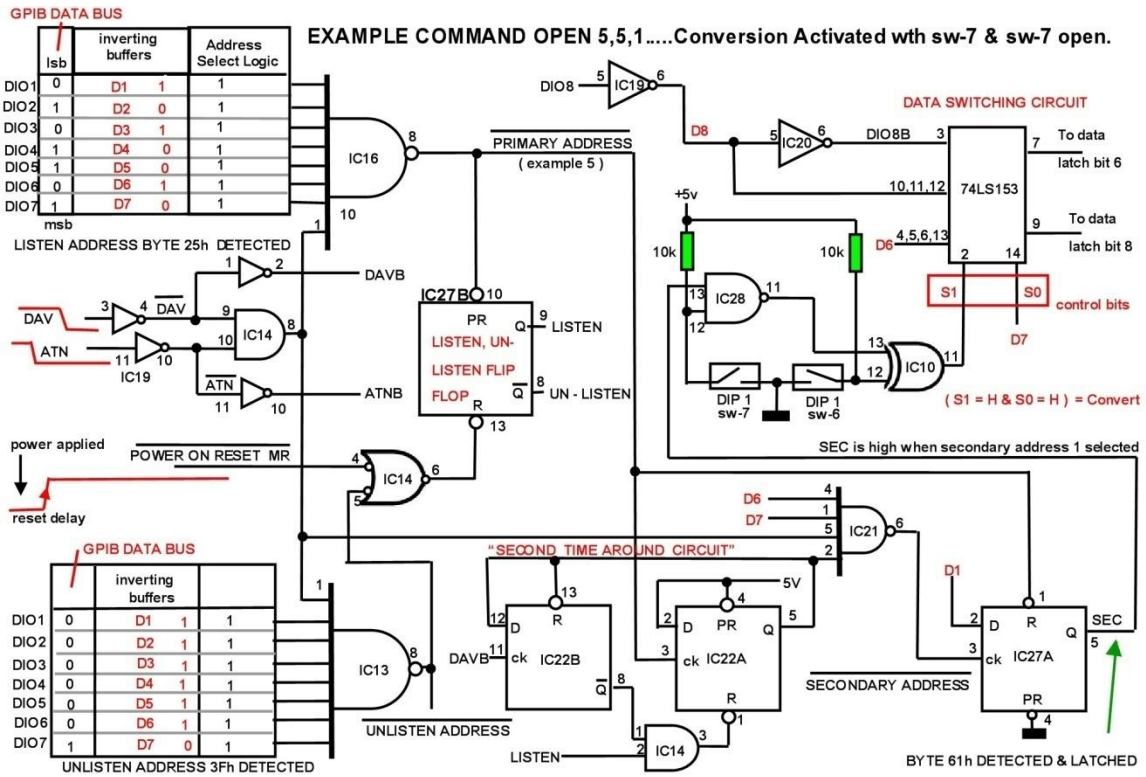


FIGURE 1. (again)

In the address receive mode, flip flop IC22B is clocked by the rising edge of DAVB, and IC22A is clocked by the rising edge of the /primary address, pin 8 of IC 16.

However, initially at least, after a reset (of the three Reset types previously mentioned) neither flip flop of IC 22B or IC 22A can respond, as they are held in a reset state, by pin 3 of IC 14 being low, because the LISTEN is low.

It is not until the master control "listen Un-listen" flip flop IC27B goes into the Listen state after the /primary address goes low and LISTEN goes high, that pin 3 of IC14 can go high. This releases the reset on IC22A's reset pin 1 and then IC22A is able to be clocked by the rising edge of the /primary address signal from pin 8 of IC16, after the primary address handshake has taken place.

In summary; only after the primary address has been received and IC 22A has been both released from reset and clocked by the rising edge of the primary address pulse on pin 8 of IC16, to give a high output on pin 5 of IC22A, can IC 22B then respond to a clock pulse, DAVB going high for the ***second time around***, which DAVB does, after the handshake involving the secondary address.

After the primary address handshake, for the /primary address to go high on pin 8 of IC16, the DAV must have already returned high, placing a low on pin 9 of IC14 and thereby removing the high from in 8 of IC14 and pin 1 of the address decoder IC16, making pin 8 of IC 16 go high.

Meaning, that the DAV has already returned high, for a short period, before the /primary address line of IC16 pin 8 goes high.

Since the /primary address line going high, clocks IC22A, which then releases the reset on IC22B *after DAV has already gone high*, then IC22B is not in a condition soon enough to respond or be clocked by the rising edge of the DAVB signal during the end of the primary address handshake.

IC22B can only respond to a clock pulse ***after the second time around*** when DAVB (or DAV) goes high.

Only after the primary address acquisition can the output of IC22A pin 5 be clocked high. This high completes one of the four inputs required to activate IC 21 and take its pin 6 low. Pin 6 of IC21 has to be low before it can rise high to clock the '74 flip flop IC27A. The other three inputs to IC21 are only high when both DAV and ATN are low (in an addressing state) and when data bits D6 and D7 are low.

When a secondary address of 1 is placed on the bus, with the command OPEN 5,5,1 as an example, and the Print# used or the CMD to send out some data, then after the primary address of 5 is received, then on the second handshake D6 and D7 are low both ATN and DAV are low and a 1 is placed on the bus, making data bit D1 high.

As can be seen, pin 6 of IC21 is fed to the clock input pin 3 of IC27A and data bit D1 is fed to its data input, which is high with the 1 as the secondary address. At the end of this second address handshake DAV goes high, but this second time around, IC22B is clocked. This results in a low on pin 8 of IC22B, forcing a low on the reset pin 1 of IC22A and causing a low on IC22A's

output pin 5. As a result the output of IC21 pin 6 goes high and in the process clocks IC27A. Therefore, if the data bit D1 was high, this bit is clocked and "remembered" on pin 5 of IC27A.

I called this latched signal on pin 5 of IC27A "SEC" on Figure one. It will be high when a secondary address of 1 has been issued by the OPEN command, and low if no secondary address issued. The SEC data bit is used to switch off, or switch on the PETSCII to ASCII converter in the T-W unit.

One purpose of IC22B is to terminate the effective length of the detected secondary address signal, the second time DAV goes high.

T-W arranged the PETSCII to ASCII converter (see Figure 1) with 2 DIP switches and using an AND gate IC28 (output pin 11) and an XOR gate IC10 output pin 10. The two DIP switches provide 4 combinations of functions:

1) Conversion switched hard **on** SW-7 ON & SW-6 ON

2) Conversion switched hard **off** SW-7 ON & SW-6 OFF

3) Conversion **on** with secondary address 1 SW-7 OFF & SW-6 OFF

4) Conversion **off** with secondary address 1 SW-7 OFF & SW-6 ON

As noted above with both the DIP switches in the off position, the secondary address 1 simply activates the PETSCI to ASCIII conversion.

How the PETSCII to ASCII conversion is achieved:

Two control bits S0 and S1 are applied to a 74LS153 data selector IC. The output of IC10 pin 11 controls S1 and data bit D7 controls S0. The way the 74LS153 has been wired is that there are only two possible switching scenarios. One scenario leaves the data pathway unaltered, with the data bits D1 to D8 passed to their respective output data latches. The other scenario occurs with S0 and S1 both high and the "switch around" or PETSCII to ASCII data conversion occurs.

Figure 3 below repeats the circuit fragment:

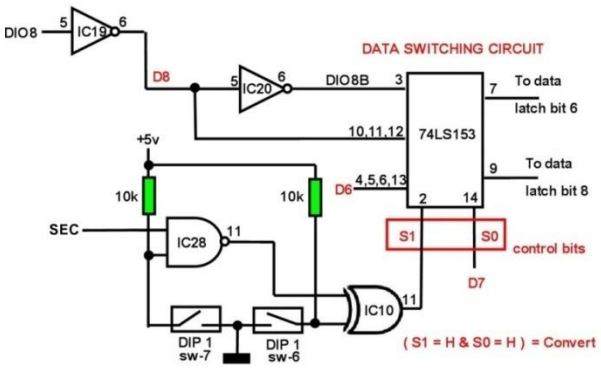


FIGURE 3.

Figure 4 below shows the two conditions of conversion, versus no conversion achieved by the condition when D7 is high and the conversion enabled by SEC (secondary address of 1 specified in the OPEN command).

In essence data bit D6 is disconnected and routed out as data bit D8.

Data bit D8 is “disconnected” and it is signal inverted by IC20 and routed out to replace data bit D6.

The example below indicates that a PETSCII lower case “a” (code 41h) gets converted to code 61h which is ASCII lower case “a”. With this conversion in operation, with the PET in the lower case mode, both upper and lower case letters appear correctly in the ASCII receiving system (such as a modern computer terminal on a serial link connected to the T-W unit instead of a printer).

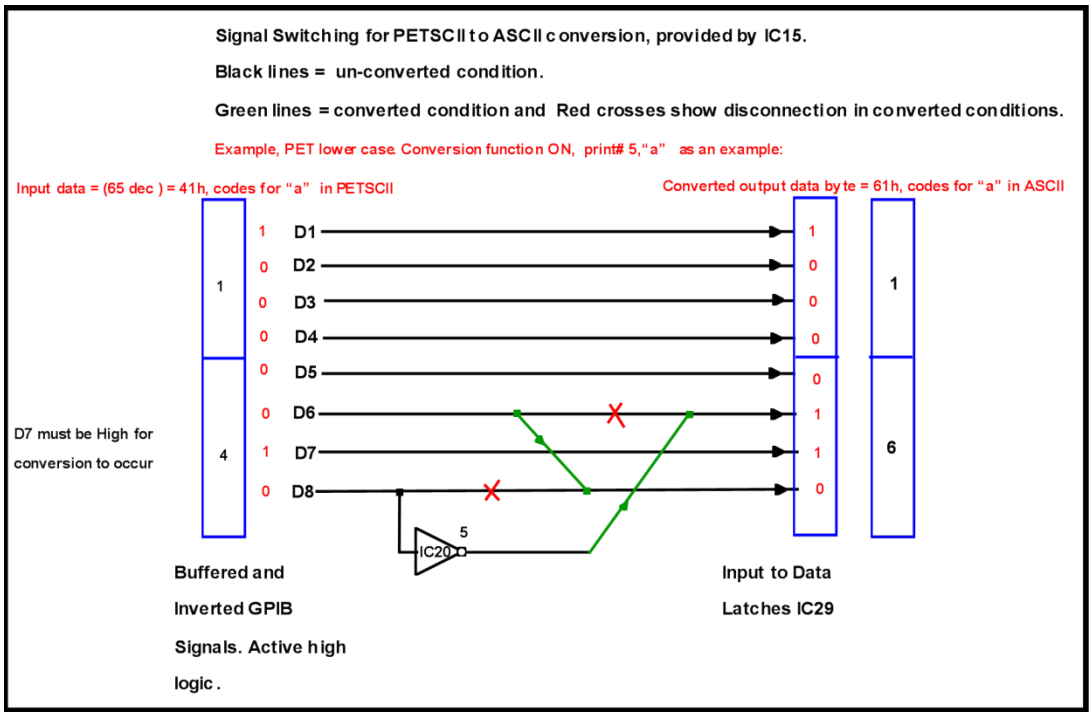


FIGURE 4.

The results of course are affected by other things, for example if the serial link to the modern computer is set up as a 7 bit or an 8 bit system. Also, if the PET is in upper or lower case mode. The table Figure 5 below shows examples just looking at the letters A and a:

PETSCII to ASCII CONVERSION EXAMPLE- PERFORMED BY TAYLOR - WILSON PRINTER INTERFACE.

PET UPPER CASE MODE- CONVERSION OFF (ASCII-X)	PET's VDU DISPLAY	OUTPUT FROM WILSON TAYLOR SERIAL PORT SEEN ON A MODERN COMPUTER WITH TERATERM	
PRINT "A" or PRINT CHR$(65)	A	7 BIT OUTPUT	8 BIT OUTPUT
PRINT "♠" or PRINT CHR$(193)	♠		
PRINT# 5,"A" or PRINT# 5, CHR$(65)		A	A
PRINT# 5, "♠" or PRINT# 5, CHR$(193)		A	⊥
PET UPPER CASE MODE- CONVERSION ON		Provides a way to print text in all lower case:	
PRINT# 5,"A" or PRINT# 5, CHR$(65)		a	a
PRINT# 5, "♠" or PRINT# 5, CHR$(193)		A	A
PET LOWER CASE MODE- CONVERSION OFF (PETSCII)			
PRINT "a" or PRINT CHR$(65)	a		
PRINT "A" or PRINT CHR$(193)	A		
PRINT# 5,"a" or PRINT# 5,CHR$(65)		A	A
PRINT# 5,"A" or PRINT# 5,CHR$(193)		A	⊥
PET LOWER CASE MODE- CONVERSION ON		Note correct conversion PETSCII TO ASCII :	
PRINT# 5,"a" or PRINT# 5,CHR$(65)		a	a
PRINT# 5,"A" or PRINT# 5,CHR$(193)		A	A

NOTE: This symbol is part of the Extended ASCII Character set.

FIGURE 5.

In the upper case mode (called ASCII-X by some), the T-W unit conversion provides a tool to be able to print the text in all lower case mode, where it otherwise comes out as upper case.

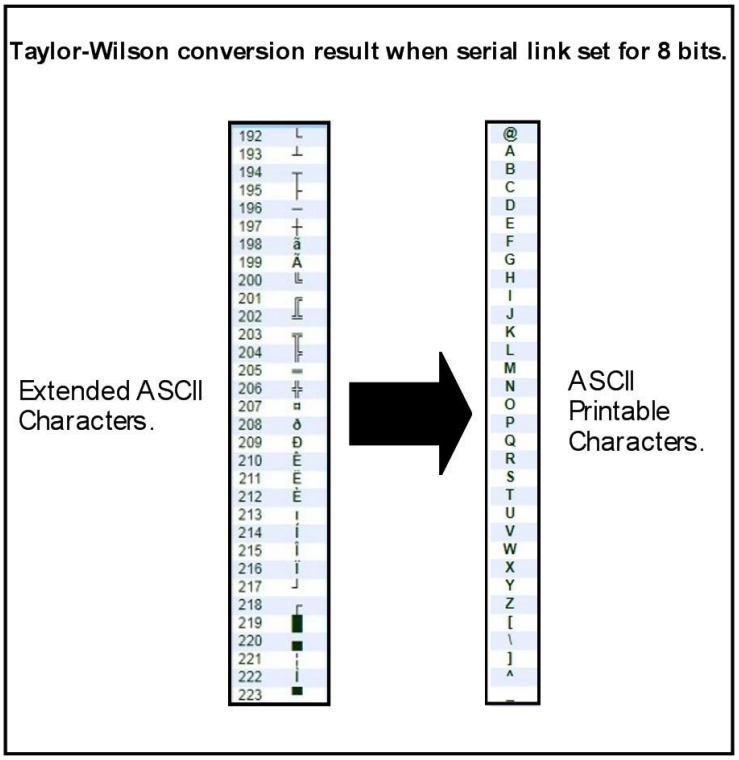
The great thing is that with the PET in lower case mode the T-W conversion results in lower and upper case PETSCII characters being printed correctly on a modern serial terminal or printer.

With the conversion deactivated, it all appears as upper case (in the 7 bit serial output example). However, if the serial port is configured for 8 bits, then characters from the extended ASCII set appear when the conversion is switched off.

(The ADA1200 printer adapter, when confronted with the PETSCII output of upper & lower case characters, the upper and lower cases are incorrect and are reversed. And when presented with upper case from the PET in upper case or ASCII-X mode, it prints upper case. I do not have the ADA1450 to try).

In the case that the serial link (from the T-W unit to a computer terminal, instead of a printer) is set for 8 bit data, then **without the conversion running**, code numbers from CHR$(192) to CHR$(223), which represent graphic symbols in the PET's upper case mode and mainly lower case letters in the PET's lower case mode, result in the extended ASCII set being displayed on the computer.

However, with the T-W's PETSCII to ASCII conversion running, these come out as upper case letters:



PETSCII to ASCII Conversion Summary:

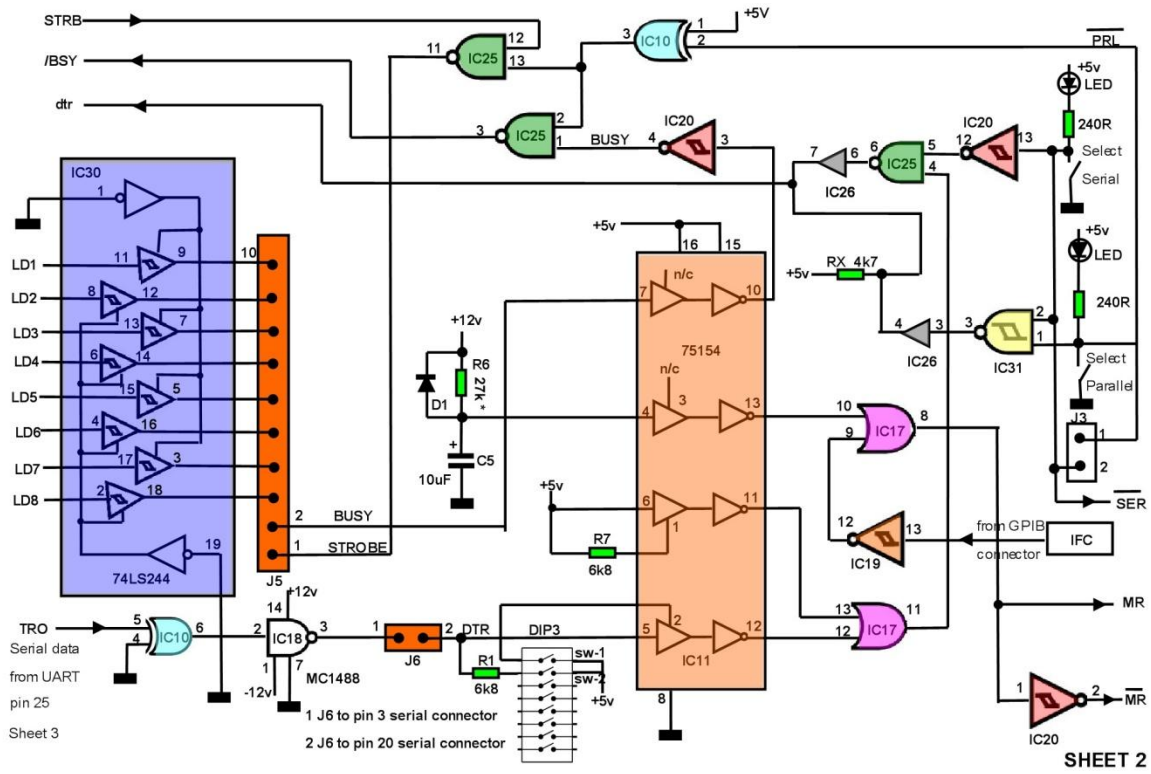
1) The conversion allows a way to convert upper case letters to lower case letters on a print out, with an 8 bit, or 7 bit, serial link to a terminal, when the PET is in the upper case mode.

2) With an 8 bit serial link to a computer terminal, the conversion provides a way to prevent the extended ASCII being displayed and displays upper case letters instead of ASCII extended graphics symbols, when the PET VDU is displaying its own graphics symbols over the code range C0h to DFh.

3) With an 8 bit or 7 bit serial link, when the PET is in lower case mode, the conversion results in both capital and lower case letters being displayed correctly on a Printer or computer terminal. This is clearly the main intention of the conversion that T-W created.

Two modes of operation:

Serial Printer & Parallel Printer modes. These are selected by the front panel switches. Looking at sheet 2 again:



The two front panel select switches select the unit to interface with a parallel or series printer. Both connectors are provided on the rear panel.

In the case that the Serial switch is closed, this illuminates the LED above it on the panel. This takes pin 5 of IC25 high and this allows the signal from pin 2 of connector J6 (which is pin 20 of the RS-232 panel connector = **DTR**) to pass via IC17 pin 12 & 11 to pin 4 of IC25 and then to pin 6 because it has undergone two inversions. So pin 6 assumes the logic state of DTR. This signal, via open collector buffer IC26 pins 6 & 7 create the **dtr** signal, which is responsive to the signal on pin 2 of J6 and 20 of the rear panel serial RS-232 D connector.

DIP 3 sw-1 alters the input threshold of IC11 pin 5. And sw-2 adds in a 6.8k pullup resistor to +5v. This is so the received **DTR** signal can be accepted as:

Both closed = RS232 level with pull-up.

Both open = TTL level with no pull-up.

S1 closed S2 open = RS232, with no pull-up.

S2 closed S1 open = TTL with pull-up.

Generally with the 75154 IC 11, when the threshold terminal is left open, the gate's input behaves similarly to a standard TTL in with a hysteresis loop between 0.8 and 2V.

When the threshold input is tied high there is a larger hysteresis loop and the input voltage has to fall below zero volts (as it does with RS-232 signals) to get the gate to change state.

With the front panel serial printer switch closed (and the select parallel printer switch open) the output of IC31 pin 3 is high and therefore the output of IC26 pin 4 does not interfere with IC26 pin 7. In addition, with the parallel switch open, and IC 10 (output pin 3) an XOR acting as an inverter, then pin 2 of IC25 is low. This ensures that pin 3 of IC25 is high and that the control line **/BSY**, used by the parallel printer, remains high, regardless of any signals that might be on the **BUSY** line from the actual parallel printer if one happened to be plugged onto the T-W unit at the same time the serial port is being used. (unless the parallel switch is turned on too)

In the case that the Parallel switch is on and the Serial switch off, the line /PRL is low, **dtr** is forced high. This raises the question of the purpose of IC31 output pin 3 and IC26 pin 3 & 4 , because it remains with its output high in both cases if either the Serial or Parallel mode is selected.

However***, if neither select switch is on***, IC 31's in 3 output goes low and the low generated in IC 26 pin 4 causes **dtr** to go forced low. This activates the wait IC28 (output pin 8). So the default state with neither of the front panel select switches closed, the T-W unit cannot respond to the PET because NRFD is inhibited and stuck low. The command OPEN 5,5 Print# 5, "some message" for example causes the PET to hang, because after the PET puts data on the bus, it checks if NRFD is high, if low, it goes into a waiting loop, prior to setting DAV low. I would think that the logic behind doing this was to alert the user of the T-W unit, that something was wrong when neither of the printer panel switches were selected.

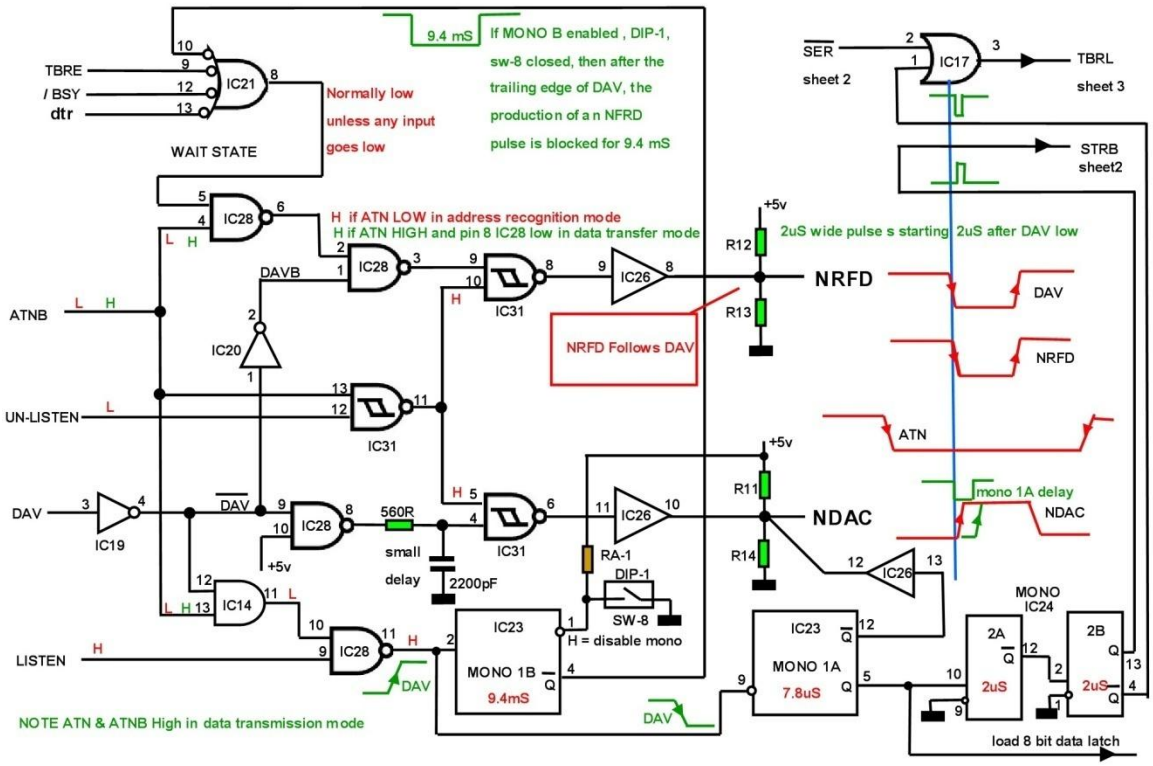
If both the printer select switches are on, with no parallel printer plugged in at least, the serial functions appear to work, and the floating BUSY line does not appear to cause an issue. It is possible that the T-W unit could simultaneously drive the two types of printers.

If the T-W unit is powered off, both NRFD & NDAC are high, BASIC reports a device not present error when OPEN 5,5 Print# 5, "some message" is executed.

Pin 20 of the RS232 connector is the DTR (Data Terminal Ready) The Serial Printer sends a signal out on the DTR line. When the printer's buffer input capacity has got close to only enough space for 30 characters the DTR line is deployed "false" or low, as is the **dtr** signal following it. This causes the T-W unit, in this case, to suspend sending characters until the DTR goes high again.

Transfer of text DATA bytes from the input to the output of the T-W unit, the DATA TRANSFER MODE:

Repeating Figure 2, but with IC21 drawn in its De Morgan equivalent:



The Logic levels of note when the ATN and ATNB line (buffered ATN) is high, in data transfer mode, are shown in green.

After the primary and/or secondary address handshake and now with ATNB high, the circuit is in the “data byte transfer mode”. In this mode, the circuit function, for the NRFD pulse can be stopped or inhibited by a “Wait State” circuit, the output of IC 21 pin 8.

The reason for the wait states is that the object receiving the byte data, in this case the T-W units target output device which can be either the UART and the serial printer it is feeding, or the Parallel printer that the T-W unit is feeding, has to acknowledge that the data is received and that the peripheral device is not overloaded with data.

In addition, the T-W unit itself has to complete its own handshake, per received byte, to tell the PET that the byte was successfully loaded into its latches. For this function a 7.8uS pulse is used to delay the rising edge of NDAC after the data has been latched during the duration of this pulse.

Assuming ATN is high and DAV goes low, the data handshake begins. If the output of the wait state IC 21 pin 8 is low, then the falling DAV low signal creates the falling low NRFD signal via gates IC19(4),IC20(2),IC28(3),IC31(8),IC26(8), where the numbers in brackets refer to the output pin numbers on those gates.

Also, assuming DIP-1 sw-8 is open, we will ignore mono 1B for now, the falling edge of DAV triggers mono 1A. The low going DAV would have also set NDAC high (with a small RC delay), but in this mode because mono 1A has been triggered, its /Q output goes low and via the open collector gate IC26(12) holds NDAC low for 7.8uS.

Now it becomes obvious why the RC delay was created in the signal chain that generates the rising edge of NDAC from the falling edge of DAV…it is so the mono 1A can clamp the NDAC low for the 7.8uS time after DAV goes low, without there being a glitch pulse being generated in the NDAC signal at the time just after DAV goes low.

During the time that NDAC is held low for 7.8uS, the Q output of mono 1A is used to latch the 8 bit data into IC29 the 74LS374 data latch. The 8 bit data is now stable and latched and available on the outputs LD-1 to LD-8. This data sent via a 74LS244 buffer directly to the printer connector and it sent directly into the 8 data inputs on the UART.

When mono 1A finishes it 7.8us pulse, the NDAC signal is released and goes high. The PET detects that the NDAC has gone high (indicating the T-W unit has accepted the data).The pet then asserts DAV high, which would cause NRFD to go high too, unless any of the wait state lines feeding IC21 were low, not allowing NRFD to go high.

Considering the T-W unit set for serial operations first:

When mono 1A was triggered, on DAV falling low, its Q output was fed to mono 2A and that feeds mono 2B. A pair of narrow 2uS pulses are generated, which start about 2uS after DAV fell low.

One pulse is sent via IC17(3) to the TBRL (transmitter buffer register load) input on the 6402 UART. This is to latch the input data into the UART's Transmitter Buffer Register.

When the TBRL pulse goes high again (2uS later) the UART acknowledges this by setting its TBRE (transmitter buffer register empty) output low and the transmitter buffer register is emptied but the data has been transferred inside the UART to its transmitter register.

TBRE is possibly low for a clock cycle, at 1200 BAUD that could be up to 52uS. Since this causes IC21's pin 8 output to go high, this disables the NRFD (holds it low) for at least this time period, even if DAV has already gone high. So during this time, the T-W unit cannot accept the next new data byte because it is not ready for new data.

In addition, if the serial output data from the UART, has started to near over-fill in the buffer in the printer, the **dtr** signal (as noted before) falls low. This takes pin 13 of IC21 low and its output high and this also disables NRFD going high, until that condition is resolved.

While all this was happening as noted, in the serial mode, the /BSY line remains high and cannot be influenced, even if there was a parallel printer plugged in. But it could be affected, if the parallel printer was plugged in and both the panel Parallel and Serial switches were both switched on at the same time.

If the UART is unable to dump out its serial data before the next byte arrives and TBRL pulse occurs, simply TBRE stays low, and does not return high until half a UART clock cycle after the end of the last transmitted serial data bit has been transmitted. Therefore the NRFD line is held low until this has happened too, because TBRL is low.

Therefore, before the T-W unit can receive the next byte and the NRFD line go high:

1) The UART must have received the previous data byte.

2) The UART must have successfully transmitted the whole previous data byte.

3) The Printer must have received the previous byte and not stalled the process because its buffer is filling up too much.

Application of mono 1B:

This is similar to a Bureaucrat. Its application is to slow the game down and not contribute any other useful inputs. If it is allowed to be active with DIP-1 sw-8 closed, voting it in, it generates a long low going 9.4mS pulse on its /Q output which feeds into pin 10 of IC 21. This also delays the time that NRFD can return high. And this limits the possible data transfer rate to about 100 characters (Bytes) per second maximum, according to the sticker on the base of the T-W unit, with the DIP switch setting instructions. Imagine trying to load cattle onto a boat, after a few are let on, a man closes the gate, waits some time, opens it again and lets some more cattle through. So it takes much longer to load the boat.

The case of the Parallel printer mode:

In this case the Serial select panel switch is open and the Parallel one close /SER line is high (pin 2 of connector J3) forcing the state of the TBRL line high via pin 2 and 3 of IC17. This is so any UART activities cannot interrupt NRFD and TBRE remains high. Referring to sheet 2, the output pins of IC25(6) and IC31(3) are both high. Therefore the open collector gates of IC26(7) and IC26(4) are not conducting. Therefore the **dtr** line (used in the serial mode) is stuck high and it cannot interrupt NRFD either.

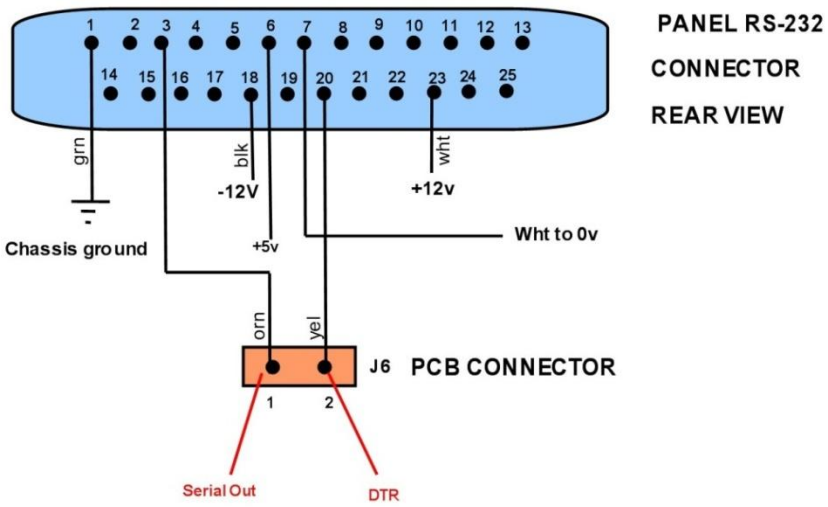
Therefore the only pin that can slow the data transfer proceedings and inhibit NRFD going high (aside from the Bureaucrat monostable) is the /BSY signal presented to pin 12 of IC21. As noted

before, in the parallel printer mode /BSY originates from the BUSY signal on the printer connector, see sheet 2 again. It is gated through IC 25(3) and inverted by it in the Parallel printer mode. Therefore, if the printer is busy, the T-W unit cannot accept the next data byte.

The logic shown on sheet 2, also shows that in the parallel printer mode the **STRB** signal is gated through IC25(11) to the printer's strobe connection. The **STRB** signal originates from the Q output of mono 2B and is again a 2uS wide pulse, starting 2uS after the DAV goes low, (at this point the data has already been latched by the T-W's 74LS373 latch). In this parallel printer mode the 2uS pulse is use to strobe the parallel latched data LD-1 to LD-8, via the 74LS2444 buffer IC into the printer's internal buffer.

Panel connector to pcb connections:

As noted these were done with multi-coloured ribbon cables. It required that these be unsoldered from the PCB and some connectors made to be able to free the main PCB from the unit for examination.



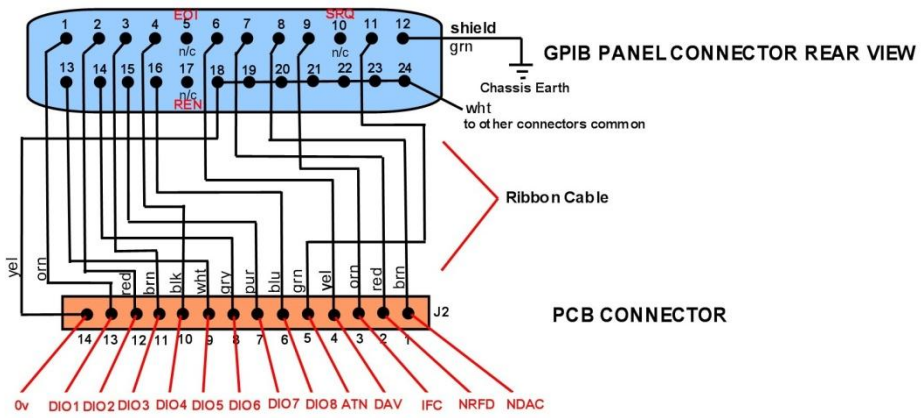
Notice how T-W applied the 3 power supply rails to pins on the RS-232 connector:

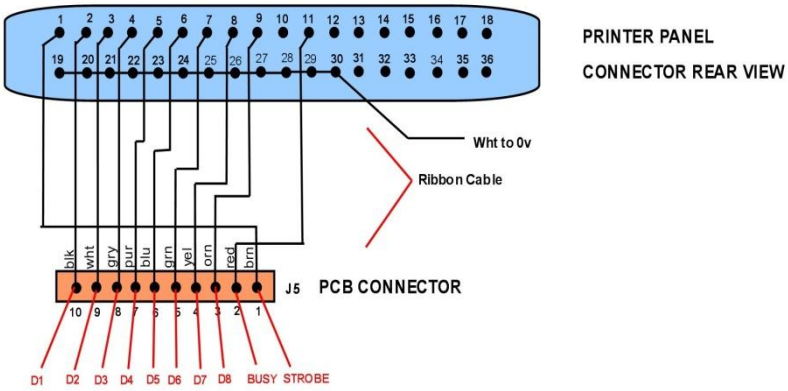
+5V applied to pin 6 which is the DSR pin.

-12V applied to pin 18, which is an undefined pin.

+12V applied to pin 23, which isa data signal rate selector in some modern systems.

(These had no obvious application in serial printers of the time, so they are better disconnected to avoid any issues)





Of note, T-W made no connection to the signal ground pin 16 or the chassis ground pin 17 in the T-W unit. I added a chassis ground to pin 17.

Note about general grounding:

I had mentioned previously, that the T-W unit's 0V line on the power supply was left floating.

This means that the 0V line acquires its Earth potential from the device, parallel or serial printer that it is connected to, or the PET's GPIB connector . In the PET the signal commons at the GPIB connector are at ground potential, due to the PET having an earthed body and the power supply common in the PET being connected to that, which is sensible.

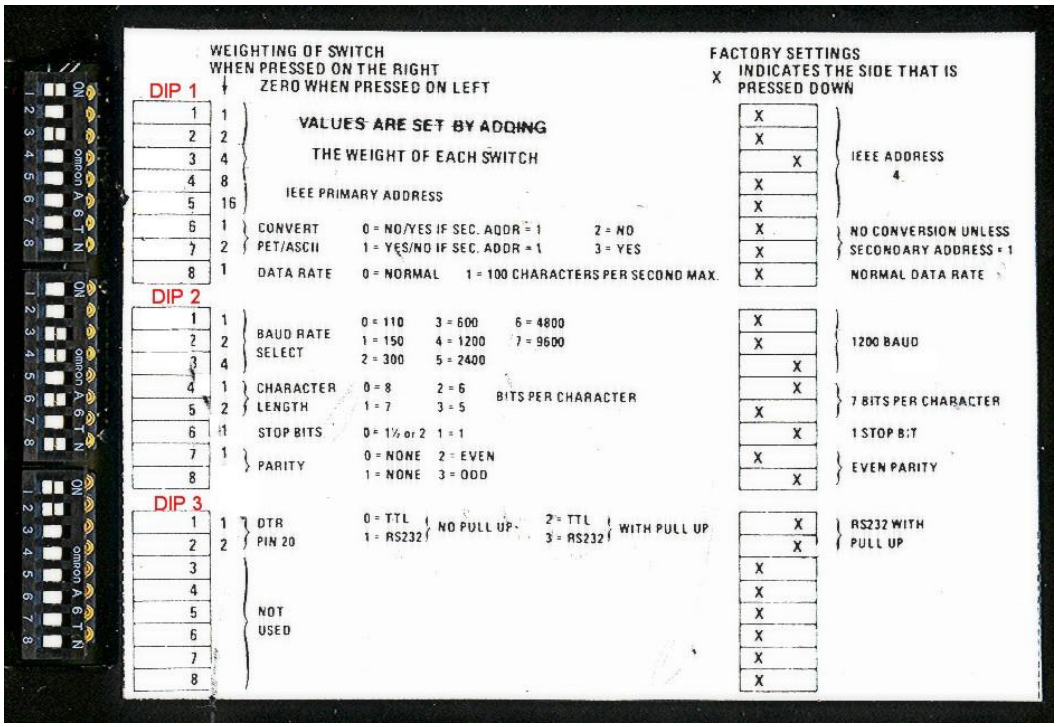
Not grounding a peripheral unit's 0V line to its chassis (or grounding it via a low R value resistor) can have some theoretical benefits in preventing earth loop problems, but these are not an issue with logic level signals of various digital units connected together and all plugged onto the same GPO (general purpose outlet) and Earth point there. If they are in different building, possibly some benefit.

However, with nothing connected to the serial or parallel connector, or the GPIB link to the PET, the 0V rail (and the power supplies +5v,+12v &-12v) in the T-W can acquire an unspecified voltage and charge up electrostatically above ground, to some unknown level. Hot plugging events can then generate transient discharge currents which can potentially damage the IC's which drive the GPIB bus in the PET or the receive data IC's in the T-W unit.

It is much better therefore if the 0V line in the T-W unit is grounded and all the interconnected devices have the same earth potential on their 0V lines. It was easy to do this in the T-W unit, simply by removing the mica insulating washer under the tab of the 7805 regulator. This also improved its thermal bond to the chassis and fresh thermal compound was applied.

7) Using the Taylor-Wilson unit with the PET & setting the DIP switches.

Fortunately this is straightforward because T-W put a large sticker on the base of the unit with the DIP switch settings. This makes the unit very easy to configure:



As can be seen above the DIP switches have been replaced with high quality OMRON types, the original blue CTS switches had significant internal corrosion.

8) RESTORATION OF THE TAYLOR-WILSON UNIT.

As noted when I received the unit it was not working. The issue of the incorrect resistor value R6 was sorted out as was the power supply with a new transformer.

Plugs were fitted to allow removal of the ribbon cables from the main PCB and single pin connectors added to the power supply board.

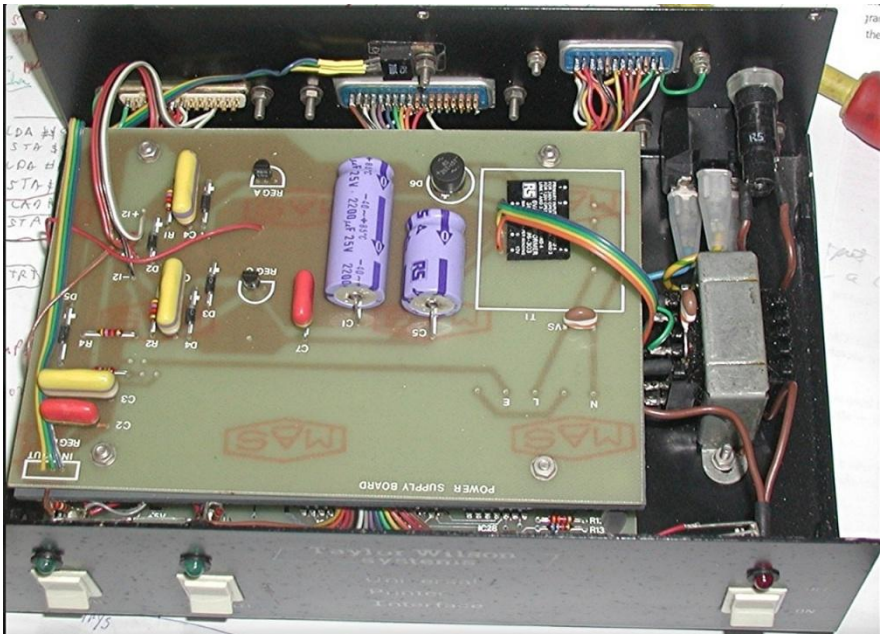
The wiring of the ribbon cables to the connectors had a poor standard of soldering, so the connections were re-done and the connector pins sleeved with heat shrink tubing.

The unit's top cover had multiple rust spots, the top cover was rubbed down and the rust treated with Fertan organic rust converter. The top was re-sprayed with a slightly off-white Dupli-Color paint which matched the original.

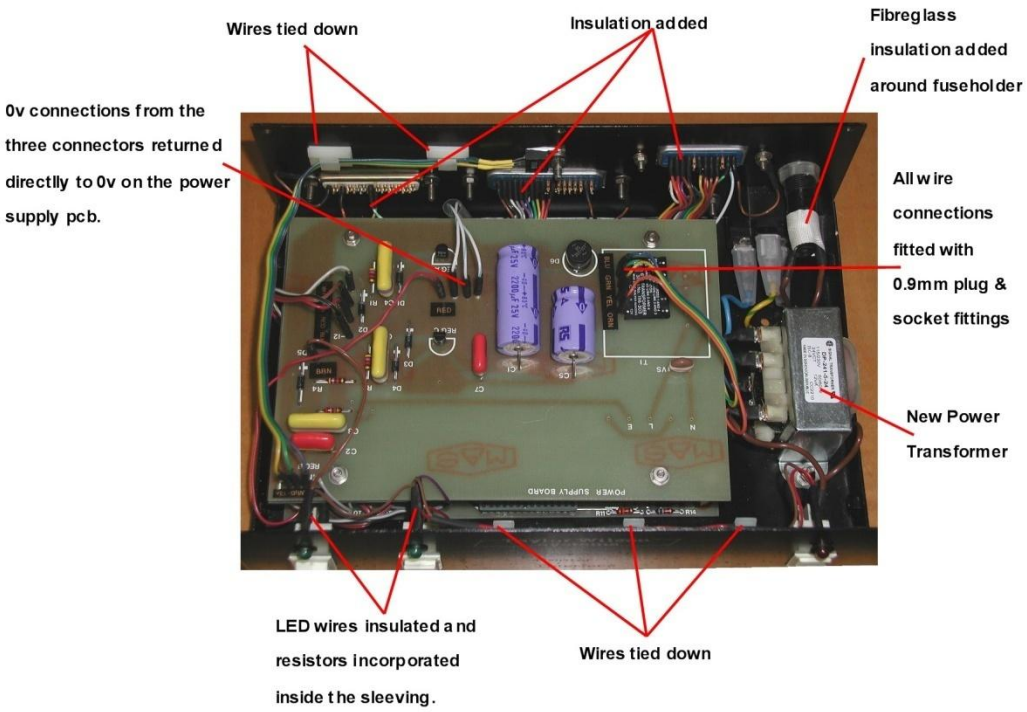
The original case side retaining screws were missing. They were machine threaded 4BA types with welded nuts inside the chassis flange. Somebody had screwed in self threading screws. Luckily the threads survived. I was able to buy new dome head counter-sunk machine screws on UK eBay.

I have attached a photo of the T-W unit as I received it. Of note the wires are soldered directly into in power supply pcb, as the ribbon & other cables are into the main pcb. In this photo the view of the main board is obscured.

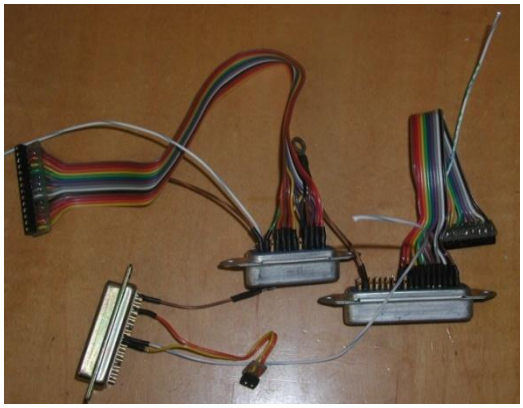
The photo also shows the original power transformer (that was inadequate) and the wires soldered to the RS-232 connector, the Printer connector and the GPIB connector. None of which contained any insulating sleeves and the soldering quality was not ideal.



The photo below shows the modifications I made to the unit to improve access and serviceability:

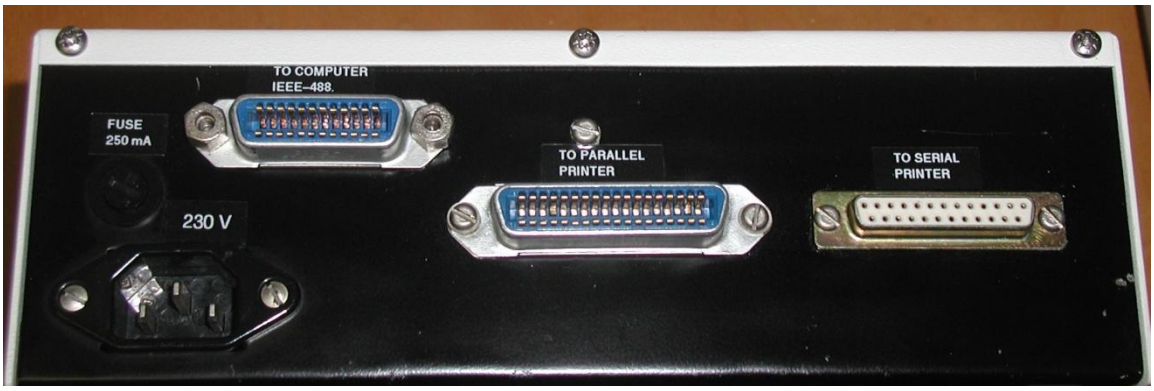


As noted previously, connectors were fitted to the main board. This allowed removal of the main board but also the connectors from the rear panel. This made it easier to re-solder and sleeve the connections. The 0v connections were previously daisy chained between the three connectors, however I modified that by sending the connections to the 0V power supply PCB area on 0.9mm connectors.



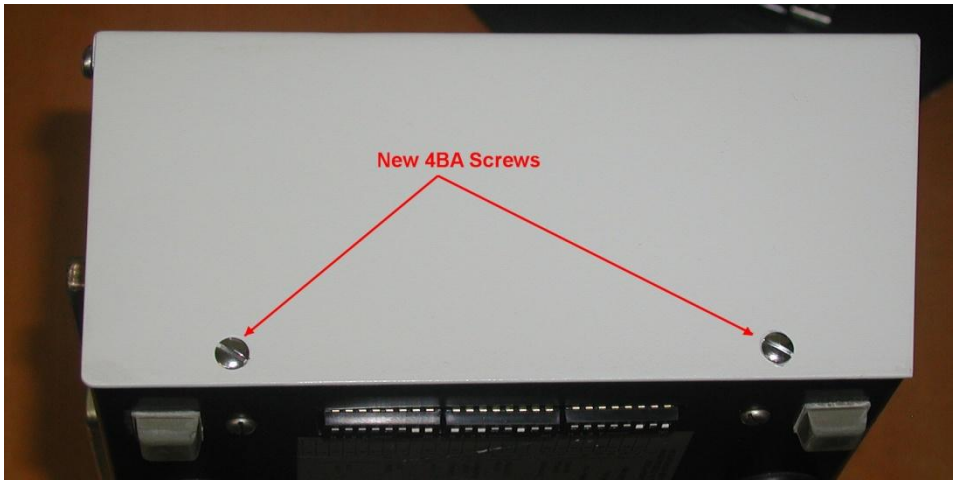
The above modifications make it much easier to service the unit and remove the PCB’s as required for repairs, or study.

As noted there were no markings of any kind on the rear panel. When I received the unit I had no idea about the connectors, so labels were added. I made them with a Brother Label machine:



As mentioned the original case side fixing screws were missing and replaced by someone with self threading screws. These screws thread into a 4BA threaded nut located inside the unit. On the other hand the three screws along the rear upper edge of the enclosure are self tapping types and screw into a thin steel panel. I replaced those with Stainless steel versions.

I was able to obtain original ¼ inch long 4BA screws. Chrome plated, countersunk, domed & slot head types from a UK eBay supplier, which give a very tidy look:



9) SUMMARY:

The Taylor –Wilson unit is a remarkable printer interface suited for use with the PET computer and designed to help deal with PETSCII. This unit is far more sophisticated than other PET printer interfaces I could find. The serial output is also very convenient for sending data out to a modern computer running a terminal program such as Tera-Term.

Taylor-Wilson Systems LTD remains somewhat of a company enigma. I can only find very limited references to them. Clearly though they were GPIB interfacing experts. The design of this unit sets a good example of how, using TTL logic techniques, to design a GPIB receiving interface which:

 # Recognises a primary address.

 # Recognises a secondary address.

 # Can Handshake with the GPIB bus.

 # How a secondary address can be used as software control mechanism over hardware functions.

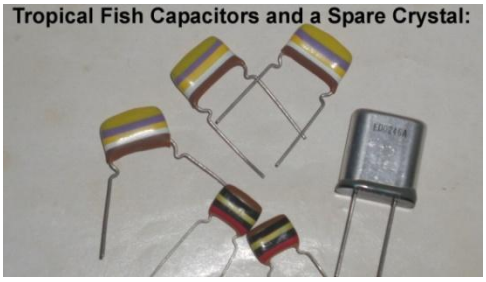
 # How to Latch the 8 bit data for use in a parallel or serial system.

 # How to add a UART for a serial system.

 # How to control the wait state to avoid data overflows in both serial & parallel systems when data is being transferred from the GPIB bus to a peripheral receiving unit.

Reverse engineering the T-W unit has been a very helpful education for me, learning about the GPIB and its properties. With the information provided in this article the T-W unit can be fully cloned. Also it can provide the basis for many home grown PET- GPIB interface projects.

When it comes to making replicas, if I do, I do it with identical or period correct parts where possible. It pays to have these on hand too, in case repairs are required for the original unit. I currently have all the IC's in stock as spares for the T-W unit, except for the interesting quad NAND gate buffer, the 7437. I have those on order now.



****