

A Computer Driven Train Controller

Article 6 – Now its time to test the train controller

Welcome back to this series which has been looking at the theory, design and construction of a computer controlled train set. Readers who have been with us from the beginning will have spent the last few months mulling over the circuit diagrams and perhaps have even commenced or completed construction.

For those who have come this far and completed construction, I'm sure you are becoming more and more eager to see your trains magically circulating about your layout without human intervention. Be patient! There is still a little more work to perform before we can play with our trains!

This month, we'll look at a step by step method for testing the circuit so that you can quickly identify any problems and hopefully, find the source of the error without too much detective work.

Testing the train controller and PC interface

Once you have completely wired the train circuit and are ready to apply power for the first time, I suggest that you take heed and plan to follow a systematic process of testing. In a scratch built circuit with so many different components as this, the chances are overwhelming that you have made at least one simple error and you should be prepared to discover and correct every error before the train controller can be connected to your layout.

Most importantly, there are some types of wiring errors which may cause damage to components or to the card itself if they are not discovered first and these need to be sought out and repaired in the least destructive manner possible. With care, no wiring error need cause damage nor destroy a part. Don't be daunted, it's really not that hard to discover wiring errors when you keep your wits about you.

This systematic approach to testing should be almost universal for home built circuits, particularly for those circuits which you've designed yourself and never before confirmed that the design works. So the principles we'll apply here will find application in almost any home construction project you should care to attempt.

So what are we waiting for?

As you commence testing, it is best if your train controller card is lying on the bench and there are no external connections yet made. All pluggable ICs should still be *out* and the point motor solenoid drivers (which should normally be mounted on or near the point motor solenoids themselves) should be disconnected. At this stage, even the power supply should be disconnected from the train controller card.

Now, grab a copy of the circuit diagram and take a deep breath.

The first step is to verify that power wiring is unshorted and has been routed to the correct pins of each IC. Take a multimeter and check that there are no dead shorts between any combination of the power input lines (-12, 0, +5, +12, +24 Volt). You will notice that some combinations have a low resistance and some have a higher resistance. You will even notice that for some combinations, the resistance appears to vary or may appear negative! The variance and negative resistances are because of the capacitors across the power supply rails which slowly charge when the multimeter

probing voltage is placed across them. If there are any pairs of power supply inputs which appear to be a steady dead short, you'll need to trace both of these power rails through the circuit and remove the short before proceeding.

Now use an ohmmeter on the x1 scale to confirm that the power input lines are connected to the right ICs and the right pins. For example, verify that the -12 Volt power input is connected to pins 11 of IC17, 1 of IC14, and 11 of IC19. It is also good practice to, for each separate power input line, quickly slide the ohmmeter probe around *every* other exposed IC socket pin (for every IC in the circuit) and make sure that there are no dead shorts to any place except to the pins where the power rail is supposed to be connected.

It is now time to carefully start connecting power but before you jump in, remember one very important thing: from this point on, if you discover or even think you have discovered a problem, you should immediately switch off the power and only then think about what to do next. Also, on no account should you connect a multimeter switched to an Ohm scale to the circuit whilst power is applied because you'll run the risk of damaging your meter. Remember, "when in doubt, remove power, then try to work it out"! Before proceeding, think about the fastest way to kill the power and remember it.

For the time being, just connect the 0 Volt, +5, +12 and -12 Volt power supply wires and apply power. Use a voltmeter to check the voltages on the pins of IC17 to ensure that only pins 2, 4 and 11 have a voltage. Pin 2 should read around 2.25 Volt, the other two pins should read the correct power supply level. Now turn off the power supply and wait 20 seconds for it to discharge before installing IC17.

When you again turn on the power supply, you should be able to use a frequency counter to measure a 45Hz triangle wave at pin 14 of IC17. If you do not have a frequency counter, an analogue multimeter on a DC Volt setting should usually read about 2.5 Volt on the same pin. As a last resort, you could use a digital logic probe to check for an oscillation. If you measured what seems to be a periodic signal, tap yourself on the back and stop for a break, the PWM frequency source appears to be working.

The next step will be to add the +24 Volt power supply input to the already connected 0 Volt, +5, +12 and -12 Volt lines. If all is still well, you should be able to read +18V on pin 4 of the empty IC19 socket and hopefully, verify the +16V75 and +8 Volt regulators are regulating to the correct values. It is OK if the regulated voltages measure just a little high or low. This is normal and due to variations between the nominal and actual value of the regulation resistors and ICs.

If you have measured the wrong voltage for one or more of the regulators, check that you have installed the correct value resistances to the Adj. pin and check that the regulator in question is not getting hot or warm. If the regulator starts dissipating heat, this is a sign that you've inadvertently swapped two or more of the regulator pins or that you've got a short somewhere on the regulated voltage rail.

If neither the +16V75 nor +8 Volt regulators are outputting any voltage, the problem is likely to be in the current trip circuit. Is the current trip LED illuminated? If so, momentarily press the Power Overload Reset Switch, verify that the LED turns off and that the regulators output the correct voltages. If the LED turns off but turns on again when you release the switch, or if the +16V75 and +8V regulators are still at 0 Volt, try turning VR1 to a mid range value (which changes the cut out current). If the +16V75 and +8V regulators are still not operational, you may have a wiring problem in

the current trip circuit or you may have inadvertently installed one or more of the transistors incorrectly.

Now is the time to verify that the current trip circuit actually trips when the current gets too high. Before deliberately shorting the +16V75 or +8V rails, adjust VR1 so that the wiper leg is at the anode of D5 side of its travel (i.e. not shorted to R52). Remember to switch off power and wait for the power supply to discharge before using an ohmmeter on the circuit. Now, when you deliberately short either of the protected rails (+16V75 or +8V), the LED should quickly illuminate and the regulators should be shut off. Verify that the *Cut Out Tripped voltage at pin 5 of the empty IC11 socket is less than 1 Volt. Pressing the Power Overload Reset Switch should restore power and extinguish the LED. Now verify that the voltage at pin 5 of the IC11 socket is +5 Volt.

The final part of the power supply testing is to confirm the correct operation of the Points Supply. With the power supply turned on and nothing connected to the points supply, you should notice that LED2 is turned on and the voltage at the Points Supply output should be about 21 to 22 Volt. Assuming that you followed my advice and have chosen not to mount C16 on the main train controller card, you should be able to connect an ammeter between the Points Supply output and the Points 0 Volt output and measure a current of around 0.6 Amp.

Caution! Make sure your multimeter is rated to measure currents of this magnitude before you try. In particular, don't attempt this when C16 is connected or you will blow your multimeter's overcurrent protection fuse.

When you measure this current, you should notice that LED2 turns off at the moment that you connect the meter and turns back on as soon as you remove it.

With the meter removed, temporarily connect C16 between the Points Supply and Points 0 Volt outputs. Caution! Electrolytic capacitors will be irreparably damaged if a voltage is applied in reverse, even momentarily. Damaged capacitors may sometimes appear to continue to work but could subsequently explode and hurt somebody.

Now, if you take a point motor and discharge it from C16 (the point motor solenoid driver circuit described in the last article need not be present) you should be greeted by a resounding "Thwackkk" and a great model railroad feeling of satisfaction! For as long as you hold the point motor onto C16 and the Points Supply output, LED2 should remain extinguished and after a little time, you should notice that IC24's heat sink becomes hot and R58 and the point motor coil becomes warm but not too hot to touch. The regulator IC will be dissipating more than 10 Watt when the point motor coil is connected and if it gets too hot too quickly, you may need to think about using a larger heat sink.

When you're sure that the heat sink is adequate and that the Points Supply is doing what you expect, disconnect the point motor coil and LED2 should again slowly ramp to full illumination within a second or so. The reason that LED2 takes a little time to illuminate now but illuminated quickly before is that with the capacitor installed, the voltage rises more slowly than with the capacitor absent. This is normal and to be expected.

Now that you've progressed this far and eliminated any wiring or component orientation problems, congratulate yourself again. You have probably gained some confidence that you can indeed get the circuit working. In fact, now that the power supplies have been verified, you have reached a stage where you can progress at a more rapid pace.

The next step will be to check the operation of the block detectors and track feed wiring. The test needs to be repeated ten times, once for each of the ten track power feed outputs.

Use a voltmeter to measure the voltage across each of the two track power outputs. The polarity orientation of each pair of track power feeds need not be identical so it doesn't really matter if you haven't been consistent for all of the ten outputs. For each pair, the voltmeter should read close to 18 Volt. If not, you have a problem with the wiring of RL111- RL120 or a problem with the wiring of R101 – R120.

Using the voltmeter, measure the voltage at the corresponding *Block #n Detect pin on IC9 or IC11. The voltage at the ICs should read about +5 Volt. Take a high valued resistor, say 10k, and bridge the two track power outputs. Verify that the *Block #n Detect voltage immediately changes to approximately 0.2 Volt. Remove the resistor from the track power outputs and verify that the *Block #n Detect voltage immediately returns to +5 Volt. If you failed to see the right voltages at IC9 or IC11, you have a problem in the wiring of the associated block detector circuit. Did you install the right valued resistors for R101 – R150? Did you orient transistors T101 – T120 correctly? Is there a wiring error or open circuit? Assuming that everything looks OK, the most likely cause of fault is a hidden short circuit because this part of the circuit is dense and there are lots of components connected across and around each other.

Next, remove power, wait for the power supply to discharge and install the RS-232 level conversion ICs, IC14 and IC15. When you reapply power, verify that the RS-232 output pins in the East and West connectors are either at +12 Volt or –12 Volt. Also verify that the voltages at pins 12 and 14 of the empty IC9 socket and 11 and 14 of the empty IC20 socket are either 0 Volt or +5 Volt. Whilst you are verifying voltages, confirm that all pins of ICs 9, 11, 16, 18 and 20 either read 0 Volt or +5 Volt and no other value.

Using a low valued resistor (approximately 100 Ohm), systematically connect each of the RS-232 inputs in the East and West connectors to either +12 Volt or –12 Volt and confirm that the corresponding RS-232 output also changes on the West and East connectors to the same level. Any errors checking the RS-232 voltages or voltages on ICs 9, 11 and 14 above will most likely be due to wiring mistakes.

Assuming all is still good, remove power, wait 20 seconds for the power supply to discharge and insert ICs 7, 8, 9, 10, 11, 12, 13, 16, 18, 20 and 21. When you reapply power, you should verify that the voltage at pins 2, 6, 10 and 12 of the empty IC19 socket are somewhere between 0 Volt and +5 Volt, that the 45 Hz waveform (or an apparent 2.25 Volt DC analogue multimeter measurement) is at pins 3 and 5 of IC19 and that the voltage at pins 9 and 13 of IC19 are 0 Volt (but it's OK for the time being for this particular reading to be anything between 0 Volt and 18 Volt).

When IC21, the reset device, is first powered up, the voltages at pins 1 to 7 of ICs 1 to 6 should be either 0 Volt or +5 Volt. If this is not the case, very briefly short pins 14 and 15 of IC21 and quickly remeasure.

The voltage at pins 9 to 16 of ICs 1, 3 and 5 should be +24 Volt (except pin 10 of IC3). The voltage at pin 9 of ICs 2, 4 and 6 should be +12 Volt and the voltages at pins 10 to 16 of ICs 2, 4 and 6 should be approximately 0 Volt. Remember that if you followed my suggestion, the orientation of ICs 1 to 6 may be the reverse of the other ICs. Again, any errors in measured voltages will probably be due to wiring mistakes. Errors at pins 10 to 16 of ICs 1, 3 and 5 will be likely to be relay coil wiring errors.

After removing power and waiting for the power supply to discharge, insert ICs 1 to 6 (carefully observing the orientation) and IC19. All IC sockets should now have an IC in them but if you're timid, insert just one of the ICs 1 to 6, just to be doubly certain not to destroy all at the same time.

When power is reapplied, do not be surprised if there is a loud "click" as one or all of the relays flip on. The relay driver ICs 1, 3, 5 should remain relatively cool. If not, you've either purchased relays with too low a coil resistance or you've made a wiring / component orientation mistake.

Without a PC yet connected, it is difficult to confirm that the two track power controllers are functional. For the time being, you can only approximately confirm their operation. If the D/A converters are outputting zero Volt, you can still confirm the operation of the circuits by temporarily shorting the op amp side of C1 – C4 to +5V before performing the following measurements.

The programmable DC power supplies can be checked by first observing the voltage at pins 10 and 12 of IC19. Unless you use a digital multimeter, you can expect your measurement to be partially in error due to the relatively low impedance of the meter. The corresponding voltages at pins 9 and 13 should be identical to the voltages at pins 10 and 12 (but not necessarily with each other). Also, the voltage on the emitters of T3 and T6 should be approximately 1.8 times the voltages on pins 10 and 12. If not, there is likely to be a wiring error around IC19, the orientation of T2, T3 and T5, T6 or the values of R45, R46 R48 and R49.

Now measure the voltage at pins 2 and 6 of IC19. Taken as a proportion of 5 Volt and specifically using an analogue voltmeter, you should read the same proportion of 8 Volt at the drains of T1 and T4. Again, a problem here most likely indicates a wiring error around IC19 or transistors T1 and T4. Did you check the orientation of the transistors?

Use an analogue multimeter to measure the convoluted voltage of each of the variable power supplies at the junction of D1 and D2 plus at the junction of D3 and D4. You may need to re-read the first article to refresh your memory about mixing DC and PWM voltages. Remember, if the variable DC supply is outputting more than about 8 Volt, then the convoluted value will also be pure DC. Otherwise, the convoluted value will be a mix of DC and PWM.

The last of the measurements needs to be repeated ten times, once for each of the track section power feeds. Take a low valued resistor, (say 100 Ohm 2 Watt) and connect it systematically to each of the track power feed outputs in turn. You should be able to confirm that each of the outputs corresponds to one of the two convoluted voltages you measured above. If not, you have a wiring error in the area of Relays RL101 – RL110. Note that if the convoluted voltage is above 6 Volt (such as if you shorted the op amp side of C1 - C4 to +5), you might feel your resistor becoming warm or even hot! More than one watt will be dissipated if the convoluted power supply voltage is above 10 Volt.

As you perform this measurement, be wary that if the current drawn rises too much, the power supply cut out may trip and illuminate the LED. If so, vary the setting of VR1 until it is no longer a problem.

Checking the parallel port adapter circuit

Now that the stand alone electrical aspects of the train controller have been verified as much as possible, it is time to get the parallel port adapter circuit verified. As with the train controller card, start testing with the pluggable ICs removed and the circuit disconnected from the PC.

Commence checking the parallel port adapter card by using an ohmmeter to confirm the power supply wiring. Referring to the circuit diagram, the +12 Volt, -12 Volt and 0 Volt pins in the East connector should be checked to ensure that they are not shorted together and then for continuity to pins 14, 1 and 7 of IC301, pin 7 of IC300 and to the regulator circuit. You should also ensure that there is no connection to the +12 Volt and -12 Volt pins in the West data connector and no inadvertent connections to the power inputs at any of the remaining IC pins.

If you have a split rail bench power supply, connect it to the power inputs on the East data connector, turn it on and ramp the voltage slowly up towards the nominal 12 Volt. If you don't have a split power supply, use the power supply you intend to use to power the train controller card or alternately, a pair of 9 Volt batteries. As you apply power, there should be no noteworthy event and the components should remain cool.

Using a voltmeter, verify that the output from the +5 Volt regulator remains off. If it is on, you are likely to have a problem in the wiring of the regulator circuit. Check T300 and T301. With the PC disconnected from the parallel port adapter circuit, the Base of T300 should be at 0 Volt, the Collector of T300 and Gate of T301 should be at +12 Volt and the Drain of T301 and input of IC302 should be at 0 Volt.

Temporarily short the PC side of R311 to +12 Volt and confirm that the regulator is now outputting +5 Volt. Make sure that you short the PC side of R311 to the power rail rather than the T300 side because you will certainly destroy T300 if you get it wrong. Having confirmed the presence of +5 Volt, check that this appears on the correct pins of IC300 and nowhere else.

You may now turn off the power, wait a short time and plug in ICs 300 and 301. When you reapply power (and short the PC side of R311 to the +12 Volt line), confirm that there are only TTL voltages present on the PC interface. Remove the R311 short and using an analogue voltmeter, confirm that all PC interface pins are floating (or have returned to 0 Volt).

Its time for Linux

Again, remove the power source and wait a short time. It is now time to connect the parallel port adapter to a PC and to boot Linux.

From now on, you should be able to independently apply and remove power to the PC and the parallel port interface card. When you power up the PC, the LEDs on the parallel port adapter could be in either state and the state of any of the parallel port data pins will be undefined – other than that the voltages at any pin on the parallel port connector should be within the TTL voltage range.

I have written the train controller test programme and the example train controller code in the C language and for the Linux operating system. Assuming that you have Linux up and running, you should obtain a copy of the archived test and example source files at <ftp://xxx.xxx.xxx/xxx/train.tgz> and place it into its own directory. From this point on, I will assume that you are familiar with Linux, have installed Linux correctly on your PC or that you know somebody who is familiar with Linux who can teach you.

Because the parallel port is a privileged resource under the Linux operating system, only the root user is normally able to access it. This means that all programmes needing to access the parallel port need to be executed by the root user or the executable file should be owned by root and have its Set UID flag enabled. If this is gobbledegook to you, you will probably want to think about investing a

book about Linux or hitting the web and immersing yourself in the Linux sites. But for the time being, just log in as root when you want to compile your train set.

Ensure that you have changed directory into the place where you have the archived train set source files and type the following:

```
tar xzpvf train.tgz
```

All of the individual source files will now be created in the directory and you will be ready to compile. In order to compile the test_ring programme, type the following:

```
make test_ring
```

Your PC should whirr into action and display something like the following dialogue:

```
gcc -O2 -Wall test_ring.c -o test_ring
chown root test_ring
chgrp root test_ring
chmod 4755 test_ring
```

You've got some thinking to do if something went wrong in either of the above steps. The problem is likely to be one of the following: Linux and DOS / Windows use different conventions for CR/LF and some tools automatically translate files as you copy them from one to the other. This would mangle the .tgz file. It is also possible that you did not type "make" from the same directory which contained the Makefile and test_ring.c files or that you did not log in as root or that your Linux installation is not configured correctly.

Now that you've compiled it, you can type:

```
test_ring
```

and after a second or two, you should be greeted with the following response:

```
Latch does not appear to be looped - 0 not passed (status port reads 0x4f)
Clock does not appear to be looped - 0 not passed (status port reads 0x4f)
```

There is a problem with the ring....

1. Is the parallel port adapter connected to the PC?
2. Is the port it's connected to located at address 0x378?
3. Are the ring cables connected correctly (orientation) and in a ring?
4. Are all of the train controller cards in the ring powered ON?
5. The PC interface or one of the train controller cards may be broken.

You will also see that the test programme then prompts you with the option to enter a diagnostic mode and you will need to press <CTRL>C in order quit. The diagnostic mode will be useful for you later on if you find that your circuit is not working.

If you saw the error message above, this is good – it tells you that your test software has the right root privileges and that you are ready to move on. If not, did you compile it as root?

Assuming that your parallel port adapter card is actually connected to the standard parallel port (at address 0x378), its time to loop the East and West connectors together. Figure 1 shows the method for doing this. If your parallel port has a different address (unlikely) then you will need to edit the test_ring.c file and change the address to something other than 0x378.

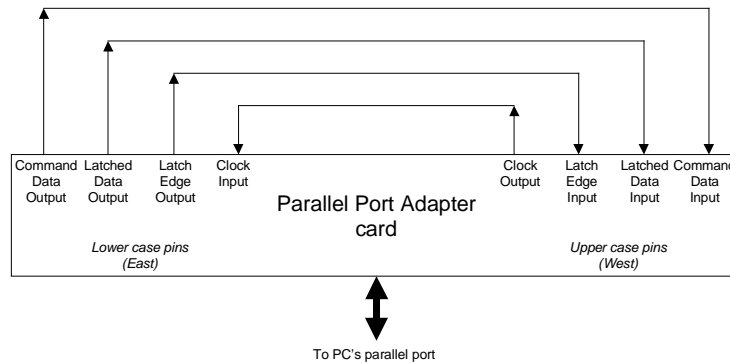


Figure 1 – Looping the parallel port adapter card back on itself

In effect, looping the connectors together means that you are creating a very tiny ring from the parallel port adapter directly back to itself. Because the parallel port adapter is normally powered through the ring cable from one of the train controllers, you will need to make a temporary powering arrangement while you've got the ring looped on itself. You can manually introduce power into the looped ring by connecting a +12 Volt and -12 Volt source or two 9V batteries as you did earlier.

Now you should see a different response when you run `test_ring` again:

```
The ring seems to be operational for the clock and latch lines.
Size of Command Data ring appears to be 0 bits
Size of Status Data ring appears to be 0 bits
```

If you remove the +/- 12 Volt sources from the parallel port adapter card and run the `test_ring` programme, you should once again see the error message advising you of possible causes.

Assuming that you've tested at least one train controller card to your satisfaction and it is ready to connect to the parallel port adapter, now is the time to do this. Remove any +12 Volt and -12 Volt power supply straps you may have made between a power source and the adapter card, remove the East to West loop back cable and cable a train controller card in their place. The East connector on the parallel port adapter should connect to the West connector on the train controller and the East connector on the train controller should connect to the West connector on the adapter card. Refer to Figure 2 (which shows multiple train controller cards looped in a ring) for an idea about how to do this.

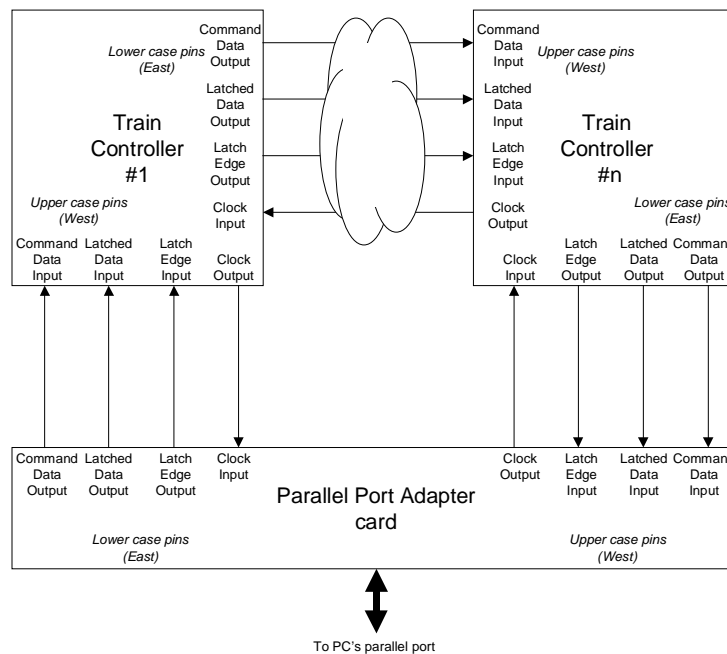


Figure 2 – Connecting multiple train controller cards to a parallel port adapter

Now when you power up the train controller, the parallel port adapter should receive power through the ring. When you run the test ring programme, the relays on the controller card may click and the response should now look like:

The ring seems to be operational for the clock and latch lines.
 Size of Command Data ring appears to be 56 bits
 Size of Status Data ring appears to be 16 bits

Congratulations, you're getting closer to being able to run some trains!

If test_ring reported a problem when you connected the train controller card, there are a number of possible causes including those causes suggested by test_ring's error message. There is also a possibility that the train controller card has a wiring error which is breaking the continuity of either one or both of the input and output rings. Remember that ring continuity could be broken either by an open circuit or a short circuit and the controller card problem could be either with the ring data wiring or with the ring clock and control wiring.

The best way to debug the train controller card and parallel port interface is to use a logic probe and to run test_ring. When the ring loop test fails, the test_ring programme will prompt you to run a diagnostic sequence. This sequence will start to generate a series of very slow square waveforms on each of the parallel port adapter's ring outputs and at the same time, read each of the ring inputs and display the result.

Interpreting the diagnostic information will be an exercise in logic. Carefully study the information dump which is displayed and it should (eventually) become apparent whether two of the logic signals have been crossed, or some signals are open or even shorted. Remember that the clock and control signals propagate around the ring instantaneously and that the data signals propagate slowly (as a function of clock and control edges).

Then, use the logic probe to trace the ring signals from the PC, through the parallel port adapter card and if still OK, device by device through the suspect train controller card.

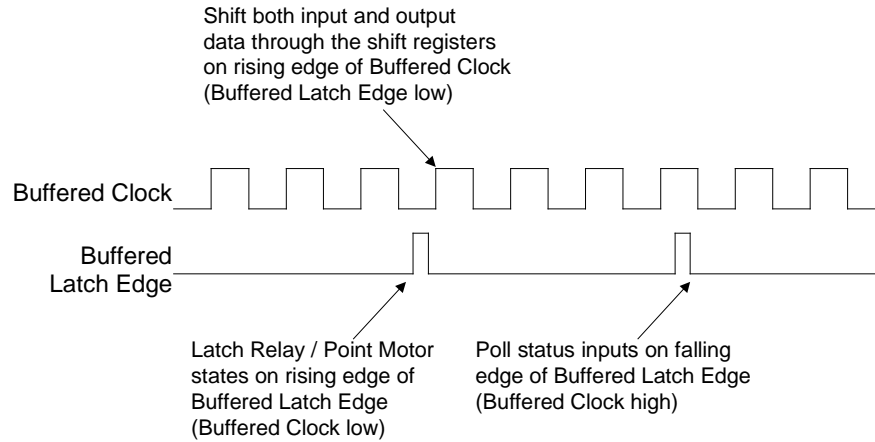


Figure 3 – Ring control waveforms as seen after conversion to TTL

Figure 3 shows the relationship between the ring control waveforms after their conversion to TTL. That is, the polarity of the voltages shown in Figure 3 are the same as would be measured at the TTL sides of the 1488 and 1489 RS-232 driver ICs. Note that the sequence shown in Figure 3 is typical of a working system and the probing part of `test_ring.c`. But the slow diagnostic waveforms generated by `test_ring.c` do not follow this specific pattern but rather, generate a four bit ascending binary sequence (i.e. 0000, 0001, 0010, ..., 1111) across the four ring control signals.

As shown in Figure 3, the Buffered Latch Edge signal is normally low and in this state, data bits are shifted into and out from the serial rings on each rising edge of the Buffered Clock signal. While the Buffered Latch Edge remains low, data will be shifted through the 74HC595s and 74HC597s but will not be latched to the output and input registers of these devices.

In order for the data to be latched into the 74HC595's and 74HC597's registers, the Buffered Latch Edge needs to be changed to a high state. The logic in the train controller card uses the polarity of the Buffered Clock line to determine whether the ring data will be latched into the 74HC595 (Buffered Clock low when Latch Edge strobed) or whether the input data will be polled from the input pins of the 74HC597 and written into the input ring (Buffered Clock high when Latch Edge strobed).

If you study the source code in `test_ring.c`, you will (eventually) see how these waveforms are synthesised by the software. This sort of operation is often referred to as "Bit Bashing" and the software to do it is usually ugly and difficult to follow unless (and maybe even if) you have the benefit of a diagram such as Figure 3.

Testing the distributed point motor coil drivers

There is one last part of the circuit which has not yet been tested – all of those transistors and resistors which you've mounted underneath your train set for driving the point motor solenoids.

The easiest way to test these is to connect a 12 to 24 Volt power source with at least a few Amp capacity to the Points Supply bus under the layout. Manually ground the *Point n Set / Reset connection of each instance of the driver circuit and it should be obvious whether the circuit is working or not by the thumping of the solenoids. Just be careful to ensure that you don't fire any solenoid for more than a second or two because when you are using a 24 Volt source rather than the regulated Points Supply on a train controller card, there is no current limiting protection. If you hold the solenoid on for too long, it will get hot and after a while could destroy itself and even burn the train set.

As you're testing your point motor drivers, also test the points themselves to make sure that they reliably change from one state to the other each and every time you enable the solenoid. If your point motors do not reliably move the points *all* the time, you're going to have to find the mechanical problem and fix it. The train controller software assumes that the points are reliably positioned and you'll witness some interesting crashes, derailments or controller card overcurrent trips if one of your points is marginal and doesn't change reliably.

Connecting the train controller to a train set

At this stage, you should be as confident as possible that the train controller card is operating correctly and that the point motor drivers which are scattered around your layout are also functional. It is time to connect everything together into its final configuration.

When it comes to mounting the circuit, it is by far the most practical to dispense with any thoughts of building the circuit into an aesthetic box and displaying the box proudly to all your friends. The reason you should be humble is because model trains and their peripherals require (relatively) high currents and to avoid the problems of resistive and inductive losses, the circuit needs to be located as close as practical to the train layout itself. In my case, I have permanently mounted the cards beneath the scenery in my layout (actually, the scenery isn't quite there yet) and the longest conductor length to connect them to a part of the train set is well less than 2 metre. So if you wish to show your creation off to your friends, you'll have to move mountains or tell them to stoop down low and look at the underside of your table.

Immediately, it becomes apparent that it is also impractical to build the train controller circuit into a computer. Not only would the nasty voltage and current spikes be a hazard for your (relatively) delicate computer hardware, if you keep all wires to shorter than 2 metres as advised, you may need to stoop below the layout to get to the keyboard and screen. An arrangement like this would make it difficult to operate the computer and watch the trains at the same time but I guess that you could always be the first!

The design specifically uses RS-232 level ring voltages to allow you to make rings that are virtually as long as you like. Normally, the computer can be located somewhere accessible and nearby to the train set and you can choose the neatest ring cable path. There is no real need to try to minimise the length of the ring cabling.

When permanently wiring the train controller cards to your train set, be careful to avoid making "one turn transformers" with any of the under-layout wiring: a "one turn transformer" happens each time a specific wire "cuts through" the magnetic field created by a pair of current carrying wires. Such

accidental transformers effectively couple voltage spikes from one pair of wires to another and are difficult to recognise and discover. The easiest way of avoiding "one turn transformers" is to either obtain twin conductor cable or to twist each pair of track block wires along the entire run between the train controller card and the track block and don't let any other wires pass between them. If you permanently wire all of the track blocks before starting the points wiring or the stand alone mounted C16, then you will likely avoid inadvertent "transformer coupling" between the block and points wiring too.

If you've managed to complete all of the tests that are described in this article, you should be satisfied that your PC interface and controller cards are basically operational and can communicate. However, at this stage, we have not tested the functioning of the power controllers, the relays and point motors. Detailed tests for these functions are built into the train controller software and will be described later in the series.

With those comments, its time to round off the sixth article and leave you to your testing. In the next article, we'll look at the example software in a lot more detail and explain how to configure and compile it to suit your particular layout. So, plan your time this month and we'll catch up again in the next issue.

Stefan Keller-Tuberg BE (Hons) MEM

Stefan Keller-Tuberg is a professional engineer with an honours degree in Electrical Engineering from the University of New South Wales and a masters degree in Engineering Management from the University of Technology, Sydney. He and his family are presently living in the United States where Stefan works in the telecommunications industry planning service architectures and product evolutions for high speed xDSL and Optical Internet access products. They are enjoying life in the USA but are looking forward to their return to Australia some time in the coming year.

Copyright 2001, Stefan Keller-Tuberg.