

A Computer Driven Train Controller

Article 3 – The circuit for the controller

If we continue to do things in the conventional way, the future will be conventional. Well, perhaps in relation to model trains, this is overstating things a little - but if you've not yet had the chance to read the earlier articles, you'll now have a first impression about what this project is all about!

For the veterans, welcome back to this series of articles in which I am describing how I've designed, built and programmed a computer controller for my model train set. In the last months' articles, we looked at some of the fundamentals of model train layout wiring, designing a layout for automated or semi automated control, different control approaches and the operation of DC motors. This month, we look at the first half of the design and circuit for the train controller.

There are likely to be many different reasons model train hobbyists may have for building automated control into their layouts. For most, it will probably be a shared interest in both model trains and computers and for others, just trying to simplify the complexity of controlling multiple trains on an expanding layout. My interest in computer controlled trains started in my student days when I took a software subject dealing with computer operating systems. A major component of the assessment was a project involving a "train operating system" - software to control a model train layout. After having written train controller software successfully for somebody else's layout, I just couldn't settle down until I'd built a bigger and better version for myself.

In a conventional operating system, the software is required to manage the computer's resources which include memory, tasks, CPU time, peripherals and so on. The operating system makes sure that all things run smoothly and that the computer doesn't crash or hang when different tasks want to access the same resource at the same time. In my university train operating system, the resources were the track segments, points and power supplies and the numerous trains were the different "tasks" amongst which the resources needed to be shared. Of course, the train's "operating system" was written to run on a more formal computer operating system as is the software for this project.

The computer controlled circuit presented here is generic in the sense that it can be used with virtually any type of 12 Volt train set and with virtually any type of computer or microcontroller. In my case, I have used a PC running Linux to control my layout but most old PCs or even programmable microcontrollers will be sufficiently fast enough to manage even large train set layouts. For example, something like an old 6502, Z-80 or 8088 system with only the BASIC language would make a great first off train controller - that is if you specifically wanted to use the published circuit but write your own software. If you think you're creative, the electrical design of the circuit could even be adapted to control something other than a model train set.

Before you jump to conclusions about how complicated (or straightforward) such a train control programme might be, let me assure you that even if writing software is not your strongest skill, if you are able to install the Linux operating system onto your PC and recompile the kernel, it won't be that much harder to configure the example train controller software to work with your particular train layout. Of course, you are likely to eventually want to modify it to optimise it for your layout and once you start down this path, the possibilities are virtually endless.

From the controlling PC's perspective, the train controller cards are just peripherals with a swag of different parameters which need to be managed. Each set of points has a magnetic coil which needs to be activated in order to switch the points one way or the other. Each track block has two relays which need to be set or reset depending on which power supply is required and which direction the

train needs to move. Each power supply needs to be set to the appropriate speed and this speed needs to be changed to simulate inertia or to compensate for steep grades. Finally, each track block detector needs to be regularly polled so that the software can determine a train's location as it moves around the layout.

Because every hobbyist is likely to have completely different layout plans and because layouts tend to grow in size with time, the train controller card has been designed to be modular. To get started, all you need is a single train controller card and a single parallel port interface adapter and if you build the circuit presented here without modification, this will allow you to control two trains in a layout with up to ten blocks and ten sets of points.

As the layout grows, additional train controller cards can be cascaded from the first card so that additional track blocks or sets of points can be added or more than two trains can be controlled. My layout has provision for 30 electrical blocks, 20 sets of points and 6 programmable power sources and is likely to continue to grow as I run out of things to do!

The circuit of each train controller card looks like two large shift registers as shown in Figure 1. One shift register is used when the computer needs to set any of the parameters such as speed, points or relays and the other shift register is used when the computer needs to poll the block detection inputs.

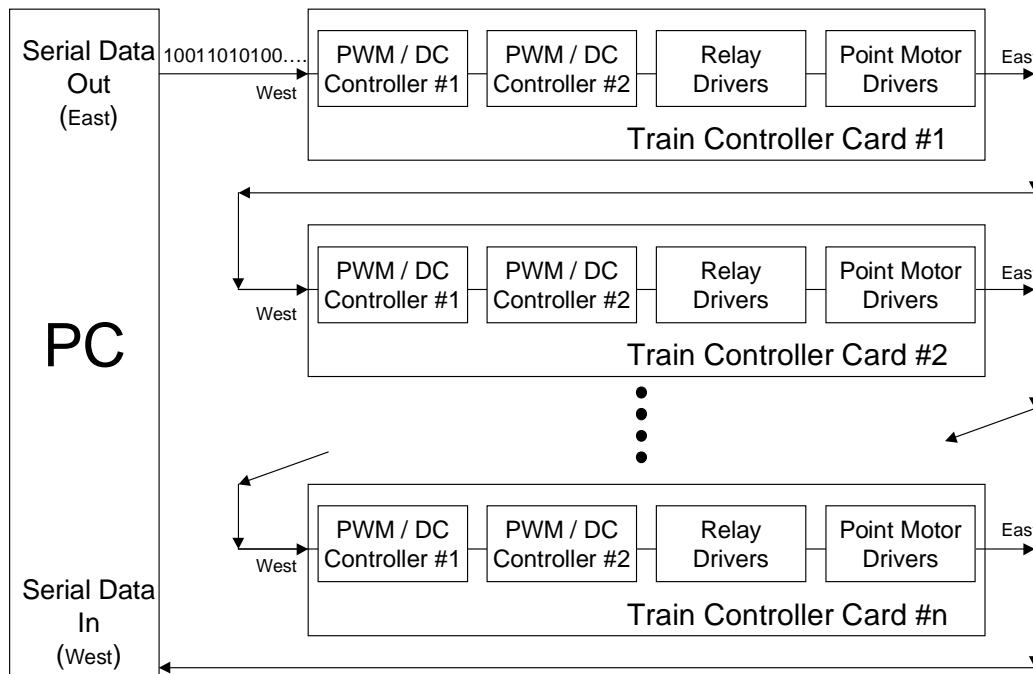


Figure 1 Connecting a group of train controllers into a single system

Because the cards are connected together as a ring, the PC must communicate with the cards one bit at a time. This is often called serial communication and is the same technique as used between computers and dial up modems or computers and LANs. Figure 1 shows only one of the two rings: the ring which is used for outputting information to the train controllers. The second ring, the input ring, looks similar but is wired as an independent circuit. The two rings follow the same path and can be operated in parallel or one at a time.

Each bit which the computer sends into the ring (or reads from it) represents the state of each parameter. In order to simplify the software, the computer treats the cascaded shift registers as a single very long string of bits. Practically, this means that the computer firsts shifts all bits into the ring and then issues the command to the controller cards to latch the bits into their registers. In the other direction, the computer tells all controller cards to sample their inputs before they are shifted into the computer and processed.

You might have noticed that because the communication path is a ring, the computer's output data eventually finds its way back towards the computer as input data. Why does the design include separate rings for the input and output functions?

In truth, both the input and output rings really could have been connected in series to form a single mega ring and electrically, the controller design would have worked perfectly. But there's another reason for having separated the rings into independent input and output functions. The answer to this question has to do with the important relationship between hardware and software: hardware design needs to be optimised so that software is simplified and efficient.

Under normal train set operation, the computer only needs to change the output settings infrequently (perhaps every few seconds). But in order to keep track of the location of all trains, the software must poll the controller's block detection inputs many times per second. Having a short input ring makes polling considerably faster and less microprocessor intensive than it would be if the input bits were muxed with the output bits.

Figure 1 also includes some strange terms such as "East" and "West". The meaning of these terms and the operation of the ring and its control signals are going to be discussed later in this article and in the following months. So keep on reading.

The Circuit

Each train controller card includes two independent power supply circuits which are shared, through a matrix of relays, amongst the ten block outputs. The two power supply circuits are actually programmable waveform generators and are shown in Figure 2.

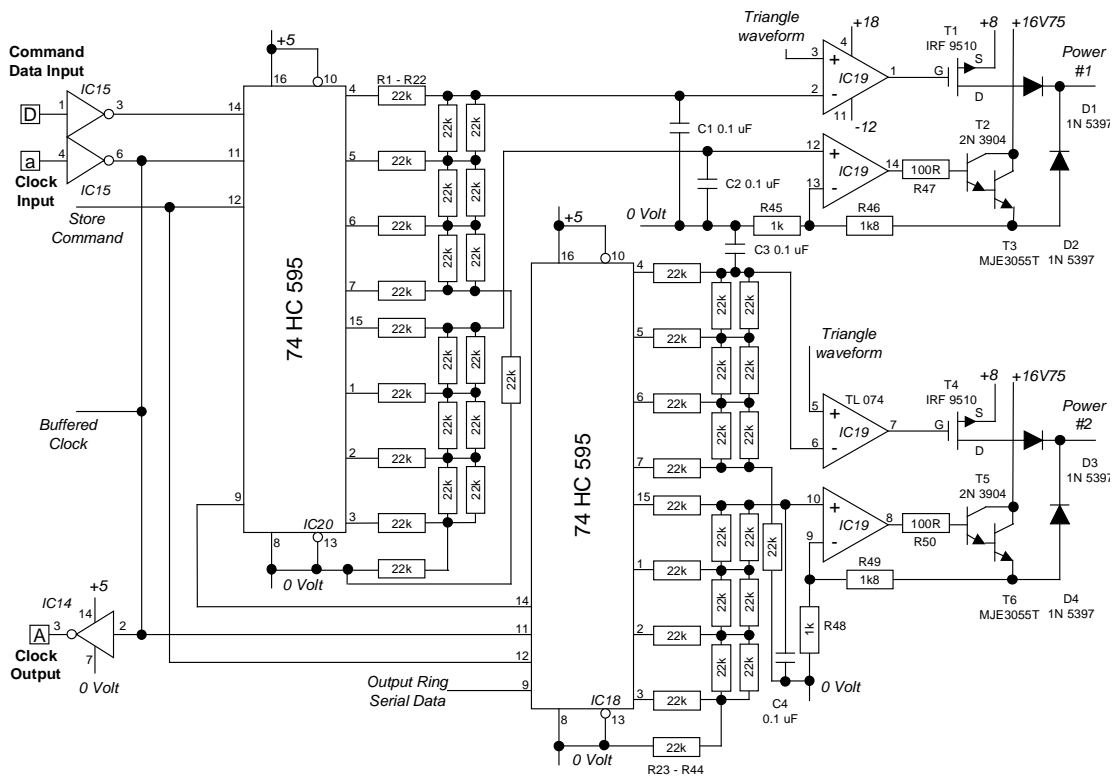


Figure 2 Dual waveform track power supplies

The two waveform generators each include a PWM (Pulse Width Modulation) generator and a pure DC regulator. The voltage outputs from these two generators are superimposed upon each other to produce the hybrid driving voltage which is delivered to the track.

This would also be a good time to explain that in the circuit diagrams in this article, explicitly writing “East” and “West” on each connector became too messy and another approach was needed. In the end, I settled on using different cases for East and West and the boxes with upper case letters in them represent pins in the West data connector and the boxes with the lower case letters in them represent the pins in the East data connectors.

But what do “East” and “West” mean? These are terms which are frequently used in the telecommunications industry to refer to different sides of a device which resides in a ring. But the East and West equipment ports could just as easily be referred to as “A” and “B” or “1” and “2” and there is certainly nothing special about the way you need to orient the equipment with respect to the compass!

By convention, when you connect different cards or devices together to form a ring, such as when there is more than one train controller card controlling the layout, the East connector of one card connects to the West connector of the subsequent card and so on.

As explained in the previous article, the PWM waveform is intended to be used during low to medium speed operation and so doesn't need to deliver the full 12 Volt which would be required for nominal full speed. The peak PWM level is supplied from an 8 Volt source, which after traversing some diodes (D1, D3, D101 - D110), is delivered to the track as a 6 Volt peak signal.

The PWM frequency has been designed to be approximately 45 Hz which is low enough to be below the practical audible hearing range and low enough not to be affected by the small inductance present in the track wiring. As a comparison of this frequency with the frequencies implemented in typical on-train DCC decoders, most DCC decoders operate their PWM in the range 60 Hz to 70 Hz with the higher end programmable decoders supporting frequencies in the wider range of 40 Hz to 100 Hz. If it ever becomes necessary, it is simple to vary the PWM frequency away from the default 45 Hz and there are some notes on how to achieve this in the following article.

The DC component of the voltage waveform is intended for medium to high speed operation and is variable from 0 Volt right through to about 14 Volt. After diode drops (D2, D4, D101 - D110) and resistive losses in the track wiring, the voltage which is presented to the locomotive can be varied by software between zero and about a 12 Volt maximum.

ICs 18 and 20 register and latch two four bit values each, representing the independent levels of PWM and DC power. The array of 22k resistors at the outputs of these ICs (R1 - R44) implement crude but surprisingly accurate and reproducible D to A converters. Eleven resistors are required for each of the four 4 bit D to A converters.

As described, the two D to A converters are each able to provide sixteen different output levels. In total, there are sixteen times sixteen (256) different combinations which could be configured, but only some of these are useful. Practically, when the voltages from the DC and PWM converters are convoluted, there will be a little more than twenty useable settings between 0 Volt and 7 Volt and around another five useable combinations between 7 Volt and 12 Volt.

The example software has been configured to provide 29 different useable settings; actually an off setting, 28 forward settings and an additional 28 reverse settings. I have found this to be ample but there may be some who would contest me, I'm sure. If you feel that more than 28 speed settings are desirable for your layout, then feel free to add additional bits to either the PWM or DC Digital to Analogue converters, or even to both. The example software can be relatively easily adapted to suit.

When the 22k resistors within each cluster of eleven are closely matched (meaning that having the specific individual values of resistance equal to 22k is far less important than having all of the resistors close in value), the voltage present at the top of the resistor ladder will vary in even steps. For the 74HC series of devices, the output voltage swing is virtually the full rail swing of 0 Volt to +5 Volt. If the resistors are not so well matched, the accuracy of the D to A voltage steps suffers a little, but for this particular application it is hardly worth the trouble to try to match the resistors.

The 0.1 uF capacitors at the head of each D to A ladder (C1 - C4) are to ensure that any small ripple in the +5 power rail of the 74HC595s is not directly transferred as noise on the analogue value.

The filtered analogue values are passed to IC19, a quad operational amplifier, to control the DC voltage and PWM conversions. The feedback from the two DC voltage converter outputs (IC19 pins 14 and 8) makes the op amps behave as amplifying buffers. T2, T3 and T5, T6 are configured as Darlington Pair voltage regulators and the feedback mechanism of the op amps will at all times try to ensure that the voltage at the emitter of these transistors will equal the multiplied D to A voltage values.

The analogue D to A output at the top of the resistor ladders varies between 0 Volt and 5 Volt but the DC power supply output needs to be able to get as high as approximately 14 Volt. So the D to A

voltage needs to be multiplied by 2.8 and this is achieved using the 1k and 1k8 resistors in the op amps' feedback path (R46, R47, R48, R49).

For the two PWM generators (IC19 pins 1 and 7), the op amp behaves as a comparator. A triangle waveform varying between approximately 0V and 5V is applied to the negative (inverting) input of the op amps and the PWM reference voltages from the D to A converter are applied to the non inverting inputs. When the triangle wave has a higher voltage than the PWM reference, the op amps output 12 Volt and when it is lower, the op amps output -12 Volt. If you looked at the waveform on an oscilloscope, it would appear to be a square wave of the same frequency as the triangle wave but as the reference voltage is changed by changing the D to A output, the duty cycle of the square wave also changes.

The P channel power MOSFETs (T1, T4) are used as current choppers to repeatedly connect and disconnect the 8 Volt supply in accordance with the PWM waveform. A conventional PNP power transistor such as an MJE2955T could be substituted for the MOSFET if desired but in order to generate sufficient base drive current for an MJE2955T, an intermediate transistor with associated resistors would also be required in a Darlington Pair configuration.

Finally, the rectifier diodes (D1 - D4) at the outputs of each power supply superimpose the analogue and PWM voltages onto each other so that the resulting waveforms are a hybrid of PWM over a DC floor. The specific identity of the diode is not so critical as long as its current carrying capacity is sufficient for the expected peak load of a few Amp.

Figure 3 shows the input latches and the shift register control circuit.

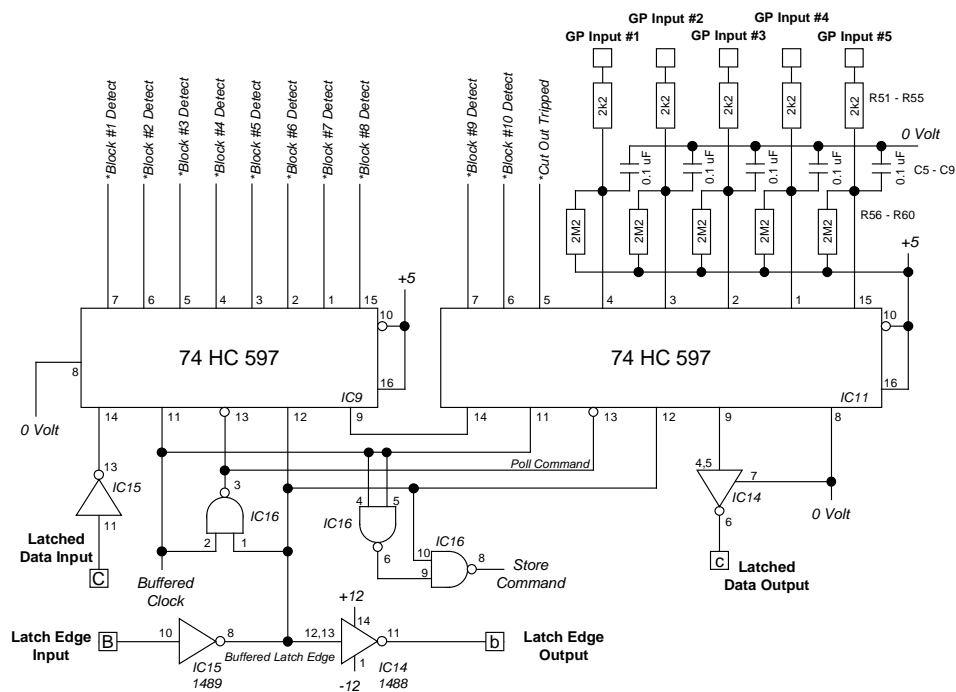


Figure 3 Input latches and interfacing logic

IC9 and IC10 are used as shifting input latches for gathering the status of the track block detection circuits, the overload trip circuit and 5 unassigned general purpose inputs.

In the case of the general purpose inputs, each pin has a 2M2 resistor and a 0.1 uF capacitor to provide some small amount of debouncing and tie up. However, in the case of each of the block detection inputs, no debouncing is provided. Rather than to debounce these inputs in hardware, I chose to save the components and let the software perform this task. We'll get into some more of the specific details about how the software works later in the series.

The operation of the latching and shifting is not complicated, but in order to save one control signal in the serial ring from the PC to the controller cards and back again, the procedure has been convoluted by IC16. IC16 takes two control signals from the PC (Clock Input and Latch Edge Input) and produces three internal signals (Buffered Clock, Store Command and Poll Command).

The voltage levels on the ring's serial signals follows RS-232 but the waveforms which drive the Clock Input and Latch Edge Input signals do not look like the waveforms that you would normally see on an RS-232 interface between (say) a modem and a PC. These waveforms will be described in one of the later articles but as a challenge, you might like to work them out for yourselves.

You will notice that as the RS-232 signals enter each train controller card at either the East or West ports, they are "repeated" to the matching RS-232 output interface on the West and East ports. Because the RS-232 signals are regenerated in this way as they pass through each card, the length of the ring and the number of cascaded cards in it will not be constrained by any practical limit. Regenerating the signals in this way also ensures that they will be far more immune from degradation and noise.

However, buffering and repeating the RS-232 signals as they pass from one card to the next will inevitably add some propagation delay and this delay will vary according to the number of train controller cards connected in series as well as of course the physical length of the ring, its capacitance, inductance etc. Because the propagation delay is variable, the software needs to have a simple method of determining how fast it can send data into the ring.

In order to solve this software problem, the design of the ring has been "closed" so that as the PC writes information into the ring, the software can simultaneously read the same information back from the other side of the ring. When the PC changes the voltage level on Latch Edge for example, it can immediately start polling the other end of the Latch Edge ring to determine when the change has propagated through all of the train controller cards and returned to the PC. When the change arrives back at the PC, the software knows that all cards have registered the RS-232 voltage transition because the transition must certainly have propagated through each card in order to have returned to the PC. The PC can then send the next transition and so on.

In this way, the data rate on the ring is truly asynchronous (or "variable bit rate") and typically, the peak throughput will be many hundreds of kilobits per second. Certainly, the time taken to read all input bits off the ring (or to send all output bits onto it) is far faster than an eye blink: you'll need access to a storage oscilloscope or you'll need to hack the software to slow it down if you want to be able to capture and look at these waveforms.

Now we come to a highly parallel part of the design, the power distribution and block detection circuit is shown in Figure 4.

(Note that this circuit is repeated 10 times,
once for each block)

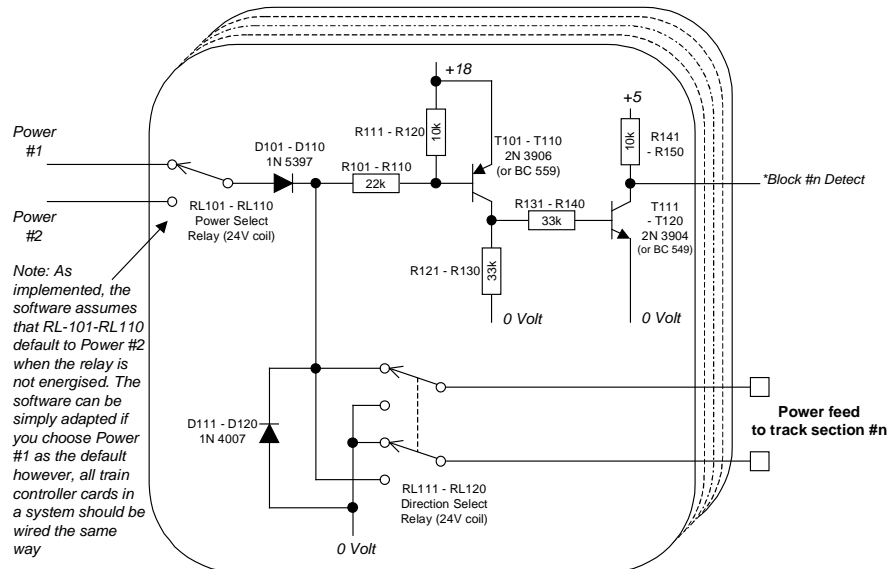


Figure 4 Block detection circuit and reversing relay

Having made the design decision to implement just two independent power supply circuits, the outputs from these power supplies must be shared between the ten track blocks. In sharing the power, each of the track blocks should remain electrically independent to allow polarity to be set independently and to prevent false block detection from trains in adjacent blocks.

A switching arrangement is therefore required to distribute the power to the many blocks and although it would have been possible to use solid state techniques for both polarity selection and ensuring block independence, simple relays seemed to be the most practical choice.

The operation of the power selection relays is straightforward. These relays (RL101 - RL110) select one of the two power supplies for each particular track block. There is no option to “not select” a power supply and so even if a track block is not assigned to a locomotive, power from one of the power supplies will still be present. This should not be a concern because it does not really matter if power is present on a track block if there is no train on it or about to enter it!

The power selection relays are immediately followed by diodes (D101 - D110). The diodes isolate each block detector circuit from the power supplies and the other detectors. After the diode, power is fed to the direction selection relay (RL111 - RL120) which selects the polarity for the track block it is serving.

Although some DC motors in locomotives predominantly present a resistive load to the power supply, in others a significant inductive component is also present. Inductive loads will always cause a reverse voltage “kick” in proportion to the rate of change of applied voltage and these sorts of spikes are potentially damaging to the surrounding components.

For a PWM power source, the voltage changes are very sharp and each time the PWM controller (or the DC controller) disables its current source, the current flowing through the motor's inductance will give rise to such a spike. Shunt diodes D111 - D120 provide the protective reverse current path to dissipate the energy before the magnitude of the impulse has the chance to wreak its havoc.

Finally, the block detection circuit comprises a handful of resistors (R101 - R150) and transistors (T101 - T120).

When this block detection circuit is compared with some of the many other published block detectors, it will become apparent that this one is amongst the simplest. The reason for its simplicity is as a result of being able to integrate the block detectors into the power supply prior to the polarity reversing relays so that voltage reversal need not be taken into account.

The block detection circuits work by "biasing" each track block to 18 Volt through a high enough impedance (22k via R101 - R120, T101 - T110) such that a small current will be able to flow into a train's DC motor but not nearly enough current as to make the motor turn. The bias voltage will always be present on the track block, regardless of which of the power supplies has been selected and how the power supply's voltage waveform has been configured. The bias level of 18 Volt was chosen so that it will always exceed the maximum voltage which the power supply is capable of delivering and so always keep the block detectors activated. In fact, the specific value of 18 Volt was chosen because it is required elsewhere and was therefore already "available".

When there is no train (or other load) on the track block, the open circuit voltage will read very close to 18 Volt (using a digital Voltmeter) and transistor T101 – T110 will be turned off. As soon as a train enters the track block, at least the electrical part of the train, a small current will immediately start to flow and will be detected and amplified by the transistors. If you measure this voltage across the tracks with an analogue Voltmeter, the chances are that the impedance of the Voltmeter will be low enough to trip the block detector and you'll get a false voltage reading.

The reason the block detection power source is larger than the highest normal track voltage becomes apparent if you consider what happens when a particular block is connected to a power source which is delivering the nominal maximum of 12 Volt. Even if the power controllers are already outputting a voltage waveform, let's assume that the track block does not yet contain a locomotive. Because the block detection reference voltage (18 Volt) is greater than the highest DC power voltage (12 Volt), D101 - D110 will be reverse biased and so the voltage presented to the track will be the higher of the two: the 18 Volt reference.

When a locomotive or other load enters the track block, a current flows and a voltage appears across R101 - R110. The load presented by the locomotive will almost appear to be a short circuit and the voltage drop across these resistors will easily exceed a few Volt. So diodes D101 - D110 will start to become forward biased and the locomotives will start to receive additional power from the "real" power supply. In order for at least some current to flow through R101 - R110 even when the power supply is outputting a full 12 Volt, the block detection reference voltage must be slightly higher. This ensures that when a train is in a block, a small amount of current will always be drawn through the block detector so as to bias T101 - T110 and to detect it.

The back end of the block detection circuit is based on T111 – T120 and the components around these. The detection signal at the collector of T101 – T110 is a logic voltage which is either +18 Volt or 0 Volt. In order to drive a HCMOS gate, a voltage between 0 Volt and +5 Volt is required. These transistors perform an inverting buffering function to translate the voltage levels. The ten *Block #n

Detect logic levels are then presented to the inputs of the input latches IC9 and IC11 as moderated voltages.

A criticism often made of this type of load sensing block detector is that the need for isolating diodes (D101 - D110) introduces a voltage loss and therefore introduces a dead spot around zero as the track voltage is swept from maximum positive through zero to maximum negative. However because this design is under the full control of software, PWM and DC setting combinations can be easily derived so that the diode voltage drop is taken completely into consideration. The dead spot is thus totally avoided by appropriate software controls.

The circuit, as drawn, specifies relays with 24 Volt coils. Relays with alternate coil voltages could also be substituted with the constraint that all relays be rated at the same coil voltage. The value of 24 Volt was chosen because in general, relays with higher coil voltages have higher coil resistances (Caution! Not always!) and the current they consume is often lower than for lower voltage relays. ICs 1-6 are discussed below but the power dissipated by them when all relays are activated needs to be kept within the chips' design limit. As long as each relay averages no more than 150 mA when turned on, the ICs' design limit will be respected.

When you take a look through the component list for this project, as well as noticing that there is a large total number of parts, you might also notice that the large total number comes mainly from the replicated block detection part of the circuit. As there are twenty relays per controller card and relays are generally priced in the range of dollars, a significant proportion of the cost will lie here. So if you are prepared to shop around for components, you may be able to save a considerable amount of money.

Don't overlook the value of "scrap" cards which can be obtained from a number of surplus suppliers or from regular suppliers' "bargain bins". The reason that such cards have been scrapped is rarely because their relays did not work and relays can be easily desoldered and used in the train controller circuit for nominal cost.

There are now many excellent Internet based mail order electronics shops, particularly in the USA and I was able to find surplus relays on the Internet for between \$US0.50 and \$US1 each. When buying internationally, a trap to be aware about is the cost of import duty and you should be prepared to pay a duty when it comes time to collect your delivery. It would be best to research duty rates before you place an international order.

No matter where you choose to shop, if you take the time to find two or three of the shops with the best prices, you will find that all of the components are simple to obtain and individually, relatively low cost.

The solenoids used in remote controlled model railroad points are often called "point motors". Figure 5 shows the relay and point motor output shift registers and drivers. In this part of the circuit diagram, the ring nature of the design becomes immediately apparent because each of the 74HC595s (IC7 - IC13) is effectively daisy chained into a ring through pins 9 and 14.

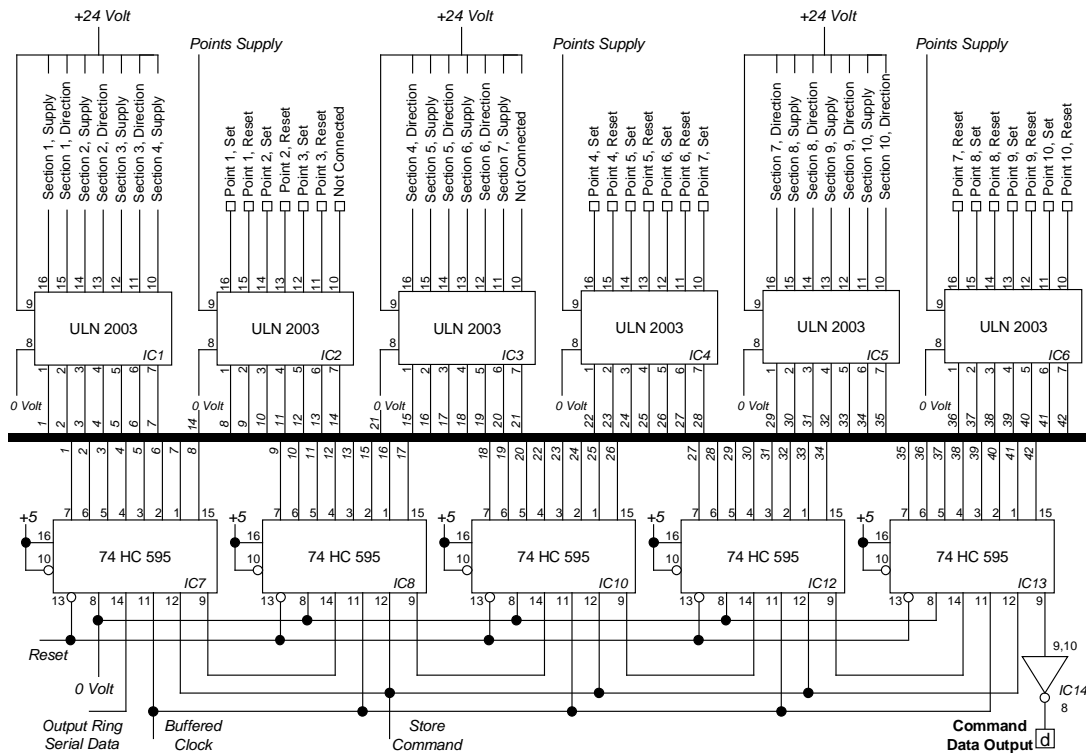


Figure 5 Output shift registers and Darlington drivers

It is worthwhile discussing the drive ICs (IC1 - IC6) in a little more detail because the function of these chips is very minimal but also very important. The ULN2003 ICs contain little more than a set of transistors configured as a group of Darlington pairs and a set of shunt diodes at their outputs configured to absorb any inductive voltage spikes which the external load might throw back. The Darlington pairs effectively provide a robust and higher current switching capability to assist the low current outputs from the 74HC595s.

Because the relay coils and point motors are supplied from different voltage rails, the point motor driver and the relay driver chips need to be segregated: each group with its own connection to the appropriate voltage rail via pin 9. Pin 9 appears to be a conventional power supply pin but is really only a common connection to all the internal shunt diodes: ULN2003s do not perform any logic functions or intrinsically consume power. If pin 9 is left unconnected or is incorrectly connected, the ULN2003s will seem to operate satisfactorily for a while, but in time, an incorrectly wired ULN2003 is likely to be destroyed through inductive kickback transients.

Special mention should be made about the power-on initialisation of the output drivers. Upon power up of the train controller card, the state of each of the 74HC595 devices will be indeterminate and potentially, some of the point motor drivers will be in an 'on' state. Even though the point motor supply on the train controller card has been regulated to limit the steady state current it provides, the regulator itself will quickly get hot because it dissipates more than 10 Watt in this state. The LM317 regulator is rated to continuously dissipate this power and I've tested and run mine to prove to myself that the board won't "burn" itself, but it is still possible to "burn" yourself if you inadvertently brush against the hot heat sink.

To reduce the chance of this, a reset circuit will be described in the next article to hold the outputs of the 74HC595 devices in a tri-stated mode until the software sends the first store command onto the

ring. In the absence of an electrical input, the ULN2003 devices will switch their outputs off and the point motor regulator will remain cool.

The repetition in the output shift register and Darlington driver circuit illustrates the simplicity with which additional track blocks or point motor drivers could be added or removed. If for any reason, “ten sets” is the wrong number for your layout, the chain of 74HC595s could be extended or reduced to suit the individual layout requirements with appropriate adjustment in the number of the accompanying Darlington drivers.

In order to help you decide the best strategy for extending or contracting the number of point controllers and track blocks from each train controller card, you will first need to decide upon the maximum number of trains you would like to independently control on your layout. If the layout (or budget) is small and you only foresee a maximum of two trains, then this output chain (and the accompanying block detection circuits) could be extended to provide as many independent blocks as required. Remember that with two independent power supplies, two trains can be controlled without restriction on an arbitrarily large layout. However if the layout needs to support more than two trains at the same time, there will be a clear need for more than just the two power sources provided on a single controller card and at the same time, you can support more blocks and points by building a second or third controller card.

If instead of constructing a single 10 block train controller card (as described in this article), you construct two cards, there will be a total of four independent power supplies, 20 track blocks and 20 point controllers available. Although the second train controller card will have an additional two power supplies, only one additional train (making a total of three trains) can be practically added to the layout and even then with a constraint. When a train moves from the domain of one train controller card to the next, the new controller card needs to have at least one “unused” and available power supply. Think about it: it would be rather difficult to move an additional train into the domain of a controller card which was already controlling two trains.

This means that if you tried to run four trains on a layout with only two controller cards, trains could never switch from the track domain of one controller to the domain of the other. That is, each train would be constrained to operating in just one half of the layout!

The point motor drivers are also implemented using the ULN2003 ICs but because the resistance of a point motor solenoid is so exceptionally low (generally in the range 2.5 Ohm to 5 Ohm) and their drive voltage high (generally 16 VAC or 24VDC) the ULN2003 is unfortunately unable to drive point motors directly because of the substantial current which the point motor coils draw.

In the next article, we'll continue this series by seeing how the point motor current problem is solved and also present the remainder of the train controller circuit. In the following months, I'll provide some tips on construction, testing and configuring the software.

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.

Parts List for one train controller

Resistors (all 0.125W unless marked)

1 x 2R (5W)
1 x 2R (1W)
23 x 100R (0.25W)
3 x 220R
2 x 1k
1 x 1k2
3 x 1k8
6 x 2k2
1 x 2k7
1 x 3k0 (or equivalent parallel arrangement)
1 x 6k8
1 x 8k2
44 x 10k
55 x 22k
21 x 33k
1 x 270k
1 x 470k
5 x 2M2

1 x 1k trimpot

Capacitors

11 x 0.1 uF ***
1 x 10 uF 5V tantalum
1 x 47 uF 10V tantalum
1 x 47 uF 25V electrolytic
2 x 1000 uF 25V electrolytic
1 x 2500 uF 25V electrolytic
1 x 10,000 uF 35V electrolytic

*** Note *** 0.1 uF power supply bypass capacitors should also be liberally used in accordance with your layout implementation and are not shown in the circuit diagrams

Semiconductors

16 x 1N5397 (1.5A rectifier or equivalent)
1 x 1N4148 high speed switch (or 1N914)
32 x 1N4007 (1A rectifier or equivalent)
1 x Red LED
1 x Green LED

3 x IRF9510 (or IRF9530 or equiv.) P channel power MOSFET
13 x 2N3904 NPN GP transistor (or BC549 or equivalent)
11 x 2N3906 PNP GP transistor (or BC559 or equivalent)
20 x MJE2955T PNP TO-220 power transistor (or equivalent)
2 x MJE3055T NPN TO-220 power transistor (or equivalent)

Integrated Circuits

6 x ULN-2003 (or MC1413 or equiv.) Darlington drivers
7 x 74HC595 8 bit shifting output latches
2 x 74HC597 8 bit shifting input latches
1 x 1488 (TTL to RS-232 Tx. Could adapt for MAX-232)
1 x 1489 (RS-232 to TTL Rx. Could adapt for MAX-232)
1 x 74HC00 quad NAND

2 x TL074 quad op amp
1 x 74HC123 dual monostable
4 x LM317 TO-220 voltage regulator IC

Miscellaneous

220mm x 220mm Veroboard (or larger)
10 x SPDT relay, 24V coil, 2A contacts
10 x DPDT relay, 24V coil, 2A contacts

5 x 14 pin DIL IC socket
16 x 16 pin DIL IC socket
9 x TO-220 heat sinks (for TO-220 cased main-board mounted transistors and MOSFETs)

Push Button Switch, Normally closed

44 way screw or IDC type power connector for track and point motors (or multiple smaller connectors)

5 way screw terminal connector for power supply

2 way screw terminal connector for points supply

Connectors, sockets and cables for east and west data interfaces (suggest DB-9 style)

PC switch mode power supply which includes a 24V 2A output

Estimated cost: \$A250, variation depending mainly upon the chosen relays, power transistors and construction method

Parts List for one PC interface card

Resistors (all 0.125W unless marked)

4 x 100R
2 x 680R
4 x 1k
2 x 10k

Capacitors

1 x 0.1 uF

Semiconductors

1 x 1N4007 (1A rectifier or equivalent)
2 x Green LED

1 x IRF9510 (or IRF9530 or equiv.) P channel power MOSFET
1 x 2N3904 NPN GP transistor (or BC549 or equivalent)

Integrated Circuits

1 x 1488 (TTL to RS-232 Tx. Could adapt for MAX-232)
1 x 1489 (RS-232 to TTL Rx. Could adapt for MAX-232)

Miscellaneous

50mm x 50mm (or equivalent area) Veroboard

2 x 14 pin DIL IC socket

Connectors, sockets and cables for east and west data interfaces (suggest DB-9 style)

Connectors, sockets and cables for parallel port interface

Box - anything simple

Estimated cost: \$A20