WWU Train Lab Documentation

Cody Baxter

Western Washington University

6/5/2011



# Welcome

Welcome to the Computer Science department’s Real Time Systems model train lab. This lab is primarily used for upper division courses concerning real time programming. This document will give you an overview of the lab and also walk you through starting up and running trains without the use of a program. It will also explain how to program the different electronic items on the layout and also troubleshot any problems that might arise.

You will first be given a brief overview of the layout and components, including the different electronic boards and items needed to run the layout. Then it will explain how to get a train running. Once you have the train running this will then go into how to configure and program the electronic items and use the computer to do some of this also.

Table of Contents

[Welcome 1](#_Toc330396372)

[1. Overview 5](#_Toc330396373)

[1.1. Trains used 5](#_Toc330396374)

[1.2. Track plan 6](#_Toc330396375)

[1.3. Hardware used 9](#_Toc330396376)

[2. Running a train without the computer 15](#_Toc330396377)

[2.1. Starting up the system 15](#_Toc330396378)

[2.2. Selecting an engine 17](#_Toc330396379)

[2.2.1. DT402D 17](#_Toc330396380)

[2.2.2. UT4D 19](#_Toc330396381)

[2.3. Controlling speed 19](#_Toc330396382)

[2.3.1. DT402D 20](#_Toc330396383)

[2.3.2. UT4D 20](#_Toc330396384)

[2.4. Controlling functions 20](#_Toc330396385)

[2.4.1. DT402D 21](#_Toc330396386)

[2.4.2. UT4D 22](#_Toc330396387)

[2.5. Controlling turnouts 22](#_Toc330396388)

[2.5.1. DT402D 23](#_Toc330396389)

[2.5.2. UT4D 24](#_Toc330396390)

[3. Running the Trains using JMRI 25](#_Toc330396391)

[3.1 LocoBuffer 25](#_Toc330396392)

[3.2 JMRI 25](#_Toc330396393)

[3.3. Starting 25](#_Toc330396394)

[3.4. Starting JMRI 26](#_Toc330396395)

[3.5. Monitoring the Loconet with JMRI 26](#_Toc330396396)

[3.6. Running a train with JMRI 28](#_Toc330396397)

[3.7. Controlling turnouts with JMRI 30](#_Toc330396398)

[4. DCS200 and Loconet 32](#_Toc330396399)

[4.1 DCS200 32](#_Toc330396400)

[4.1.1 Reinitializing the DCS200 33](#_Toc330396401)

[4.2 Loconet 34](#_Toc330396402)

[4.2.1 Loconet Message Paths 38](#_Toc330396403)

[5. Turnouts 41](#_Toc330396404)

[5.1. Turnouts on layout 41](#_Toc330396405)

[5.2. Tortoises 42](#_Toc330396406)

[5.3. DS-64s 42](#_Toc330396407)

[5.4.    Wiring 44](#_Toc330396408)

[6. Power Control of Track 47](#_Toc330396409)

[6.1. Wiring 47](#_Toc330396410)

[6.2. Reversing sections 48](#_Toc330396411)

[6.3.    Boards 48](#_Toc330396412)

[7. Sensors 51](#_Toc330396413)

[7.1. Reed switches 51](#_Toc330396414)

[7.2. Breakout board 51](#_Toc330396415)

[7.3. TC-64s and Sensor Numbers 52](#_Toc330396416)

[7.4. Wiring 53](#_Toc330396417)

[8. Programming 55](#_Toc330396418)

[8.1. Engines 55](#_Toc330396419)

[8.1.1 Yellow Engine Programming 55](#_Toc330396420)

[8.1.2 Atlas Engine Programming 56](#_Toc330396421)

[8.2. DS-64s 57](#_Toc330396422)

[8.3. TC-64 Towers 57](#_Toc330396423)

[8.4. Power District boards 62](#_Toc330396424)

[8.5. DCS200 62](#_Toc330396425)

[9. FAQ section 64](#_Toc330396426)

[Appendix 67](#_Toc330396427)

[A.      Standards 67](#_Toc330396428)

[A.1.   DS-64s 67](#_Toc330396429)

[A.3.   Towers 67](#_Toc330396430)

[A.4.   Power District boards 67](#_Toc330396431)

[A.4.1 Power District set-up 67](#_Toc330396432)

[A.4.2 Reversing Section set-up 67](#_Toc330396433)

[A.5.   Main unit 67](#_Toc330396434)

[B.      Diagrams 68](#_Toc330396435)

[B.1.   Trackplans 68](#_Toc330396436)

[B.2.   Wiring Diagram 75](#_Toc330396437)

[B.3.   DCC Stuff 78](#_Toc330396438)

[B.4. Throttles 83](#_Toc330396439)

[B.6.1 DT402D 84](#_Toc330396440)

[B.6.2 UT4D 84](#_Toc330396441)

[C.      Trains 85](#_Toc330396442)

[C.1.   Engines 85](#_Toc330396443)

[C.1.2 Function Numbers 85](#_Toc330396444)

[C.2.   Cars 85](#_Toc330396445)

[C.3.  Cabooses 86](#_Toc330396446)

[C.4.   Track Cleaning Car 86](#_Toc330396447)

[D. Manuals 86](#_Toc330396448)

[References 87](#_Toc330396449)

# Overview

In this overview section you will get an understanding of the numerous components that make up the train lab. This overview will include the trains, track and other components on the top of the layout. You will also get an overview of the electronics under the layout.

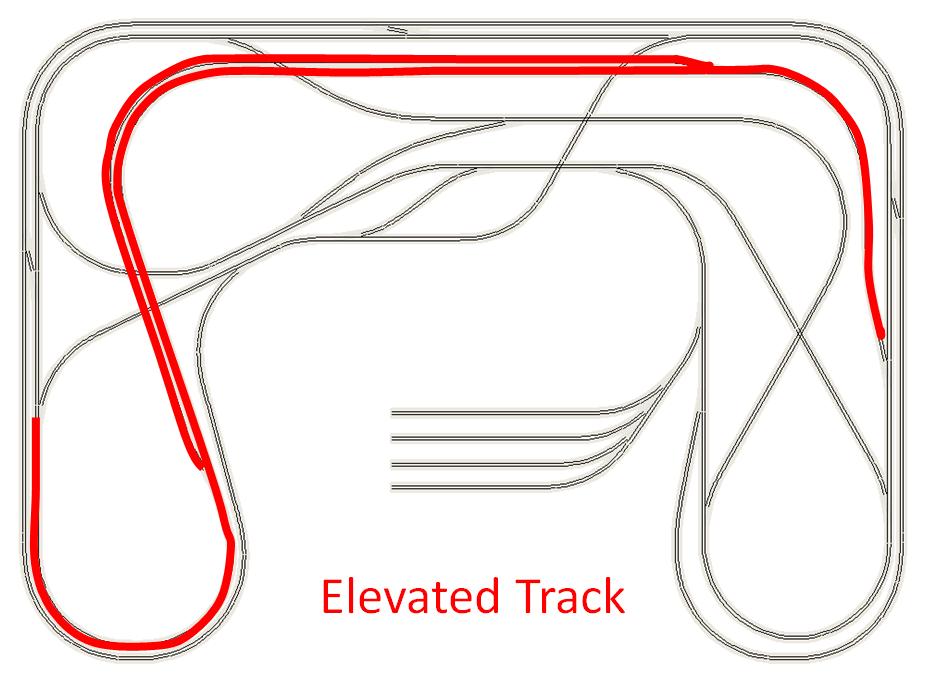
1.1. Trains used

The ***trains*** are one of the most important parts of the entire train lab. They are the physical items that move around the layout when it is running. A train is made up of three key parts: the ***engine***, ***train cars***, and the ***caboose***.

|  |  |
| --- | --- |
| The first part is the engine, which is on the front of the train. There are currently four different engines for use on the layout. All engines are labeled on their fuel tank what their number is; this is used to run the train. The number labeled on their fuel tank is assigned to them by programming the engine with that number. This will be discussed later in the programming section.  Each engine has a light on both the front and rear of the engine. The engine also has a speaker inside that provides sound. Both the lights and sound are controlled by the ***throttle***, also explained later. The yellow rngine does not have any sound. | ATL-10000272-0.jpgATL-10000287-0.jpgATL-10000332-0.jpg |
| Train cars make up the second part of the train. A train can have zero to numerous train cars on a train. There are currently 15 train cars available for use on the layout. | ATL-1600-0.jpgATL-1930-0.jpgATL-10643-0.jpgATL-10902-0.jpgATL-16055-0.jpgATL-19891-0.jpgATL-20000140-0.jpgATL-20000154-0.jpgATL-20000197-0.jpgATL-20000202-0.jpgATL-20000206-0.jpgATL-20000336-0.jpgATL-20000340-0.gifATL-20000341-0.jpg |
| The caboose is the last part of the train. Each train has one caboose at the end. There are currently three cabooses available for use on the layout. | ATL-62121-0.jpgATL-13122-0.jpgATL-62101-0.jpg |

More information about the trains can be found in Appendix C.

## **1.2. Track plan**



(Diagram taken from reference 4)

This is a diagram of the ***track plan*** for the ***layout***. The red section on the diagram is the elevated portion of track. The design was created to make the layout have the most crossovers and intersections within different sections, thus making the programming of the layout more difficult.

The main part of the top of the layout is the ***track***. It is composed of three main components: plain track, crossings and turnouts.

|  |  |  |  |
| --- | --- | --- | --- |
| The ***plain track*** is composed of sections where the track does not cross over another plain track. Highlighted on the diagram to the right are examples of plain track.  With plain track you may also have ***inclines***, which allow the train to change elevation. The inclines allow the train to transition from a flat area to an elevated area. On the elevated area, the plain track will also run on top of bridges.  ***Bridges*** allow the plain track to go over gaps in the elevation. | | continous sections.jpg | |
| ***Crossings*** are where two plain track sections intersect each other. The plain track in a crossing is connected together as one piece of track. The trains can only go straight through a crossing and can’t go to the other plain track. There are 3 crossover sections on the layout, with two highlighted in the diagram to the right. Bridges are not considered crossover sections, since the tracks are at different levels. | | | cross over sections.jpg |
| ***Turnouts*** allow a train to move from one plain track to another. They are located in several spots on the layout, with several highlighted in the diagram to the right.  Turnouts have either a ***closed*** direction or ***thrown*** direction. Pictured to the right is a turnout, with the different directions labeled.  The directions of the switches are controlled by a ***Tortoise*** slow motion switch machine, pictured to the right. The Tortoises are connected to the turnouts by way of a rod.  There are three pairs of turnouts on the layout that work together. These are called ***crossovers***. When you change the direction of one of the switches in the pair, the other switch also changes direction. The pair of switches are either set in the close direction or thrown direction. They can’t be set opposite of each other. | switch sections.jpg  Closed  Thrown  http://www.circuitron.com/index_files/image4321.gif  Crossover switch.jpg | | |

All of the track components are connected by either a metal or plastic ***track connector***. The metal track connectors are conductive and electrically connect one track component to another. The plastic track connectors are non-conductive and do not electrically connect one track component to another.

1.3. Hardware used  
  
The hardware used on the layout consists of all the electronic components used to run the layout. The hardware consists of items from Digitrax, RR-Cirkit, Circuitron, and a few items from other manufactures.  
  
***DCS200 --*** The Digitrax items include all electronics used to control the trains and turnouts. The DCS200 is the main control unit for the entire layout.

The DCS200 gets all of its power from an AC transformer that is plugged into a normal wall outlet. The transformer converts the AC power to DC. The DC power is what the DCS200 uses for all of its power needs. The main function of the DCS200 is to take the DC power that it gets from the transformer to run the trains. This DC output of the DCS200 can range from 12 volts to 20 volts. The setting that we use on the layout supplies 15 volts.

The DCS200 has two settings that are controlled from two switches located on the front of it. The first switch controls the amount of power that is put out. This switch is labeled “Scale” and should always be set to HO. The second switch, the “Mode” switch, controls the on and off of the DCS200. This switch should always be in the run position when using the layout.

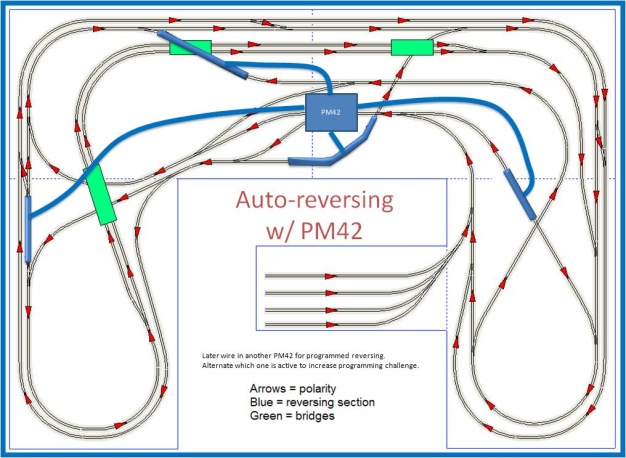
The DCS200 provides output in two ways. First, it uses the DC power that was converted from AC in the transformer, and sends that to both of the PM42 Power Managers. Before sending the DC power to the PM42s, the DCS200 imposes a square wave signal on the DC power. The square wave signals are the messages that the DCS200 sends out to the engines on the track. The messages will be discussed in greater detail later.

The second way the DCS200 provides output is over the Loconet. The ***Loconet*** is the layouts own internal network. The Loconet has its own set of protocols and message standards that it uses. This will be discussed in a later section. The Loconet is also how the DCS200 sends and receives messages so that it can tell the engines what to do. All of the electronic boards on the layout are connected to the Loconet. This includes the PM42s, DS-64, and UR92 Duplex Transceiver/IR Receiver. The information provided below will give you enough of an overview so that you can start running trains. More information on each component will be talked about later.

***PM42 --*** The first PM42 takes the power it gets from the DCS200 and sends it to four different sub-districts on the layout. The track is divided into four ***sub-districts*** by using the plastic track connectors. This allows the power from one output of the PM42 to go to that section of the track but nowhere else.



PM42

This PM42 also provides short circuit protection for each district. The track is shorted when a conductive object is placed across the rails. This can happen when a train de-rails or a metal object is set on the track, like a screw driver or nail. This then causes the short circuit protection on the PM42 to trip, shutting power off to that sub-district. If there wasn’t a PM42 and the plastic track connectors separating the track into four sub-districts, then when the track shorts the entire layout would shut down.

The second PM42 provides power to the four ***reversing sections*** on the layout. The reason for the reversing sections is there are four places on the layout where two components of track come together but the polarity of the DC power is incorrectly matched on the track. A reversing section (blue, in the picture to the right) is placed between these two components of track. A reversing section has plastic track connectors on each end of it, isolating it from the other track components. When a train has wheels on both the plain track and the reversing section, this creates an electric connection between the two parts and a possible short circuit. What the PM42 does is it detects if there is a short circuit. It then quickly flips the polarity in the reversing section. When the train exits it does the same thing again. This makes sure that the layout does not short out.

|  |  |  |
| --- | --- | --- |
| ***DS-64 --*** The DS-64, pictured to right, provides control for the Tortoise switch machines. The DS-64 has four or five Tortoises connected to it by wire. The DS-64 takes in commands over the Loconet and either sets the Tortoise in the closed or thrown direction. The DS-64 also reports when the Tortoise has finished moving. | | http://www.digitrax.com/images/statdec_ds64.jpg |
| ***Throttles --*** We have two types of Throttles for use with the layout. We have a DT402D and UT4D Throttle. They take in all of the user input. Throttles need 9v batteries installed in them when they aren’t plugged in. Both throttles communicate with the DCS200 via the UR92, discussed later. Both Throttles can communicate with the UR92 in three ways: Plugged in, IR, and Wireless. When you initially turn on the power to the layout and put batteries in the throttles, each one will need to be plugged into the UR92 | | |
| ***DT402D Throttle*** -- The DT402D Throttle pictured below has several main areas. At the top is the display. This gives you current information about the trains and also the mode you are in. The Throttle also has a left and right control. Each control has two main parts. The control knob allows you to control the speed of the train and also change direction by double clicking the knob. Both of the control knobs are located at the top of the Throttle. Each control also has a direction button, it gives you control over the train’s direction. When changing the speed or direction of a train, it makes that control the active control. Having two controls allows you to control two trains.  There are four mode buttons so that you can enter the different modes: Function, Mu, Loco, Switch. In the lab we only use the Function, Loco and Switch mode.  The ***Function mode*** allows you to control the different functions on the engine when you are running it. This is the default mode when the throttle starts and after you have selected an engine.  The ***Loco mode*** allows you to select which engine you want to run.  The ***Switch mode*** allows you to close or throw the different turnouts on the layout.  You then have the ***numeric keypad***. This allows you to control functions when you are in the Function mode, enter an engine number in Loco mode or enter a switch number when in Switch mode.  Beneath the mode buttons and between the direction buttons are the ***Yes(+)/No(-) buttons***. These buttons are used for turning track power on/off and allow the user to input a yes or no command into the Throttle.  Next are the ***programming buttons***. They are used when you are programming either an engine or electronic board on the layout. More on programming will be discussed later.  The ***switch buttons***, at the bottom, allow you to either close or throw a switch when in Switch mode.  The ***power button*** on the bottom left of the Throttle, puts you in the Power mode. This allows you to use the +/- buttons to turn the track power on or off.  The DT402D Throttle display gives you a lot of information about what the Throttle is currently doing.    The display has several different areas to discuss.   * Engine numbers – There is a left and right engine number. If a SEL appears in either engine number then there is no engine currently selected for that control. * Mode – This displays the current mode that the Throttle is in. * Engine Speed – Above the Engine Numbers will be the engine speeds. This will be zero if there is no engine selected or if the engine selected is stopped. The speed ranges from 0 to 99. * Speed Graph – Above each engine speed is a graph showing the speed. * Functions – Above both engine graphs is a row of numbers. If the number is on, then that means that function is turned on for the active engine. * Active control – To the left/right of each engine number is a small engine symbol with smoke. If the smoke is blinking that means that that control is the active control. When you change the direction or speed of a train, it automatically becomes the active control. When either the left or right control has no engine selected, the engine will not be displayed. * Track Power indicator – To the right of the function numbers is a power indicator. If the indicator is on, then the track power is on. If it is off, then the track power is off. | | |
| ***UT4D Throttle --*** The UT4D Throttles are the less advanced Throttles. They have four main parts. First you have your direction switch on the top of the throttle. The direction switch has a reverse, brake and forward setting. When in the brake position, the engine selected will stop and not respond to any speed changes. Then you have your speed knob which controls the train speed.  Next you have your address selection dials. This is how you select the train you want to run. Then you have your function keys.  The UT4D does not allow you to close or throw switches. Although the UT4D doesn’t allow you to control switches, the control of a train is identical to the DT402D Throttle. |  | |
| ***UR82 Duplex Radio Transceiver --*** The UR92 is the bridge between the Loconet and the Throttles. It either gets information from the Throttles and sends it over the Loconet or gets information from the Loconet and sends it to the Throttles.  The UR92 allows for three different ways of communication with the Throttles: Plugged in, IR, and Wireless. All three methods can be used at the same time by three different throttles. | Wirle UP92 Duplex | |

This is a brief overview of the trains, track and electronics on the layout and more will be talked about in later sections and pictures/diagrams can be found in the Appendix. This provides you with enough information so that you can start running the trains.

# **2. Running a train without the computer**

In this section you will find information on how to start up the system, run a train and control basic functions on the layout. For this section you will be using both types of throttles, DT402D and UT4D.

2.1. Starting up the system  
  
When arriving at the layout you will find that it is shut down and that all of the power to it is turned off. In order to run the trains, you must get the main system powered on and the track power turned on.

First, you will want to turn on all of the power strips located under the layout near the location of the computer and main command station. This will turn on the power to all of the electronic items: command station (DCS200), PM-42s, DS-64s, UR92, and Tower Controllers. You will hear the Tortoises move into position.

Once you have turned on the power strips you will want to verify that the settings on the command station (DCS200) are correct. The “Scale” switch should be in the “HO” setting. The “Mode” switch should be in “Run”. Once the settings have been verified to be correct, we can get the track power turned on.

|  |  |
| --- | --- |
| You will want to plug the DT402D Throttle into the UR92 receiver. Once you connect the throttle to the receiver, it will display a few start up screens. | (From reference 2)    (From reference 2) |
| Once the Throttle has started, you will get the following screen (the image will not fully match physical Throttle screen).  This screen image displays a throttle with the power turned off and engines 25 and 5234 selected. Your throttle may display SEL if an engine isn’t selected or different engine numbers corresponding to engines on the layout. | (From reference 2) |
| To get to the correct mode to turn on the power, you press the power button located on the bottom left of the throttle. | (From reference 2) |
| Once you click this button you will get the following screen on the throttle. | (From reference 2) |
| Once at this screen you can turn the track power on or off. To turn the track power on you will press the + key and to turn the track power off you will press the – key. | (From reference 2) |
| Once you press the + key you will see this screen.  The only change in the screen should be the power indicator in the upper right hand of the screen. In this image it is highlighted red, it will be black on the screen. The track power should now be on and you should hear some sounds from the engines on the track, if they aren’t muted  If the power indicator is blinking, just press the + key again and the track power should be on. If for some reason the track power isn’t coming on, turn the power off then back on to the system by way of the power strips and start the process over. | (From reference 2) |
| Now that the track power is on you can disconnect the throttle from the UR92 and you should see the following on the screen. You will notice that the wireless indicator is now on (highlighted in red). This means that the throttle is in wireless mode. | (From reference 2) |

If you don’t see the wireless indicator, refer to the FAQ section on how to get wireless working on the throttle.

You are now ready to run a train. Section 2.2 explains how to select an engine, 2.3 tells you how to control the speed of a train, 2.4 explains controlling functions of an engine, and 2.5 tells you how to throw switches.

## 2.2. Selecting an engine

This section describes how to select an engine with the two types of throttles used on this layout. If you have not already done so, please turn on the track power. Section 2.1 explains how to turn the track power on.

By selecting an engine on the throttle you are setting the throttle up so that you have access to that engine. After selecting an engine you will be able to control its direction, speed and numerous light and sound functions. Without selecting an engine the throttle has no way of knowing which engine you want it to control.

You will want to make sure that the engine you are using is on the track.

### 2.2.1. DT402D

The first throttle that we will use will be the DT402D Throttle.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| First you will want to check and make sure that the track power is on. This can be done by looking at the throttles display and seeing the track power indicator is on (highlighted in red).  This throttle screen has no engine selected on the left control and engine 00 selected on the right control. | | | (From reference 2) | | |
| Now decide which control you want to put the train on, either the L or R. Activate the DT402’s (L/R) control by turning the control knob a ¼ turn in either direction or by pressing the (L/R) direction button once if there is a current engine address or double click if “SEL” appears in the (L/R) address. If there is an engine address, the (L/R) Loco Smoke will start flashing. If no engine has been previously selected on this throttle it will flash “SEL”.  After you have selected the control you want, press the Loco key. | | | | | (From reference 2) |
| After pressing the LOCO key the Throttle screen will begin to flash and you will see a screen similar to this (This is if you are selecting on the left control). The right engine number may be different then the image. The image has engine 00 selected on the right throttle.  The SEL on the screen should be flashing at this time. This means you are ready to enter the engine address into the throttle. If there is already an engine selected, then the engine number will be flashing. | (From reference 2) | | | | |
| Using the numeric keypad enter the engine address that you want to use. The engine address is the number on the side of the engine or located on the bottom.  You will now see a screen like this (This screen shows someone selecting loco number 3).  The stat 128 displays in the information message area, means that the engine is setup for 128 speed steps, explained later. This is the standard for all engines on the layout. | | (From reference 2) | | | |
| Once you have entered the number you want, press the Loco key to make the selection. You will now see the engine number stop blinking and the smoke above the train, on the screen, to start blinking. (The smoke is colored red in the picture below) If you get a message saying steal, press the Y key.  It may say “steal” on the screen because the engine may already be selected on another throttle. The system allows you to select an engine that is already in use but only after you tell it you want to.  This throttle screen shows the left control having engine 3 selected and the right control with engine 00 selected. Both engines have a speed of 0 and the left control is the active control. No functions are turned on for the active engine. | | | | (From reference 2) | |

At this time you are ready to start running the engine you have selected. You should have also noticed that the sound on the engine has started, if it wasn’t already on. You can now go to section 2.3 to see how to change the speed.

### 2.2.2. UT4D

The first step in selecting an engine with the UT4D is to dial up the engine address. By using the dials on the front of the UT4D select your engine address. If the address is only two digits long, leave the first two dials set to zero.

Once you have dialed up the Engine address, simply press the SEL key and you have selected the engine. If the light is red, this means that you will need to steal the Engine. Simply press the STEAL button and you will have control of the engine.

## 2.3. Controlling speed

Now that you have selected your Engine and turned the track power on, you will learn how to control the speed and direction of the engine. If you haven’t already done so, please select an engine by refereeing to section 2.2.

### 2.3.1. DT402D

The directions listed here will apply to either of the controls on the DT402D Throttle you are using. There are two ways to change the direction of a train. The simplest way is to press the change direction button.

You simply press the change direction button. This also sets that control to the active control.

(From reference 2)

There are two buttons pictured here, one is for the left throttle (L) and one is for the right throttle (R). Pressing either button will change the corresponding direction of that train.

The other method for changing the direction of a train is to double click the control knob. Doing this will cause that control to change direction. This is the non-preferred method, since it can sometimes be a little harder to do.

When you change the direction of an engine, the speed stays the same as it was before the direction change. The engine will although come to stop and then start going in the other direction, matching the same speed it had before. This also applies to the UT4D Throttle.

On the DT402D all you have to do to control the speed is turn the control knob of the throttle you want to change the speed on. Turning the control knob right increases the speed and turning it left decreases the speed. When you change the speed, it also sets that control to the active control.

### 2.3.2. UT4D

Changing the direction of a train using the UT4D consists of flipping the direction switch located at the top of the Throttle. All that you need to do is flip the switch to the direction you want. Flipping the switch to the left sets the direction to reverse and flipping it right sets the direction to forward. If you set the switch to the middle, then any speed changes are not sent to the train until a direction is chosen.

Now that you can change direction, controlling speed will be easy. On the UT4D all you have to do to control the speed is turn the knob on the Throttle. Right increases the speed and left decreases the speed.

## 2.4. Controlling functions

Now that you have selected your engine, and can control the speed and direction of the engine, you will also see how to control functions of the engine. If you haven’t already done so, please select an engine by refereeing to section 2.2.

Compared to the other tasks you have done with the engine, controlling functions are much easier. All you must do is press a button. For now we will learn how to turn the light on and off, blow the horn, sound the bell and mute all the sound. For a list of the other functions that can be controlled refer to Appendix section C.1.2. Controlling those functions is the same as controlling the light.

### 2.4.1. DT402D

On the DT402D you have access to control functions numbered from 0 to 12. Although you have access to 13 function controls, you will most likely only use a few of them when running trains. When controlling the functions you will want to make sure that the train you want to control is currently active. This can be done with a single click of the control knob.

|  |  |
| --- | --- |
| Controlling the lights is done with function 0. This is the same with all trains on the layout. All you must do to control the lights is to press and release function button 0.  After pressing that button you will then see the function number appear at the top of the screen. If it was already on then it will turn off. This screen show engine 3 selected on the left control and engine 1652 selected on the right control. The right control is currently the active control and both engines have speeds of 0. You will notice that 0 has appeared in the row of function numbers, highlight in red. This means that function 0 is turned on for the left control. | (From reference 2)    (From reference 2) |
| To control the horn you will use function button 2. This is the same with all trains on the layout.  This function button is a little different than the others. You have to hold it down when you want it to be on. Once you let it up, it will turn the horn off. | Throttle Horn Button.jpg  (From reference 2) |
| To control the bell you will use function button 1. This button works the same as the light button. You will press and release it to either turn the bell on or off. | Throttle Bell Button.jpg  (From reference 2) |
| To turn the sound on or off you will use function button 8. When function 8 is turned on, the sound from the engine is muted. Meaning no sound, even if you press a function key that controls sound, will be heard. When function 8 is off, all of the sound will be heard. | |

When dealing with the other functions, you will just press and release the corresponding function button and that number will appear at the top of the screen, indicating that the function is on.

You can click the function button again to turn that same function off/on.

### 2.4.2. UT4D

Dealing with functions on the UT4D is the same as the DT402D. Although to access functions 7 thru 12, you must hold down the shift key when pressing that function button. Refer to the DT402D instructions on functions (section 2.4.1) for how to control functions. You will just use the UT4D numeric keypad when turning a function on or off. There is no display to indicate if the function is on or off.

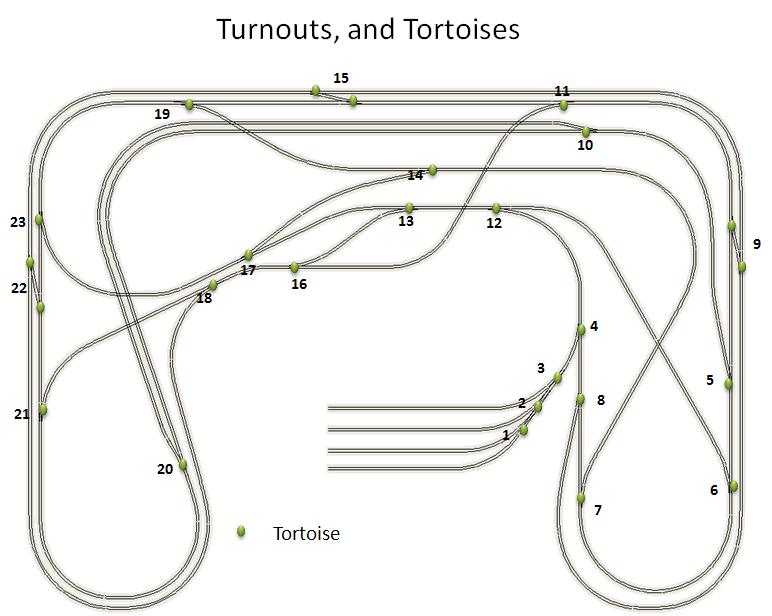
## 2.5. Controlling turnouts

In order to enjoy the full layout you need to be able to throw and close turnouts.

With the Throttles that we have, you can only control turnouts using the DT402D, not the UT4D. So we will cover just that Throttle.

All of the switches on the layout have a number corresponding to them. This number is used when you are controlling that switch from the Throttle or computer, talked about later. Each switch has a flag next to it displaying its corresponding number.

In the diagram below you will see all of the switches with their corresponding number. Again, you will notice that the crossover switches have one number corresponding to the pair of switches. This is because they work as one when you want to throw or close the pair of switches.



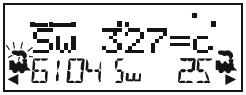
### 2.5.1. DT402D

With the DT402D, throwing switches is pretty simple. First you have to go into the mode that allows you to control switches. You will want to press the switch button, located in the top row of buttons on the right.



(From reference 2)

Once you press the switch button you will see the following screen. The screen shows that the Throttle is in Switch mode with switch 327 currently selected and set to close (c). Engine 6104 is currently selected on the left control and is the active control. Engine 25 is selected on the right control.



(From reference 2)

You will notice that the Fn for Function mode has now switched to Sw for Switch mode. At this point it is very easy to control a switch. You will enter the switch number, by using the numeric key pad, and then use either the close button to close a switch or the throw button to throw a switch.

(From reference 2)

Once you are done controlling switches you can press the Function key to return to the function mode.



(From reference 2)

### 2.5.2. UT4D

You cannot control switches using the UT4D.

# 3. Running the Trains using JMRI

In this section we will discuss using the computer to interface with the layout. You will learn how to start software called JMRI and how to use some of the features in this software. We will first cover starting the computer then starting the software. You will see how to monitor the Loconet and then how to run trains.

## 3.1 LocoBuffer



The LocoBuffer is the bridge between the Loconet and the computer. It allows a computer to receive all of the messages that are being sent over the Loconet. It also allows the computer to send messages to the Loconet from the computer.

The LocoBuffer employs an internal buffer the puts all messages that it sees coming over the Loconet onto the buffer. You are then able to grab those messages using a computer program and see what they are.

When running a computer program that needs to communicate with the layout, you will want to make sure that you have the LocoBuffer connected to any USB port on the computer. The LocoBuffer also requires that you have the most up to date drivers installed on the computer.

## 3.2 JMRI

JMRI (Java Model Railroad Interface) is a Java based program that uses the Locobuffer to communicate with the DCS200 and the other electronic boards over the Loconet. JMRI can do everything the Throttle can plus some.

JMRI should be currently installed on the lab computer. If not, it can be found at jmri.sourceforge.net. Here you can download the latest working copy and also find information on how to get the source code for the program.

The source code can be handy when developing your own software for use in the lab. Though it is in Java, it does give you an example on how JMRI handles connections to the Loconet and reading and writing to it.

## 3.3. Starting

In order to start JMRI you have to make sure you have done a few things. First, start the computer in the lab and boot it into Windows. Once Windows has started, log into the computer and wait for everything to load.

Next you need to make sure that the USB connection is connected to the computer. Find the USB cable from the layout and make sure it is connected to one of the USB connections on the computer. It doesn’t matter what USB connection you use. The system will recognize that it is the LocoBuffer.

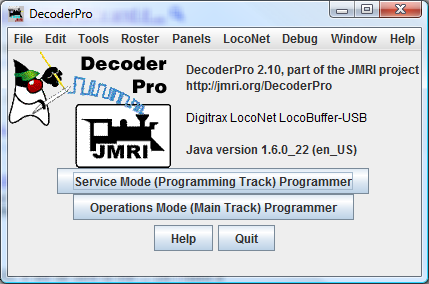
After making the USB connection and starting up the computer, you are now ready to run JMRI.

## 3.4. Starting JMRI

Now that you have the computer and USB connection ready, you can start JMRI. Locate the icon on the desktop and double click it.



Once you double click on the icon, the JMRI software will take a few seconds to load. Once the software has finished loading you will see the main software screen.



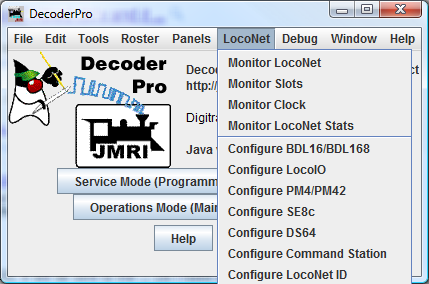
Now that you have the software loaded and running, you are set to start using it.

You can control pretty much any aspect of the layout from within JMRI. We will now cover three basic things: monitoring the Loconet, running trains, and controlling turnouts.

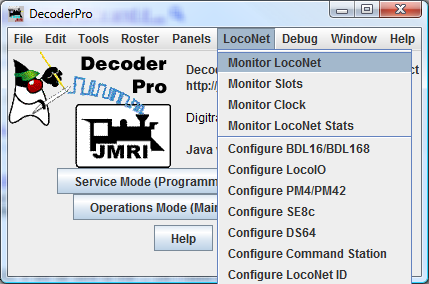
## 3.5. Monitoring the Loconet with JMRI

Monitoring the Loconet is one of the easier things to do with JMRI but also one of the more useful things with the program. This can be used to see the raw data that is being sent over the Loconet when stuff happens on the layout.

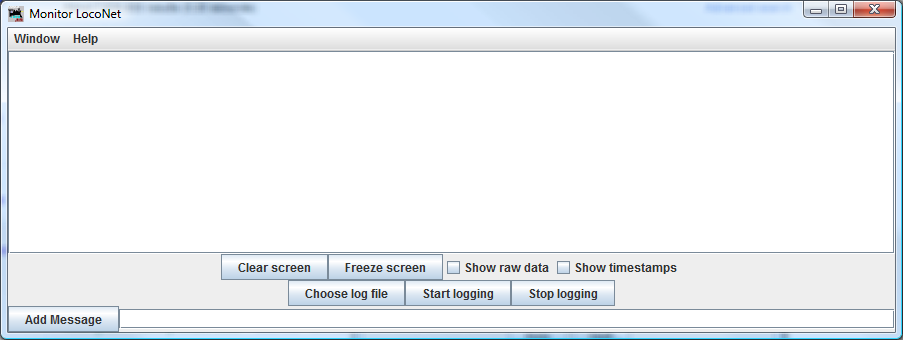
To get started, first click on the Loconet menu option.

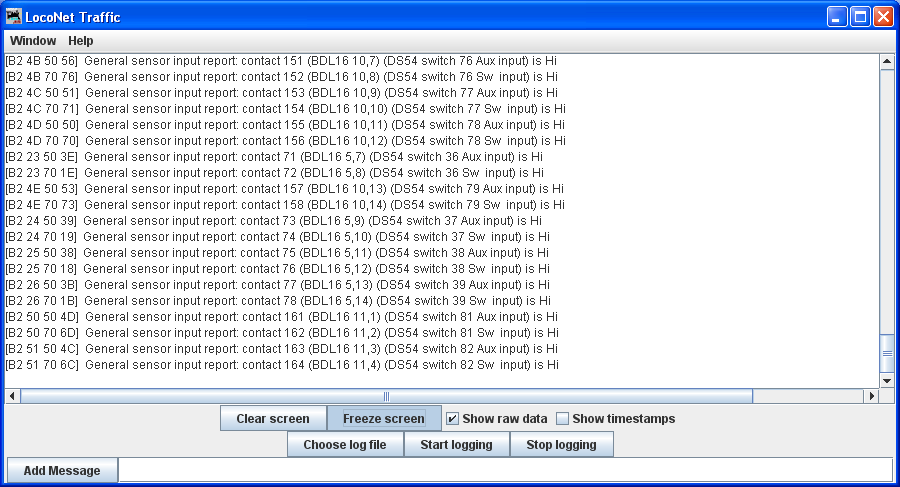


Next select the Monitor Loconet option.



This will open up a new window and you will start to see data being displayed.





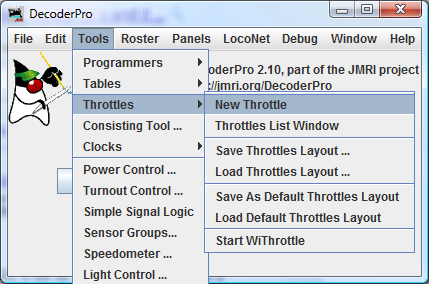
You have several different options within this window:

* You can freeze the screen, so that you can examine the data without having the screen continue to scroll.
* Display the raw data format. This displays the hex values of the bytes in the message.
* Show timestamps
* Start and stop logging to a file
* Clear the screen
* You can also send messages out over the Loconet within this window

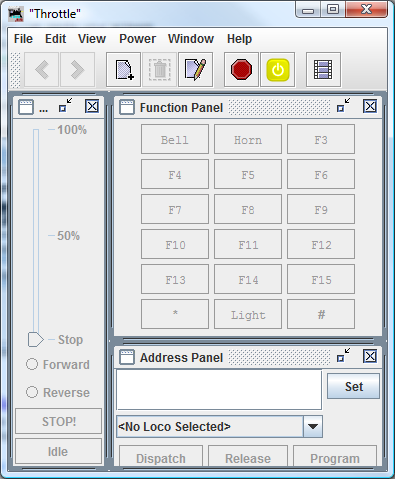
Monitoring the Loconet gives a lot of insight into how the Loconet functions and operates.

## 3.6. Running a train with JMRI

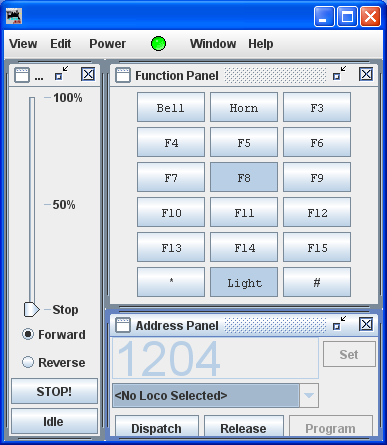
Running a train from within JMRI is easier than using the physical throttle. You will first select Tools-Throttles-NewThrottle.



This will then open up your throttle within JMRI.



In order the run a train we must first type in the train’s address in the Address Panel and click Set. Once that is done, everything on the screen will become active.

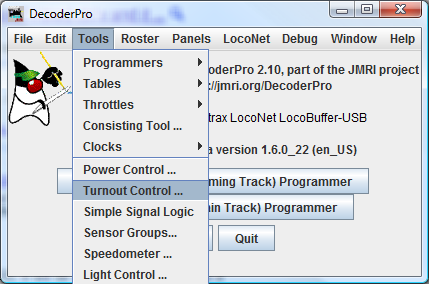


This screen shot shows engine 1204 selected and functions 0 (Light) and 8 on. The direction is set to Forward and speed set to 0.

You can now use the buttons to control the functions and the slider on the left to control the speed. You can change direction beneath the slider and also issue a Stop command.

## 3.7. Controlling turnouts with JMRI

To control turnouts in JMRI, you will want to select Tools-Turnout Control.



This will open up a new window where you can enter the turnout number and select either Closed or Thrown. It’s that simple.



# 4. DCS200 and Loconet

Both the DCS200 and Loconet have been mentioned just briefly before. Giving you enough information so that you have some understanding of how things work. Now we will explain more about the DCS200 and the Loconet.

## 4.1 DCS200

The main unit of the DCC system is the DCS200. It provides both the power and control for the trains. The DCS200 gets its power from an AC Transformer that is plugged into the wall. The power provides the DCS200 with the power it needs to run and also the power for the trains. Its main function is sending out messages to the trains. It does this by superimposing a square wave signal on top of the AC power that gets sent to the PM42s and then to the trains.

In order for the DCS200 to impose the square wave on the AC power, it must store information on the engines in its internal memory. It uses what are called slots to store each engine. There are 120 slots available to store engines, meaning that you can run up to 120 engines at one time. We currently run just four engines. When the Throttle or Locobuffer must address an engine, it uses the slot number in the messages that are sent over the Loconet, not the actually engine number. Although when we change the speed or direction on the Throttle or computer it seems like we are addressing the engine by its number. Both the Throttle and computer convert this into the internal slot number before sending the message off to the DCS200.

The Throttle or computer gets the slot number of the engine when it firsts registers it with the DCS200. When you select the engine on the Throttle/computer, the Throttle/computer sends a message over the Loconet to the DCS200 saying that it wants to register an engine with this number. The DCS200 then looks in its slots to see if that engine is currently in use.

If the engine is currently in use, it then sends a message back to the Throttle/computer asking if you want to steal that engine. If you say yes, the DCS200 then replies with a message that contains information about the current state of the engine and also the slot number.

If the engine is not currently in use, then the DCS200 finds the first free slot or the slot the engine was last in. A free slot is a slot that doesn’t have a currently active engine in it. The DCS200 then puts that engine number in that slot and sets the slot to active. The DCS200 then sends a message back to the Throttle/computer telling it the current state of the engine and also the slot number.

After the Throttle/computer has the slot number, it then refers to that when sending all of its messages over the Loconet. Whenever the DCS200 gets a message that needs to be sent to the trains, it takes the slot number in that message and looks in that slot. It then uses the engine number in that slot to address the message to the engine on the tracks.

The DCS200 uses a protocol similar to the Loconet message protocol, discussed later, to send information over the track to the Engine. When it sends a message it first sends a preamble. This informs all the Engines on the track to be ready for a message. Next are the address data bytes. This informs the Engine with that address to be ready for more information and that all other Engines can ignore the message.

After the address data bytes, there are one or more instruction data bytes. This tells the engine what it needs to do. Like reverse, speed up, or turn on a function. After all of the data bytes there is the error detection byte and an end packet bit. The error detection byte is used to determine if the message received is complete or if there was an error. If the error detection byte indicates an error, then the entire message is ignored.

The end packet bit is a 1 and the bits between each byte, end bits, are 0. Between each message the system continually sends 1’s.



### 4.1.1 Reinitializing the DCS200

When you turn the power off to the DCS200, it stills remembers all of the trains and settings that were stored in it the next time you turn it on. For this reason, there are a few steps that can be taken when starting up the DCS200 to reinitialize the system, clearing all of the stored data. You will want to perform this before doing anything in the system, as all data will be erased.

When reinitializing the DCS200 you will use the DT402D Throttle. First, move the MODE Toggle on the front of the DCS200 to “OP”. Then remove all LocoNet cables from the DCS200. Then connect the DT402D Throttle to either of the LocoNet jacks on the front of the DCS200.

On the DT402D, press the SWCH key. Then enter 36 and press the CLOC/c key. Then enter 38 and press the CLOC/c key. Finally, enter 39 and press the CLOC/c key. Next, unplug the Throttle from the DCS200.

After unplugging the DT402D Throttle, set the DCS200 MODE toggle to “SLEEP” mode. Then flip the toggle to “RUN”. This completes the Reinitializing of the DCS200. Reconnect the Loconet cables and you are ready to run.

## 4.2 Loconet

Next to the DCS200, the Loconet is the second most import part of the DCC system on the layout. The Loconet provides the communication between the DCS200, Throttles (by way of the UR92), and all of the electronic boards on the layout. The Loconet has its own set of protocols and message standards that are used for communication. We will discuss a little about the protocols and messages but more information can be found in the “Loconet Personal Use Edition 1.0 SPECIFICATION” Document. This document can be found at www.digitrax.com. All information below is taken from this document.

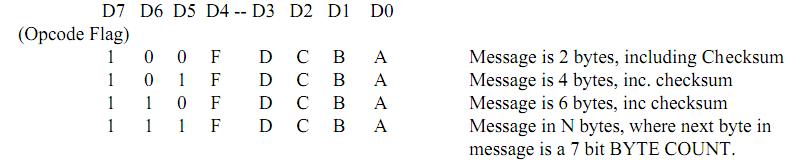
The Loconet is a “PEER to PEER” distributed network system on which all devices can monitor the network data flow. A “PEER to PEER” distributed network is a network where each Peer (Electronic Board) has equal privileges to each other. This means that each Peer can send or receive messages or pass messages along if they don’t need to handle them.

The Loconet uses a 6-wire flat cable for all of its connections. The Loconet is setup in such a way that you can run your cable from one board to another, then to another and so on. You don’t have to have everything coming back to a central location. When someone develops a network they have a choice, use an already defined standard or develop a new one. Digitrax decided to develop a new Protocol for the Loconet and this is what we will look at next.

A message on the Loconet contains multiple bytes. A message has at minimum 2 bytes and then must contain an even number of bytes from there. So a message can have 2, 4, 6, 8, 10… bytes in it. Each byte is represented as a hexadecimal number. When you convert the hexadecimal number into binary the most significant bit is called the “OPCODE Flag”. This allows you to determine if that byte is an Opcode. This will be further explained later.

The message format is of a certain format. The messages begin with what is called an opcode byte, then has 0 or some even number of data bytes, and then a checksum byte.

The Opcodes are how we determine what type of message we are dealing with. The opcode also allows us to determine how many bytes are in the message.



In the diagram above, it shows you how an opcode byte is formatted in binary. As always, the most significant bit is always one when you have an opcode and zero when you don’t. Then you can use bit 5 and 6 to determine the length of the message. If the message is 8 or more bytes long, then the next byte in the message will tell you how long the message is. Bits 0-3 are used to determine what Opcode or type of message you are dealing with. This allows for up to 32 Opcodes per message length. In normal use of the layout, you will most likely be dealing with eight different Opcodes, listed below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name of Opcode** | **Opcode in Hexadecimal** | **Use** | **Originate From** | **Used By** |
| Power message | 83(On), 82(Off) | To turn track power on or off. | Throttle / Computer | DCS200 |
| Sensor message | B2 | Used when a sensor, discussed later, is tripped. | Tower Controller, discussed later | Locobuffer |
| Set turnout message | B0 | Used to tell a turnout to move into the close or thrown position | Throttle,  Locobuffer | DS-64 |
| Turnout completed moving message | B1 | Used to report when a turnout has finished moving. | DS-64 | Throttle,  Locobuffer |
| Locomotive Request | BF | Used to request use of an engine. | Throttle, Locobuffer | DCS200 |
| Slot data message | E7 | Message that is sent by the DCS200 when an engine is selected. | DCS200 | Throttle, Locobuffer |
| Null Move | BA | Used to set a lot active/in use. | Throttle, Locobuffer | DCS200 |
| Slot Speed message | A0 | Sent when you want the speed of the train changed | Throttle,  Locobuffer | DCS200  The DCS200 then sends the message to the train by imposing a square wave on the power to the track |
| Slot Direction message | A1 | Sent when you want the direction of the train changed, or the light, horn, bell turned on/off. |
| Slot Sound Message | A2 | Sent when you want to change functions 5-8 |
| PM42 Message | D0 | Used to send a message when a short occurs on the track or a reversing section is crossed. | PM42 | Locobuffer |

Although this is just eight opcodes, the “Loconet Personal Use Edition 1.0 SPECIFICATION” document lists a total of 28 different opcodes for use.

The second part of a message is the data. This is contained between the opcode and the checksum bytes. This is where you get all the information about what needs to happen for a message. Below is a list of the eight different messages for the eight opcodes listed above.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of Message** | **Opcode** | **Total number of data bytes** | **Details** |
| Power message | 83(On), 82(Off) | 0 | These two messages just have an opcode and checksum bytes |
| Sensor message | B2 | 2 | The data bytes contain information on the sensor address and the state. |
| Set turnout message | B0 | 2 | The data bytes contain information on the turnout address and also the state that the turnout needs to be set to. |
| Turnout completed moving message | B1 | 2 | The data bytes contain information on the switch address and the current state of the switch. |
| Locomotive Request | BF | 2 | The data bytes contain the address of the engine you are requesting. |
| Slot data message | E7 | 11 plus one length byte | The data bytes contain information on the current state of the engine, including the slot number that the engine is in, speed and direction. |
| Null Move | BA | 2 | The data bytes both contain the same slot number. |
| Slot Speed message | A0 | 2 | The data bytes contain information on the slot number of the engine and speed |
| Slot Direction message | A1 | 2 | The data bytes contain information on the slot number of the engine and information on the direction and functions. |
| Slot Sound message | A2 | 2 | The data bytes contain information on the slot number of the engine and information on functions 5-8. |
| PM42 Message | D0 | 4 | The data bytes contain the PM42 address and also information on the state of the 4 power districts/reversing sections. |

The last part of the message is the Checksum byte. This byte will be different for every single message that is sent over the Loconet because it is computed based off of all the other bytes in the message. The Checksum byte is computed by taking the 1’s COMPLEMENT of the byte wise Exclusive Or of all the bytes in the message, except the Checksum itself.

The Checksum is used to validate data accuracy. If you take all the bytes in a correctly formatted message and take the Exclusive Or of all and the resulting byte value is “FF”, in hexadecimal, then the message data is accepted as good.

The following table lists all eight messages with their format and information on the data bytes. The Checksum byte is listed using <CHK>, as it is computed once the message is generated.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Type of Message** | **Format** | **Data Byte** | **Data Byte** | | |
| Power message | <83><CHK> (On)  <82><CHK> (Off) | None | None | | |
| Sensor message | <0xB2><IN1><IN2> <CHK> | <IN1>  <0,A6,A5,A4- A3,A2,A1,A0>  A0-A6 are address bits, that when converted to an integer from binary give you the address of the sensor you are dealing with. | <IN2>  <0,X,I,L- A10,A9,A8,A7>  A7-A10 are also part of the address of the sensor.  I tells you if the sensor is the first or second in a pair.  L is 0 for transition from high to low, 1 for transition from low to high.  X is reserved for future use. | | |
| Set turnout message | <0xB0><SW1><SW2><CHK> | <SW1>  <0,A6,A5,A4- A3,A2,A1,A0>  A0-A6 are address bits, that when converted to an integer from binary give you the address of the switch you are dealing with. | <SW2>  <0,0,DIR,ON- A10, A9, A8, A7>  A10,A9,A8,A7>  A7-A10 are also part of the address of the switch.  ON is equal to 1 for Output On or 0 for Output Off. This tells the system if you want feedback on the switch when it finishes moving.  DIR tells the switch what direction to move. 1 for closed or 0 for thrown. | | |
| Turnout completed moving message | <0xB1><SN1><SN2> <CHK> | <SN1>  <0,A6,A5,A4- A3,A2,A1,A0>  A0-A6 are address bits, that when converted to an integer from binary give you the address of the switch you are dealing with. | <SN2>  <0,1,I,L- A10,A9,A8,A7>  A7-A10 are also part of the address of the switch.  L tells you if the DS-64 is sending aux output or switch output.  I tells you if input is low or high or switch is closed or thrown. | | |
| Locomotive request | <BF><loco adr hi> <loco adr lo><CHK> | <loco adr hi>  This data byte allows you to have engine addresses above 127. It will be used when you want to address an engine from 128 to 9999. | <loco adr lo>  This data byte contains the least significant bits in the address. It will always contain information. | | |
| Slot data message | <0xE7><0E><SLOT#><STAT><ADR><SPD>  <DIRF><TRK><SS2>  <ADR2><SND><ID1> <ID2><CHK> | Out of all the data bytes there are only a few that concern us.  <0E> tells you how many bytes are in the message.  <SLOT#> gives you the slot that the engine is in.  <STAT> indicates if the slot is active, etc (quite complicated)  <ADR> and <ADR2> give you the address of the engine.  <SPD> gives you the current speed of the engine.  <DIRF> gives you the direction and function information.  For information on the other data bytes refer to the Loconet Personal Documentation. | | | |
| Null Move | <BA><slotX><slotX> <CHK> | slotX contains the slot number that the Throttle or Locobuffer are setting to active/in use. | | | |
| Slot Speed message | <A0><SLOT#><SPD> <CHK> | <SLOT#>  This data byte gives you the slot that the speed message is referencing. | | <SPD>  This gives you the current speed of the engine. | |
| Slot Direction message | <A1><SLOT#><DIRF><CHK> | <SLOT#>  This data byte gives you the slot that the direction message is referencing. | | <DIRF>  <D7, D6, D5, D4, D3, D2, D1, D0>  D7: always 0.  D6: always 0.  D5: Direction, 0 forward, 1 backwards.  D4: Light, 1 on, 0 off.  D3: Function 4, 1 on, 0 off  D2: Function 3, 1 on, 0 off  D1: Function 2, 1 on, 0 off  D0: Function 1, 1 on, 0 off | |
| Slot Sound Message | <A2><SLOT#><SND><CHK> | <SLOT#>  This data byte gives you the slot that the sound message is referencing. | | <SND>  <D7, D6, D5, D4, D3, D2, D1, D0>  D7-D4: always 0.  D3: SND4/F8  D2: SND3/F7  D1: SND2/F6  D0: SND1/F5 | |
| PM42 Message | <D0><><><><> <CHK> |  | | |  |

The information talked about here on messages is a good overview of the different types of messages that are sent over the Loconet when using the layout. For more information on the other types of messages that may be sent over the Loconet refer to the “Loconet Personal Use Edition 1.0 SPECIFICATION” Document.

### 4.2.1 Loconet Message Paths

The Loconet handles a lot of different messages that are sent from several different places. Here we will examine each event that can happen on the layout and trace what is sent over the Loconet. This will give you a better understanding of how the Loconet works and where each message is coming from and going to. All messages are sent over the Loconet. When a Throttle is mentioned, it is implied that it can either be the DT402D Throttle, UT4D Throttle or Computer unless stated otherwise. For the computer messages, they reach the LocoNet via the Locobuffer.

* User turns on/off the track power
  + The Throttle sends a Power message to the DCS200.
  + The DCS200 then either turns the track power on or off.
* User selects an Engine number on one of the Throttles.
  + The Throttle sends a Locomotive request message to the DCS200. The two data bytes contain the engine number being requested for use.
  + The DCS200 responds with a Slot Data Message. This message contains the slot number that the engine was put in or is currently in.
  + If the slot status byte shows the slot to be COMMON, IDLE, or NEW, the Throttle sends a Null Move message containing the slot number from the slot status message. The Throttle then stores the slot number for future use.
  + If the slot status byte shows the slot to be IN\_USE or UP-CONSISTED, then the Throttle should not use that engine.
* User changes direction of engine
  + The Throttle sends a Slot Direction message to the DCS200. It uses the slot number stored during engine registration.
  + The DCS200 then looks at the slot number in the message and sends a message over the track to the engine number stored in that slot.

* User changes speed of engine
  + The Throttle sends a Slot Speed message to the DCS200. It uses the slot number stored during engine registration.
  + The DCS200 then looks at the slot number in the message and sends a message over the track to the engine number stored in that slot.

* User turns on/off the light or functions 1-4
  + The Throttle sends a Slot Direction message to the DCS200. It uses the slot number stored during engine registration.
  + The DCS200 then looks at the slot number in the message and sends a message over the track to the engine number stored in that slot.

* User turns on/off functions 5-8
  + The Throttle sends a Slot Sound message to the DCS200. It uses the slot number stored during engine registration.
  + The DCS200 then looks at the slot number in the message and sends a message over the track to the engine number stored in that slot.
* User throws/closes a switch (UT4D cannot do this)
  + The Throttle sends a Set Turnout message over the Loconet. This message contains the switch address and which state (thrown/closed) the switch needs to be set to.
  + The DS-64 that contains that switch address receives the message. It then sets the switch according to the information in the message
* A Tortoise completes moving in either the closed or thrown position (UT4D does not get this information)
  + The DS-64 sends a Turnout Completed moving message. This message contains the switch number and the current state the switch is in, either closed or thrown.
  + The message is sent out over the Loconet. Any device that needs the information can access the message.
    - Throttle – updates screen information
    - DS-64 – May change another switches based on information contained in message. (currently not used on layout)
    - Tower Controller – May change something connected to it. (currently not used on layout)
* A sensor is tripped
  + The Tower Controller sends 2 Sensor message. It sends a low-high/high-low messages. These message contains the sensor number, if it is high or low and if it is the first/second in the pair of sensors.
  + The message is sent out over the Loconet. Any device that needs the information can access the message.
    - Locobuffer – Sends message to computer for software to use it.
    - Tower Controller – May change something connected to it. (currently not used on layout)
* A short occurs on the track or a train enters a reversing section
  + The PM42 sends a PM42 message. This message contains the PM42 address and information on the 4 sections of the PM42.
  + The message is sent out over the Loconet. Any device that needs that information can access the message.
    - Locobuffer – Sends message to computer for software to use it.

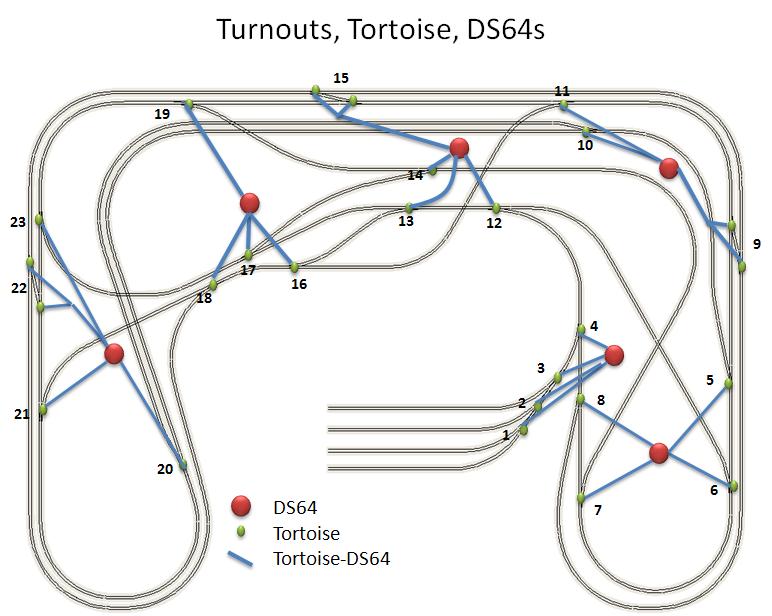
# 5. Turnouts

Turnouts were covered briefly before, giving enough information so that you could run trains on the layout. Here we will go into more detail on exactly how the turnouts work. There are 26 turnouts located on the layout. Turnouts are what allow the train to take different paths around the layout. Of the 26 turnouts, 6 of them are in 3 sets of pairs; they are called crossovers. This allows the train to switch from one parallel track to another.

All of the switches are controlled from underneath the layout by Tortoise slow motion switch machines. The Tortoises are in turn wired to the DS-64s which are then connected to the Loconet.

## 5.1. Turnouts on layout

This diagram shows where the switches are located and also how they are wired to the DS-64s.



(Diagram taken from reference 4)

Each set of four switches are wired to one DS-64, wiring will be discussed shortly. The crossover switches are wired together and to one input on the DS-64. This allows the crossover switches to be closed or thrown by way of just one switch address instead of two.

## 5.2. Tortoises

The Tortoises are what provides the motion to the switches. Each switch has its own Tortoise connected to it. Inside each Tortoise are two single throw double pole switches. When a Tortoise moves to either the thrown or closed position the internal switches are thrown or closed. This is how the DS64 knows that the Tortoise has completed moving.

The Tortoises are connected to each switch by way of music wire. This wire is put through a hole in the center of each switch and then connected to the throwing arm of the Tortoise underneath the layout.

The Tortoises have 8 output connections. Two are used for power and then two sets of three are used for each internal switch. The Tortoises are wired to the DS-64 by using a card edge connector. This prevents having to solder the wires directly to the Tortoise and helps to not cause internal damage to the Tortoise.

The two internal switches of the Tortoise are switched when the Tortoise is closed or thrown. Each switch has two inputs and one common connection. The common connection allows you to wire switches up to a central connection on hardware boards.

One of the internal switches is used to signal the DS64 that the switch has finished moving. The other internal switch can be used for controlling an LED or other external device.

The following are pictures of the Tortoise and the edge connector.



Throwing Arm

Music Wire



Card edge connector

(Picture taken from reference 4)

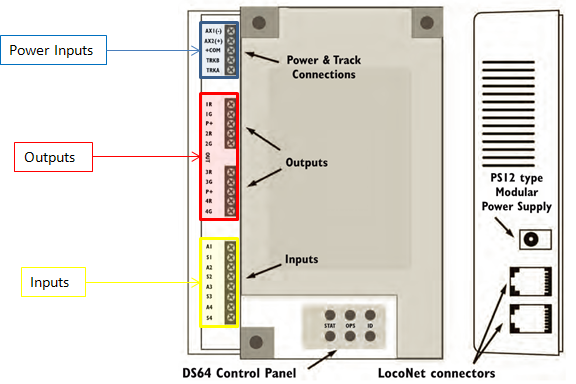
## 5.3. DS-64s



(Picture taken from reference 4)

The DS-64, pictured above, is the control unit for the Tortoises on the layout. It tells the Tortoises when to close and throw and also sends signals over the Loconet once a Tortoise has closed or thrown fully.

There are four sets of inputs and outputs on the DS-64 and also power inputs and Loconet connections. These are labeled below.



The power block has two types of power inputs. Track power and external power. In the lab we use an external power source. Track power will be used when the DS-64 needs to be programmed, explained later. The power block also provides the central connection, used for the internal switches of the Tortoise.

The two output blocks, each provide two sets of outputs, for a total of 4 outputs. Each output block has a common output for a different kind of switch machine. We don’t use the common output. Each output block gets wired up to two Tortoises or for the pairs three Tortoises.

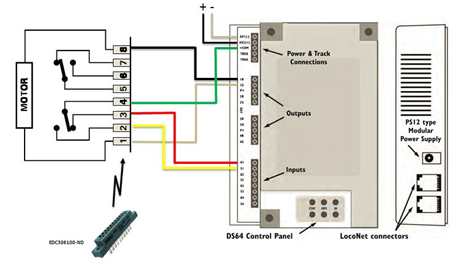
The input block is where all of the Tortoises internal switch outputs get wired up to. Each of the Tortoises internal switches get wired up to one pair of inputs. In the case of the pairs of Tortoises, each pair is wired in series and then wired to the input of the DS64.

You should never really need to worry about the DS-64. They are all programmed so that each switch is numbered exactly as the diagram has them numbered.

If for some reason the DS-64 is not functioning correctly you can reset and reprogram it. For that refer to Section 8.2.

## 5.4.    Wiring

The wiring of the Tortoises and DS-64s is pretty straight forward.



(Diagram taken from reference 4)

Pins 1 and 8 on the Tortoise provide the power to Close or to Throw it. Pin 1 is colored white on the layout and gray in the diagram and Pin 8 is colored black. These two wires go back to one of the 4 outputs on the DS-64.

Within each Tortoise there are two Single Throw Double Pole switches. We use one of these switches to inform the DS-64 what state the Tortoise is in.

Pin 4 is the common part of the switch and is colored green. This is connected with the other common wires of the other switches of the DS-64 and connected to the common plug on the DS-64.

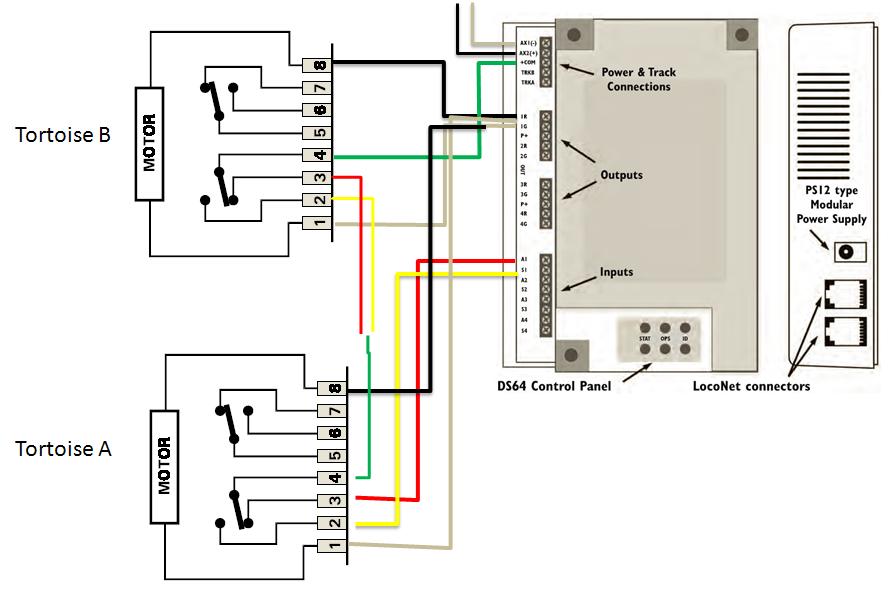
Pin 2 and 3 are the pins of the internal Tortoise switch output. Pin 2 is colored yellow and Pin 3 is colored red. All of the connections are connected to the DS64 corresponding to the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tortoise | White wire | Black wire | Yellow wire | Red wire |
| 1 | 1R | 1G | S1 | A1 |
| 2 | 2R | 2G | S2 | A2 |
| 3 | 3R | 3G | S3 | A3 |
| 4 | 4R | 4G | S4 | A4 |

Pins 5, 6, and 7 are not used. They are the other internal switch of the Tortoise and aren’t needed for use on the layout but could be used for signaling or connection to another external device.

For the pairs of Tortoises, the power inputs are wired in opposite of each other. Meaning one power input of the DS64 has a black from one Tortoise and a white from the other and vice versa for the other power connections.

The internal switches of the Tortoise pairs are wired a little differently. The common wire from Tortoise “B” is wired to the DS64. Then the common wire from Tortoise “A” is wired to both of the internal switch outputs from Tortoise “B”. Next the internal switch outputs from Tortoise “A” are wired to the inputs on the DS64. The diagram below shows the wiring.



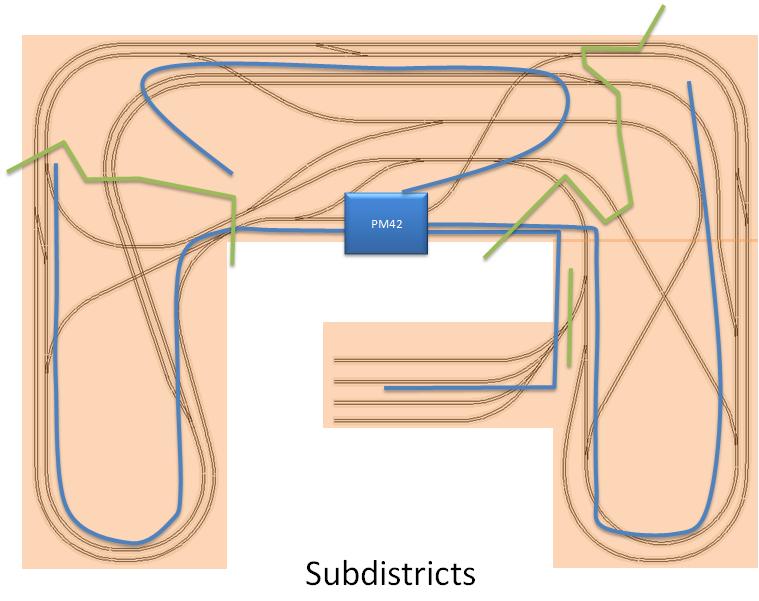
# 6. Power Control of Track

The power control on the layout is pretty straight forward. Once the power from the DCS200 is sent to the PM42 it is distributed out over the track over four common pairs of wires. We will discuss the wiring that comes from the PM42, the reversing sections and the PM42.

## 6.1. Wiring

The wiring of the track power has the least amount of wire of all of the electronics on the layout. There is a black and white wire for each side of the track. This wire comes from the PM-42 set up for power districts. The green lines on the diagram show where the layout is divided into the power districts. The blue lines show the approximate route the wire is running for each power district. The wires that are represented by the blue lines are called bus lines.

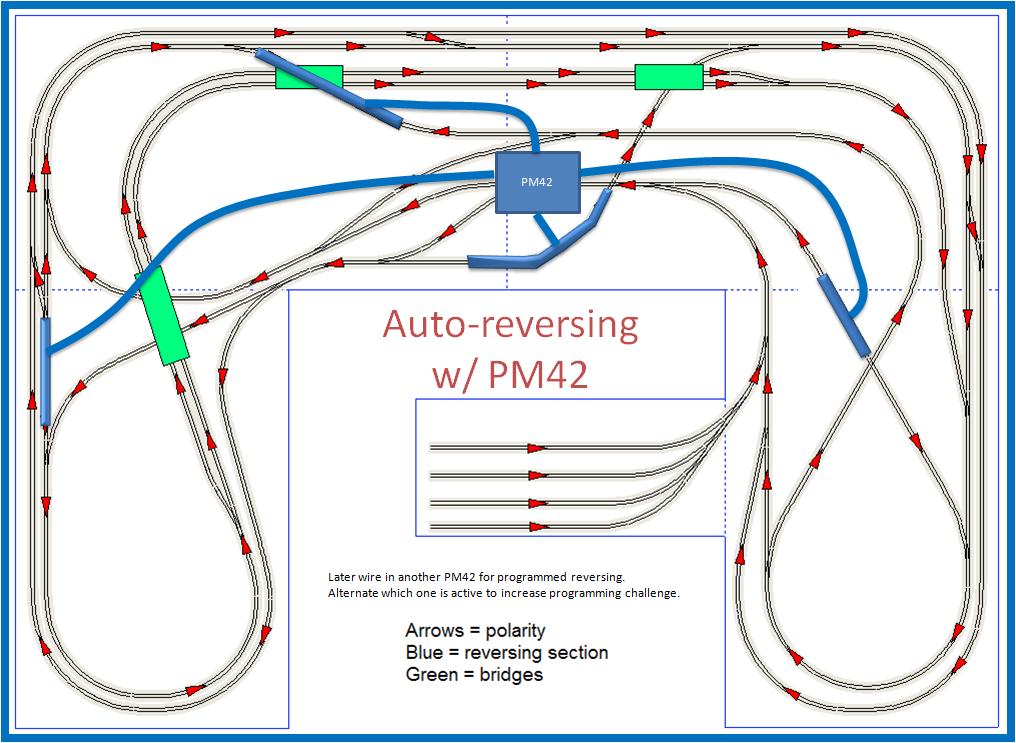
From the bus lines there are feeder wires going up to the track. The feeder wires are then soldered directly to the track. There is a constant power sent from the PM42 to the track. The power that the PM42 uses is received from the DCS200 which also has the square wave imposed on it.



(Diagram taken from reference 4)

## 6.2. Reversing sections

The second PM-42 on the layout provides the reversing sections with its logic. This PM-42 is wired and setup to detect when a train enters a reversing section and ensures the train can cross that section without shorting out that entire power district. The reversing sections are colored blue on the following diagram.



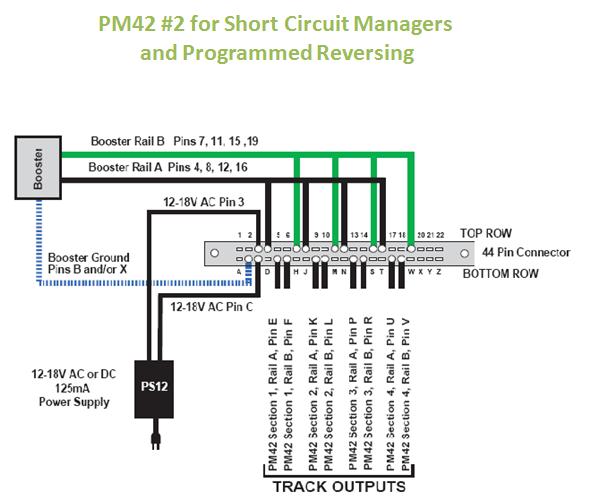
(Diagram taken from reference 4)

## 6.3.    Boards



(Picture taken from reference 4)

The PM-42s are located on the central drop down board in the middle of the layout. It will be hard to tell which PM-42 is wired for the power districts or the reversing sections. There shouldn’t be any need to fix the wiring or change the settings on the PM-42s. If for some reason you must reprogram the PM-42s please refer to section 8.4. Included below are the wiring diagrams for each board.



(Diagram taken from reference 4)



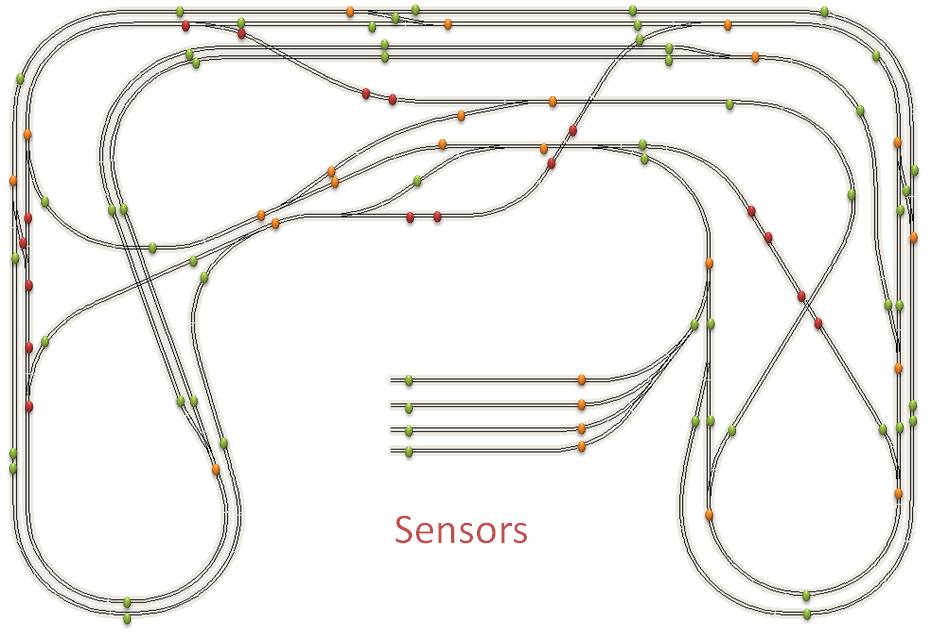
(Diagram taken from reference 4)

# 7. Sensors

The sensors are how the layout tells the system where a train is on the layout. Sensors involve three components: reed switches, breakout boards and TC-64s. We will discuss all three components and also how they are wired together.

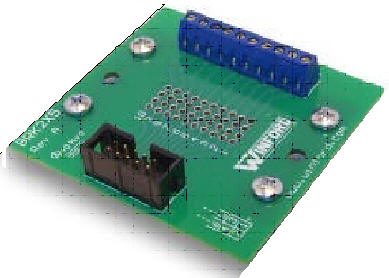
## 7.1. Reed switches

The sensors located on the top of the layout are reed switches. They are located at numerous locations on the layout, as pictured below. The sensors are the red, green and orange dots on the track plan.



The reed switches are tripped when a magnet passes by them. There is a magnet on the front of each engine and the rear of each caboose. When a magnet goes over a sensor, without stopping, the sensor sends two messages. When a magnet goes over a sensor and stops on it, it sends out one message. Then when the magnet leaves the sensor it sends the second message. Both messages sent are the same. This can be useful for determining if a train has stopped on top of a sensor.

## 7.2. Breakout board



(Picture taken from reference 4)

The breakout board, pictured above, is the connection point from the reed switches to the TC64s. The breakout boards are used to make the connection easier to make, otherwise the wiring would be very complicated.

## 7.3. TC-64s and Sensor Numbers

The TC-64s are how the sensors communicate with the Loconet. Each TC-64 takes its input from the breakout boards which are connected to the reed sensors.

Each TC-64 allows for up to 64 connections. The connections are broken up into 8 blocks of 8. Each block of 8 connections can be programmed to different sequential addresses. Meaning that block one can have addresses 1-8, block two could have addresses 11-18. You have the option of choosing the addresses that you want.

To determine the sequence of addresses for a certain block being used as input, the TC-64 uses a Base Address in the range from 0 to 1023 for each block. The base address of the block is the number that is programmed for that block, i.e. 1, 9, 17.

When using a block as input, each block uses just 4 Loconet addresses but still serves 8 inputs. When the message is sent across the Loconet you will see that the sensor messages for inputs one and two of a block have the same address. There is although a certain bit in the message that is different for the two inputs. This is how you determine which input it is. This holds true for inputs 3/4, 5/6, 7/8 of each block also.

To determine the correct sensor number, you can take the number that is reported in the sensor message and multiply it by 2. If the I bit of the sensor message is 0 then you will subtract 1 from the multiplied sensor number.

The JMRI software is already designed to handle this and will display the correct information for you. Software that you develop will need to be able to handle this and be able to determine which input the system is sending a message for.

When you are using a TC-64 block as output, which currently is not used on the layout, the addresses are figured a different way. You still use a Base Address but you just have to add one to that Base Address to get the first address in the sequence of 8. Again, the TC-64 uses just 4 Loconet addresses to serve the 8 outputs.

For information on how to program addresses and configure blocks for either input/ouput, refer to section 8.3.



(Picture taken from reference 4)



## 7.4. Wiring

Overall the wiring of the sensors is pretty straight forward. Each reed switch has two wires leading from it. There is a green (common) wire and then a colored wire. The colored wire is Red for even numbered sensor and Yellow for odd numbered sensor. The wires then lead back to the breakout board for that group. The sensors are grouped according to the following diagram.



(Diagram taken from reference 4)

Each group has at most eight sensors and the layout is divided into two parts, one for each of the two TC-64s used on the layout. The grouping is done in such a way to limit the amount of wiring needed underneath the layout.

Each of the wires from the sensor and the common are connected to the breakout board. We have one breakout for each block of sensors on the layout. The breakout board is then connected to the TC-64 by way of a ribbon cable.

# 8. Programming

In this section we will briefly cover the different ways of programming the trains and hardware on the layout. This will get you started on how to program the different items but will then refer you to that items manual for more detail.

## 8.1. Engines

The engines are programmed using the DT402D throttle. First, plug the throttle into the UR92 transceiver. If the throttle’s battery is installed, you can then unplug the throttle.

When programming engines, the engines use what are called ***CV values***. CVs, ***configuration variables***, are used to set up the many different operating characteristics for each decoder installed in each engine. Each CV controls a different characteristic of the decoder. Following is a list of some CVs and what they control on the decoder.

|  |  |
| --- | --- |
| **CV number** | **What it controls** |
| CV01, CV17, CV18 | Controls the engine address. These are never programmed directly but programmed using the ADDR screen, which displays first when entering program mode on the throttle. |
| CV33, CV34 | Sets what Function 0 controls in the forward direction, CV33, and the reverse direction, CV34. |
| CV35-CV46 | Sets what Functions 1 – 12 controls. |
| CV49-CV54 | Sets the Effect on Functions 0 – 3. |
| CV113-CV116 | Sets the Effect on Functions 3 – 6. |

With the engines that we have on the layout, three of them are programmed one way and one is programmed a different way. Once you get the engines into program mode, they are all programmed the same way. The yellow engine is the one that is put into program mode a little different.

### 8.1.1 Yellow Engine Programming

To get the yellow engine into program mode, you will set it onto the program track. This is the short section of track located next to the yard, and it isn’t connected to the rest of the layout track. You will then press the program button on the Throttle to enter the programming mode. This will automatically put you into Paged mode programming. This mode is used when programming engines on the programming track.

Throttle program key.jpg

(From reference 2)

Now you are ready to program the engine. You will use the left throttle knob to select the CV you want to program and the right throttle knob to select the value to program. You will want to refer to the *Digitrax Decoder Manual*. There is a link to this manual at the end of this document. It describes the different CV’s and what they do.

### 8.1.2 Atlas Engine Programming

The other engines, made by Atlas, that we have on the layout are special in the way that they must be programmed. In order to program an engine you will need to have the track power turned off, refer to Section 2.1, and make sure that no other engines are on the track.

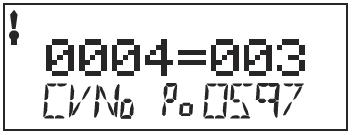
Next hold a magnet above the engine and turn the track power back on. You have now put the engine into program mode.

Select the engine that you want to program on either the left or right side of the throttle. Refer to Section 2.2 on how to select an engine. After selecting the engine, you will press the program button on the throttle to enter the programming mode.

Throttle program key.jpg

(From reference 2)

The next step is to start the Operations Mode Programming. You may have to press the program button a few times to get into the correct mode. You should see a Po in the middle of the screen once you are in Operations Mode Programming.



(From reference 2)

Now you are ready to program the engine. You will use the left throttle knob to select the CV you want to program and the right throttle knob to select the value to program. You will want to refer to the [engine manual](HO_Train_Master_Sound.pdf) included in the back of this manual for information on the different CV’s and what they do.

## 8.2. DS-64s

Programming the DS-64s is pretty straight forward. You will first want to make sure that you have the power turned off to the DS-64 that you want to program. This is best done by unplugging that DS-64s power supply from one of the power strips. You will then remove one of the power wires from the DS-64. Next you will take two jumper wires and plug them into the two track power slots on the DS-64 and connect them to the track on the layout. Make sure that the track power is turned on. Refer to section 2.1 on how to turn the track power on.

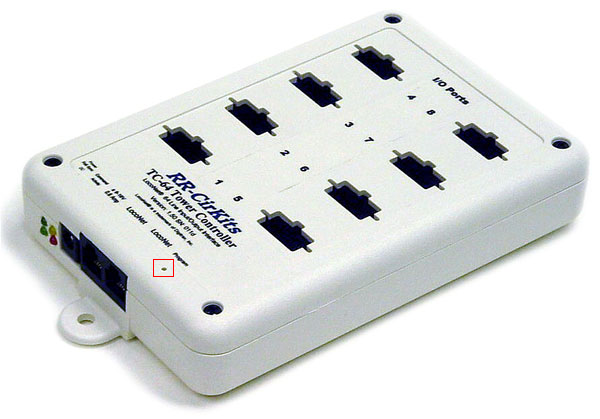
Then unplug one of the Loconet cables from the DS-64 and plug in the DT402D Throttle. Refer to the [DS-64 documentation](ds64V4.pdf) on how to finish programming. The main things that might need to be programmed are as follows, with page numbers from DS-64 documentation:

* Setting the DS-64 to operate the Tortoises, page 5
* Setting the addresses of the switches connected to the DS-64, page 8
* Changing internal Option Switches on the DS-64, page 10

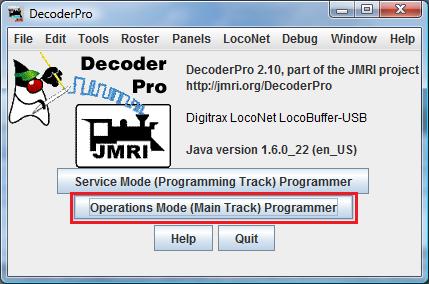
## 8.3. TC-64 Towers

To program the Tower Controllers, you will use JMRI. JMRI is already setup on computer in the lab to deal with programing of the Tower Controllers. If you are using a different computer you will need to refer to page 1 of the Tower Controller manual. You will want to make sure JMRI is running on the laptop in the lab. Refer to section [3.2](#_3.2._Starting_JMRI_1) for how to start JMRI. On the lab computer the tower controllers are labeled North/South within JRMI. The North Tower Controller is the one closest to the windows. The South Tower Controller is the one closest to the hallway.

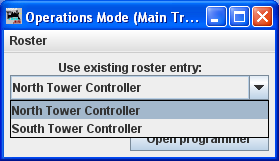
Before opening up the programmer in JMRI, you need to put the tower controller into programming mode. You will need a paper clip or something similar. Go to the tower controller that you want to program and insert the paper clip into the hole located near the Loconet connection. This is highlighted red in the picture below. Hold it down for roughly three seconds until the yellow light begins to flash. This indicates that you have put the tower controller into programming mode.



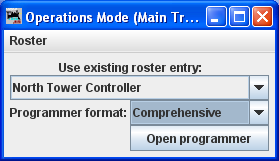
Now return to JMRI. You will want to click on the Operations Mode Programmer button on the main screen.



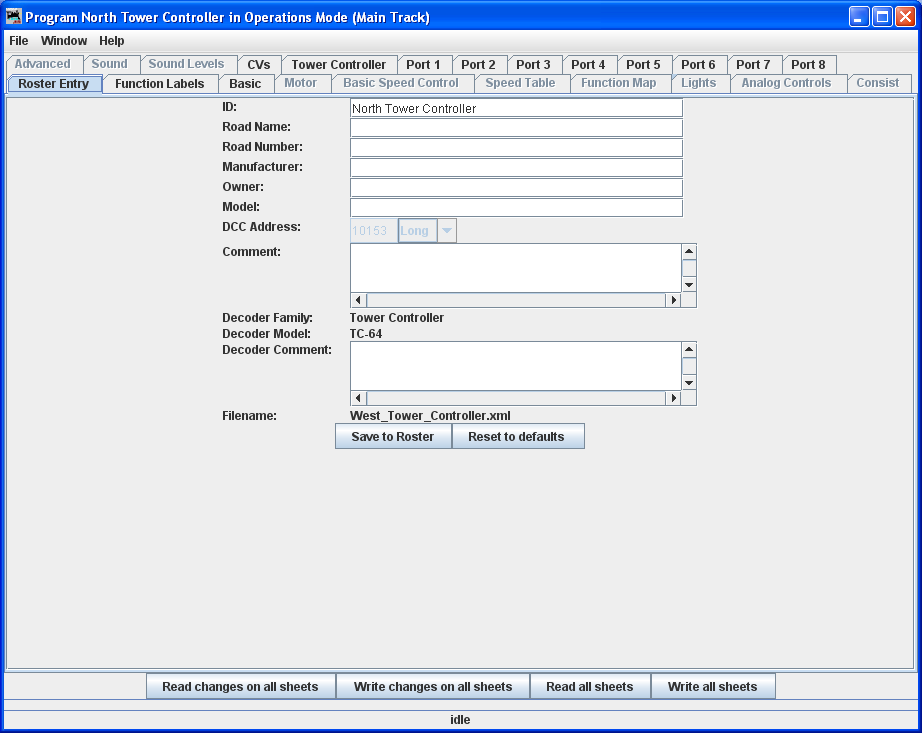
This will then open up another screen where you can select which tower (North or South) you want to program from the drop down menu.



Once you have selected the tower you want to program, make sure that Comprehensive is selected.

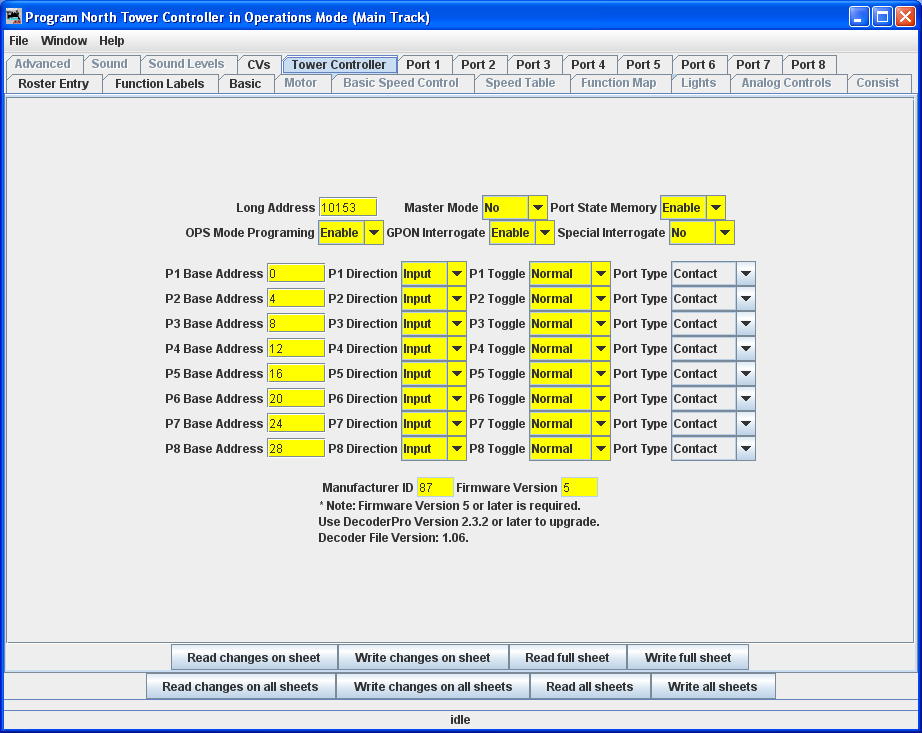


Now click Open programmer. This will bring you into the programmer for the Tower Controller.



Now you have several options/settings that can be changed. The main thing that you will most likely want to change will be the sensor addresses. This is done on the Tower Controller tab. You can refer to section 7.3 on how the addresses are chosen or page 5 of the Tower Controller manual.

On the Tower Controller tab there are mainly two columns of settings that you will deal with. The Base Addresses are used to set the addresses for each block. The Direction column is used to configure a block for input or output. The information in the first two rows at the top should not be touched. This is used to configure how the Tower Controller is programmed and doesn’t ever need to be changed. The port type column can be changed depending on the type of input or output device you are using.

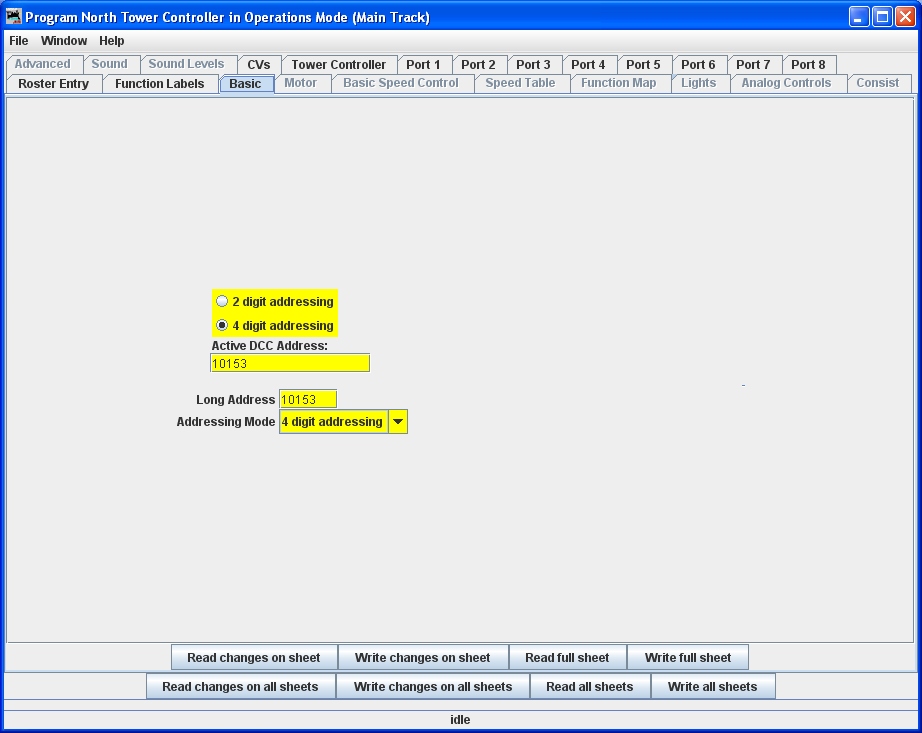


You can also change settings for each individual port on the Tower Controller using one of the Port tabs. You will want to refer to the Tower Controller manual or JMRI documentation for information on each individual setting and what it does.



This screen allows you to change the settings of each individual output/input on each port.

The last tab that may be used is the basic tab. This allows you to change the board address. Since you are using an already configured entry, the information on this screen will never be changed.



By using all of the screens mentioned you can also configure the Tower Controller to act as an output source. This feature is not currently utilized on the layout, but information on output for the Tower Controller can be found on page 5 of the Tower Controller manual.

For more information you can refer to the [Tower Controller manual](TC-64-manual.pdf).

## 8.4. Power District boards

Programming the PM42s doesn’t require anything special; you just need the DT402D Throttle with a battery installed. First you will want to make sure power is on to the system. Then unplug one of the Loconet cables and connect your Throttle. You will then press the option button on the PM42. This is the button closest to the Loconet connectors.

Then enter the switch state on the throttle by pressing the switch button. This puts the throttle into the programming mode for the PM42.

Switch Key.jpg

(From reference 2)

Next you select the Option Switch that you want to program by selecting the Switch address on the throttle. To program the Option Switch you either send a Close or Throw command, by press either the T or C button, depending on how you want to program that Option Switch.

Throttle Optn-Throw Button.jpg Throttle Cloc-Close Button.jpg

(From reference 2)

For information on the different Option Switches, refer to the [PM42 documentation](pm42manual.pdf), pages 10-12 at the end of this book.

## 8.5. DCS200

Refer to pages 42-43 in the Super Chief manual. If you did want to change the settings of the DCS200 you can change several different settings. Listed in the table below is the option switch used and what it changes.

|  |  |
| --- | --- |
| Option Switch | Effect on DCS200 |
| 13 | Changes how long an engine will stay in a slot once it is now longer in use. |
| 33 and 34 | Sets the system to restore track power to how it was when the system was turned off. |
| 44 | Enable system to have either 22 slots or 120 slots. |

For a complete list of Option Switches and their effect, refer to the [Super Chief Manual](superchiefxtra.pdf) in the back of this manual.

# 9. FAQ section

In the FAQ section, any questions you may have during operation of the train lab will be answered. When troubleshooting something, work from one down to the last item

**Switches**

* The switch doesn’t move when I send a command from the throttle or computer.
  1. Is the power on to the DS-64 that the switch is connected?
  2. Try another switch connected to the same DS-64. If that works, try pulling the card edge connector and re-attaching it.
  3. Pull all card edge connectors for the switches attached to the DS-64
  4. Replace the Tortoise
     + The lights on the DS-64 should be flashing
  5. Are the Loconet cables connected to the DS-64?
  6. Are all the wires connected to the DS-64?
  7. Try plugging in the throttle to the Loconet.
     + If that works, you need to change the battery in the throttle
  8. Reprogram the DS-64 by referring to Section [8.2](#_8.2.___1).
     + You will want to make sure you first reset the DS-64 by programming CV 7 to closed
  9. Try replacing the Tortoise
  10. If none of this works, you may try contacting Digitrax or referring to the [DS-64 manual](ds64V4.pdf)
* The Tortoise moves but the switch does not move
  1. Make sure that the switch isn’t jammed
  2. Make sure that the rod connecting the switch to the Tortoise is connected correctly

**Engines**

* The Engine runs very bad it stops and starts all of the time
  1. Clean the engines wheels using a paper towel and rubbing alcohol.
  2. Clean the track
* The Engine doesn’t respond to any commands
  1. Have you turned the track power on? Refer to section 2.1.
  2. Have you selected the correct engine? Refer to section [2.2](#_2.2._Selecting_an_1).
  3. Is the Engine on the track?
  4. You may need to clean the Engines wheels and the track
  5. Try resetting the engine. Refer to section [8.1](#_8.1.___1). Setting CV 8 to 8 and then reprogramming the engine. You will need to select engine 3 after you reset it before you are able to reprogram it to the correct engine number.
  6. Contact Atlas Model Railroad
* The Engine moves but there is no sound
  1. Make sure that function 8 is not turned on. Refer to section [2.4](#_2.4._Controlling_functions_1).
  2. Contact Atlas Model Railroad
* None of the functions work on the engine but it moves
  1. You may need to clean the Engines wheels and the track
  2. Try resetting the engine. Refer to section [8.1](#_8.1.___1). Setting CV 8 to 8 and then reprogramming the engine. You will need to select engine 3 after you reset it, before you are able to reprogram it to the correct engine number.
  3. Contact Atlas Model Railroad

**Sensors/Tower Controller**

* Sensor doesn’t report when tripped
  1. Is the sensor broken? If, yes, replace it with a new one
  2. Are the wires connected to the break out board?
  3. Is the break out board connected to the tower?
  4. Does the tower have power? Some of the lights should be on.
  5. Are the Loconet cables connected to the tower controller?
  6. Reset the tower controller. Refer to section [8.3](#_8.3.__).
* The sensors aren’t reporting as sensors but something else
  1. Refer to section [8.3](#_8.3.__) and make sure they are set to send sensor messages.
* The sensors aren’t reporting the correct numbers
  1. Refer to section [8.3](#_8.3.__) on how to change the sensor numbers.

**DCS200**

* The track power won’t turn on but there is power to it
  1. Make sure the mode switch on the DCS200 is set to run.
  2. Refer to section 2.1 on how to turn the track power on.
* When I turn the track power on I get a strange message
  1. Turn power off to the entire system and start over.

**General**

* Trains run really rough
  1. Clean all track.
  2. Clean all wheels on Engines.

# Appendix

# A.      Standards

## A.1.   DS-64s

All Option Switches are set to closed expect for the following: 01, 15, 21.

## A.3.   Towers

Refer to the screens in the JRMI software for standards.

## A.4.   Power District boards

### A.4.1 Power District set-up

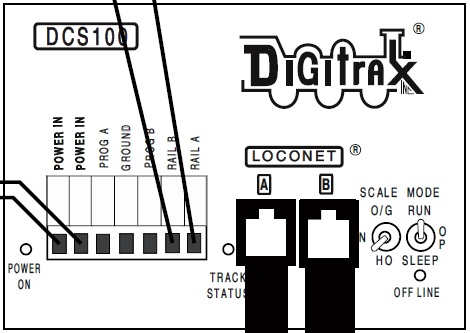
The following option switches are set to thrown: 1, 2, 3, 5, 11, 13, 19, 21, 27, 29.

### A.4.2 Reversing Section set-up

The following option switches are set to closed: 3, 6, 11, 14, 19, 22, 27, 30.

The following option switches are set to thrown: 5, 13, 21, 29.

## A.5.   Main unit



(Diagram taken from reference 1)

This is a diagram of the main command station unit on the layout. It is a Digitrax DCS 100 Command Station. The following are the standard settings for this unit:

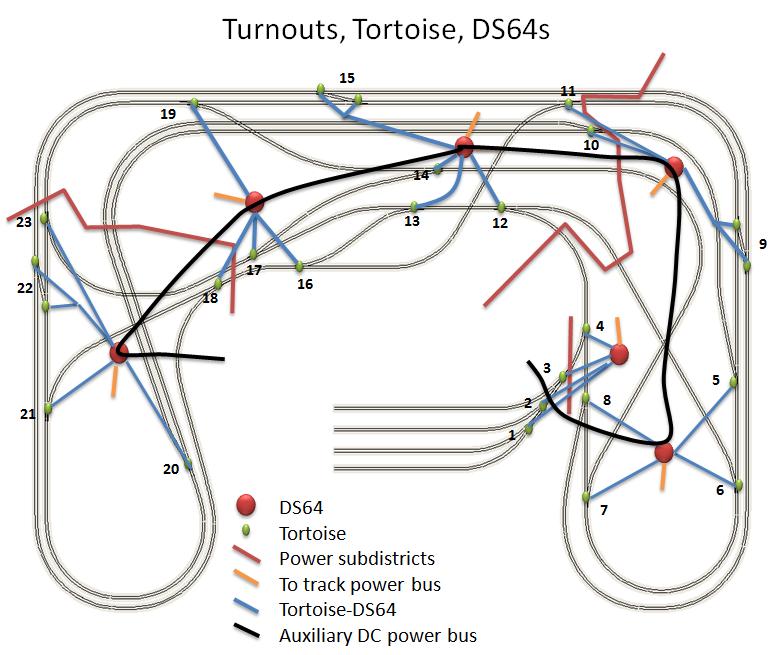
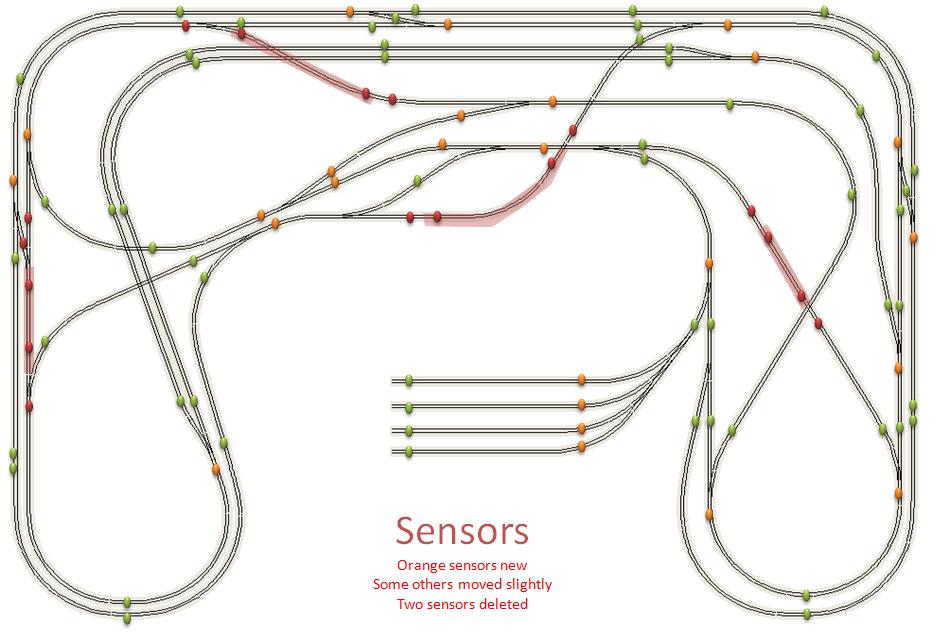
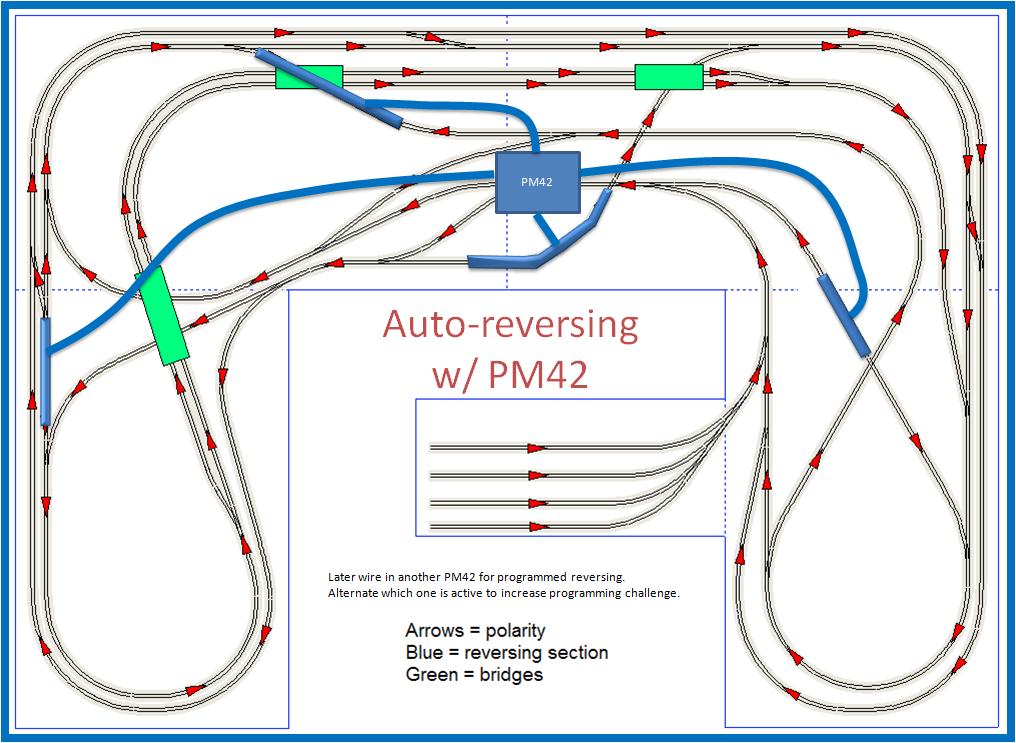
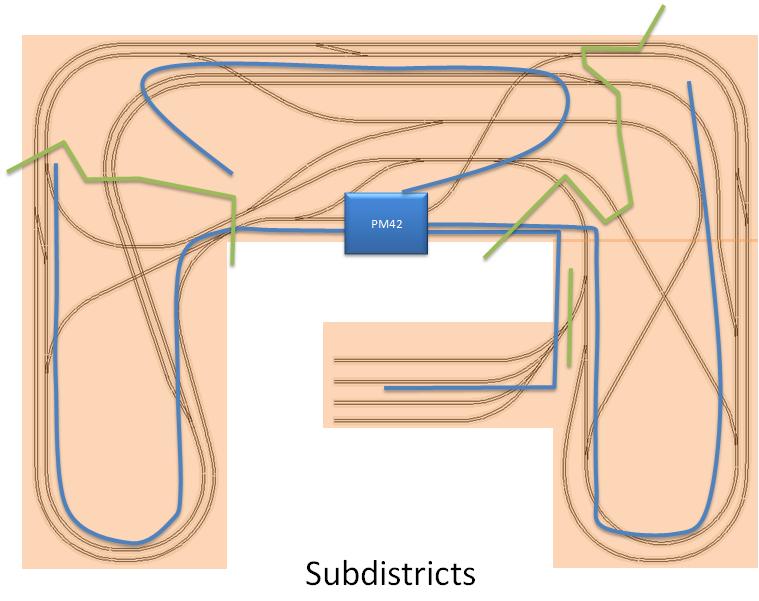
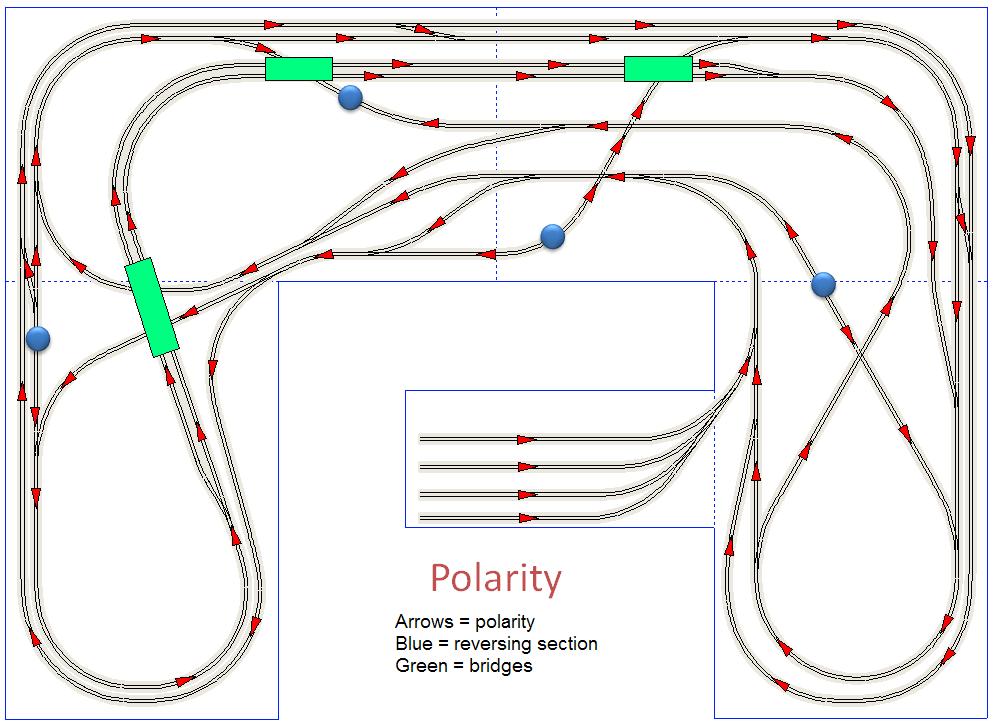
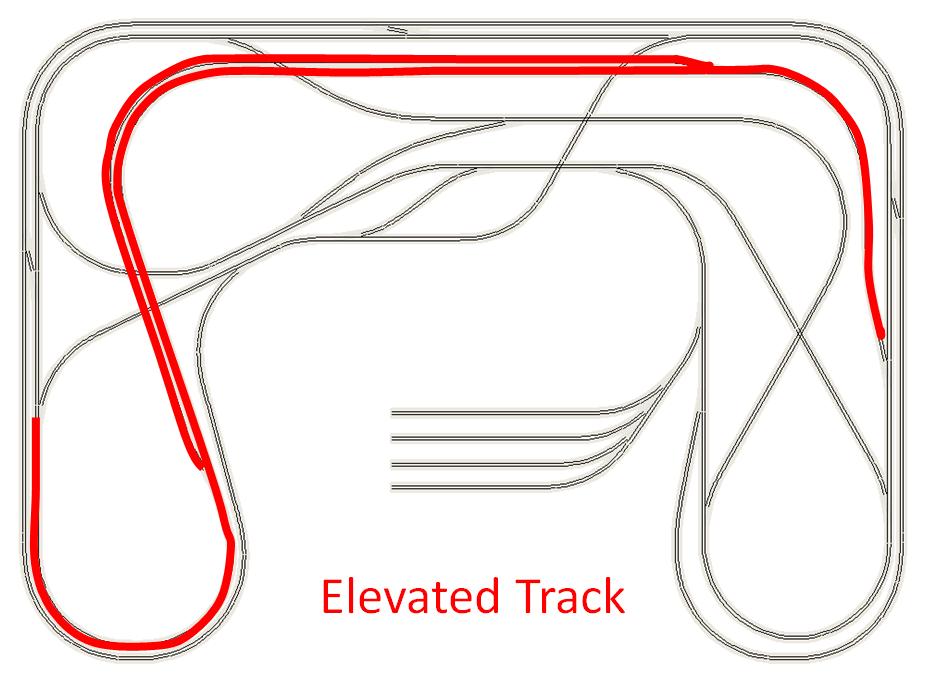
* Scale switch: HO Settings (**SHOULD NEVER BE CHANGED**)
* Mode: Run
* Both Power In plugs should have wire going to the power supply
* Rail A and Rail B should be connected to both of the PM-42s
* Prog A and Prog B should be connected to the program track, located directly above on the layout
* Ground should be connected to both of the PM-42s

The Mode switch should always be left in Run unless the unit is being programmed. Information on programming the DCS 100 can be found in section 8.2.

# B.      Diagrams

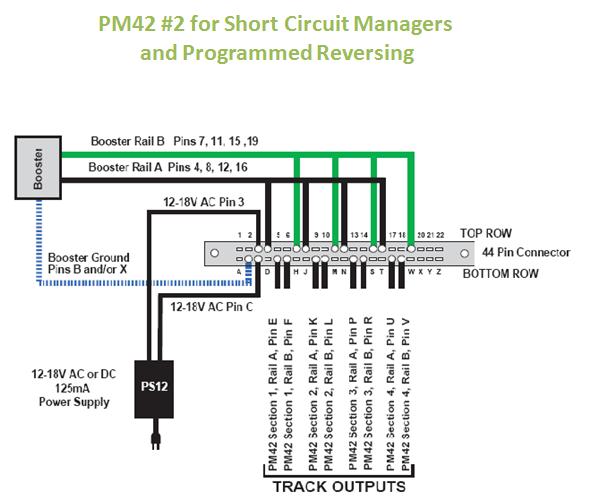
## B.1.   Trackplans

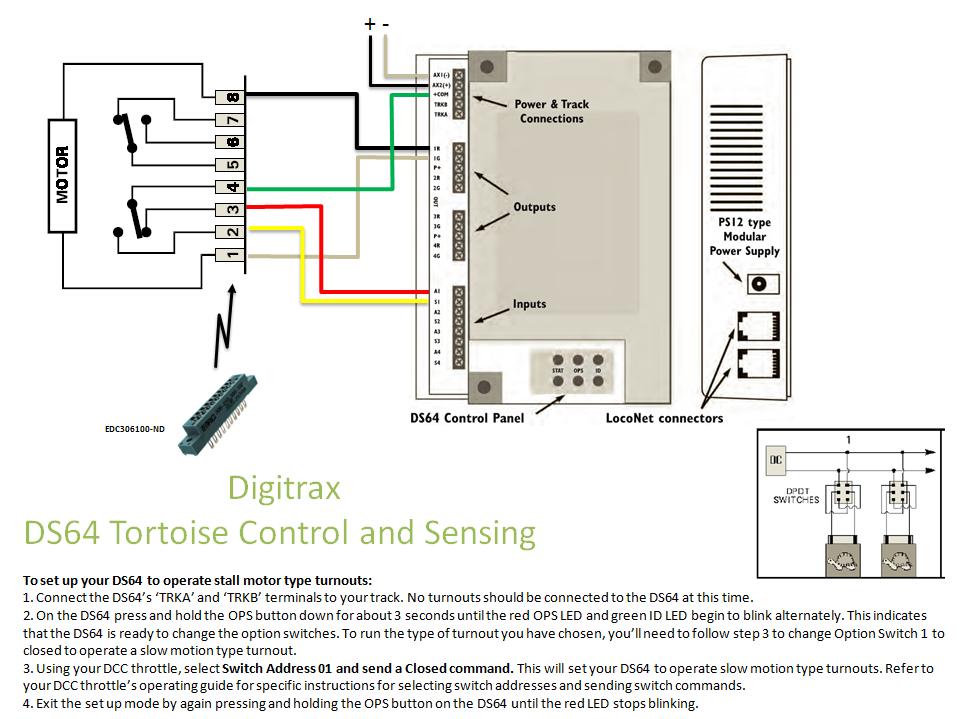
(All Diagrams taken from reference 4)



## B.2.   Wiring Diagram

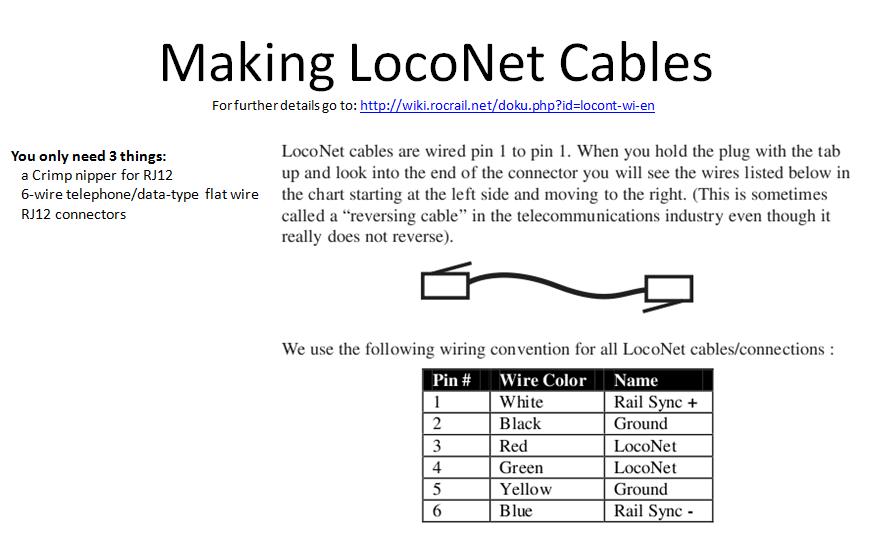
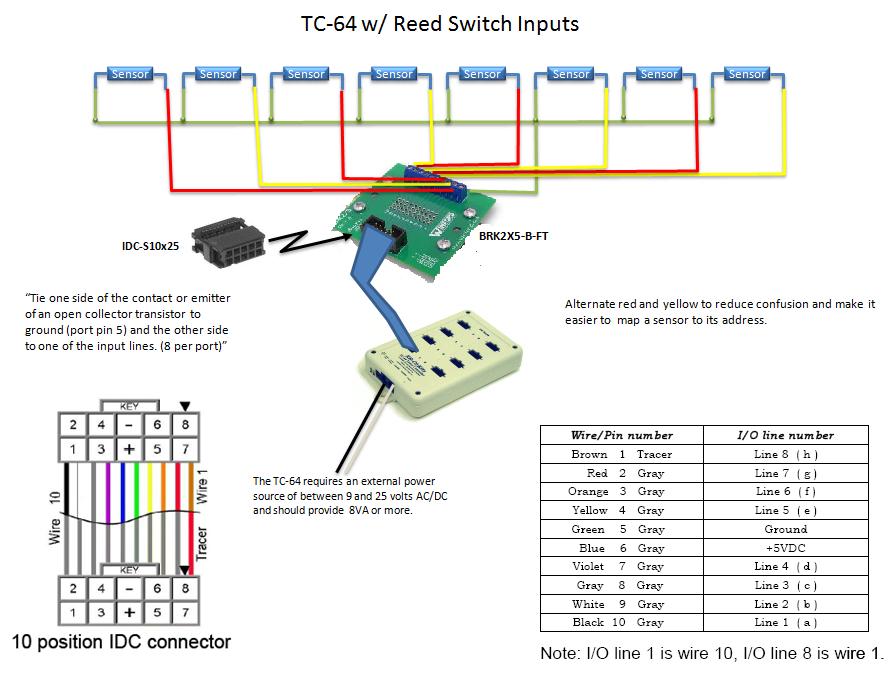
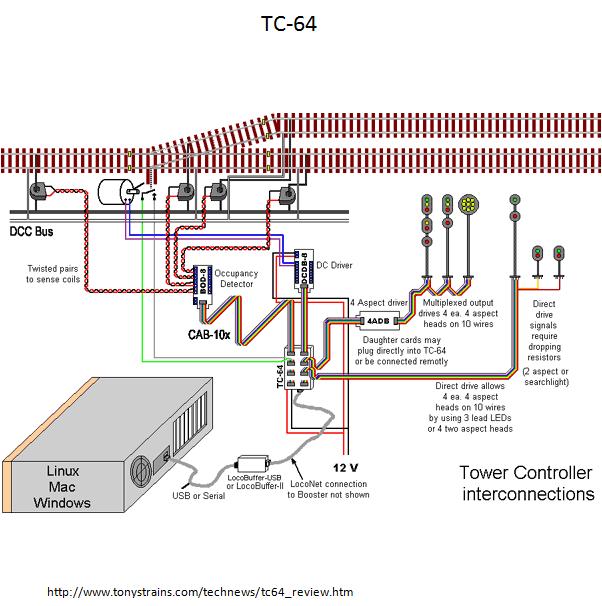
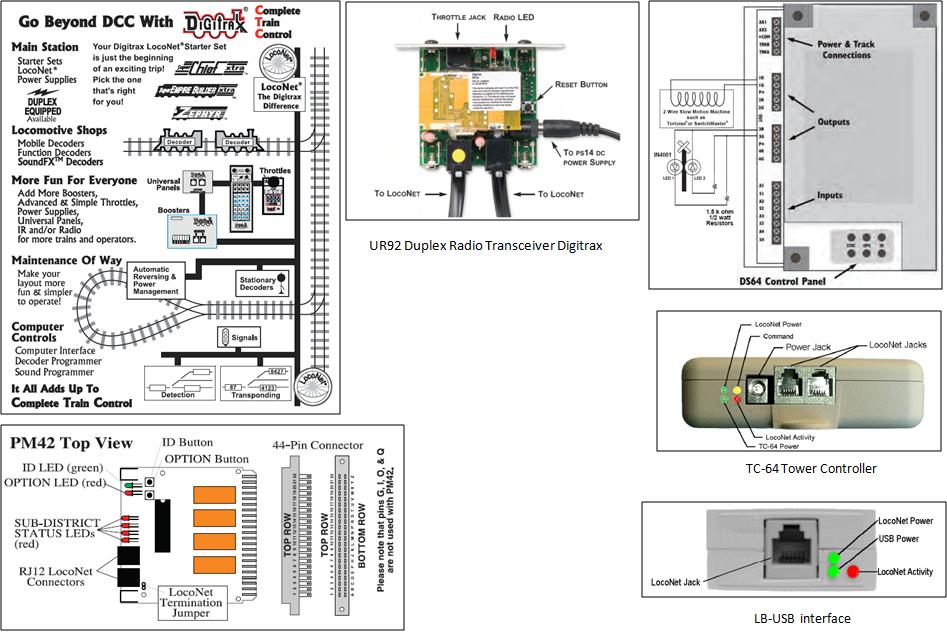
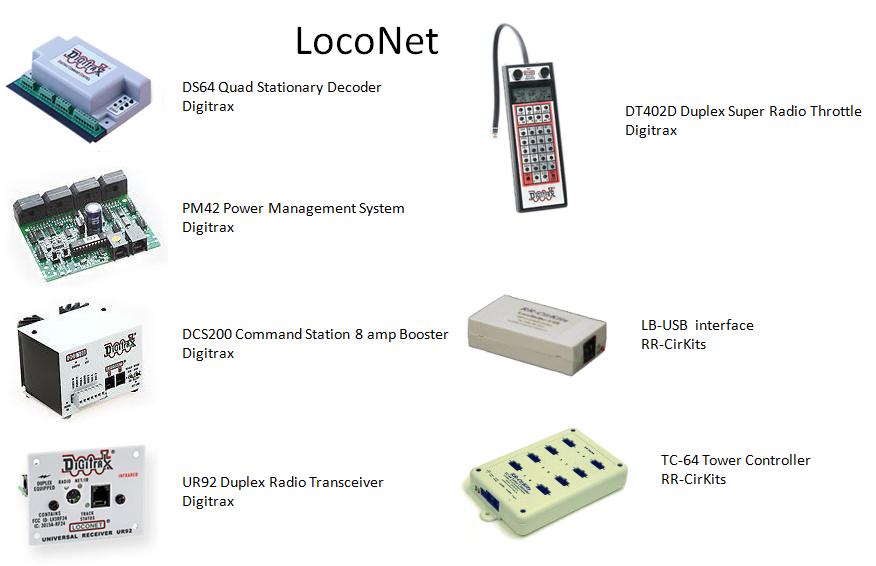
(All Diagrams taken from reference 4)





## B.3.   DCC Stuff

(All Diagrams taken from reference 4)



## B.4. Throttles

(All Pictures taken from www.digitrax.com)

### B.6.1 DT402D



### B.6.2 UT4D



# C.      Trains

(All Diagrams taken from <http://faculty.cs.wwu.edu/martin/ModelTrain/My%20Plan/Trains/trains.htm>

## C.1.   Engines

ATL-10000272-0.jpg

### C.1.2 Function Numbers

## C.2.   Cars

ATL-1930-0.jpgATL-20000154-0.jpgATL-20000197-0.jpgATL-20000202-0.jpgATL-20000336-0.jpgATL-20000340-0.gifATL-20000341-0.jpg

## C.3.  Cabooses



## C.4.   Track Cleaning Car



# D. Manuals

[Digitrax Super Chief Manual](http://www.digitrax.com/ftp/superchiefxtra.pdf)

[Digitrax DT402 Throttle](http://www.digitrax.com/prd_dt402.php)

[Digitrax UT4D](http://www.digitrax.com/ftp/UT4D.pdf)

[Digitrax UR92](http://www.digitrax.com/ftp/UR92.pdf)

[Digitrax PM42](http://www.digitrax.com/ftp/pm42manual.pdf)

[Digitrax DS-64](http://www.digitrax.com/ftp/ds64V4.pdf)

[Digitrax Decoder Manual](http://www.digitrax.com/ftp/Decoder%20Manual.pdf)

[RR-Cirkit TC-64Tower Controller](http://www.rr-cirkits.com/manuals/TC-64-manual.pdf)

[RR-Cirkit LocoBuffer USB](http://www.rr-cirkits.com/locobuffer-usb/LB-usb-flyer.pdf)

[Atlas Engine Manual](http://www.atlasrr.com/pdf/HO_Train_Master_Sound.pdf)

# References

1. Digitrax Super Chief xtra manual
2. Digitrax DT402 throttle manual
3. Digitrax UT4D throttle manual
4. Martin Osbornes Power Point slides