

Jacobsen Declaration Exhibit D

Decoder Commander® Software

KAM Industries

KAM Industries
2373 NW 185th Ave
sales@kamind.com
Hillsboro, Or 97124

Phone 503 291 1221
email:
Fax 503 291 1221

Decoder Commander® software

Configuration and User Manual



Decoder Commander® Software

Copyright © 1991 - 2005 KAM Industries. Decoder Commander, Classic Panel, Computer Dispatcher, IFeedback, IEngComIfc, Kamind, Decoder Commander, LocoCe, LocoWinCe, Train Controls, Train Priority, Train Server, Train Tools, Video Speed are registered trademarks of KAM Industries. Products covered under Patent 6065406, 6267061, 6270040, 6530329, 6460467, 6494408, 6676089, Ger 29923834.2, GB 2353228, CAN and other US and international patents pending. All rights reserved..

*Other copyrights owned by their respective owners. Use with permission from Cisco Corporation, Dell, Lenz, and Hewlet Packard Corporation and Lenz Elektronik, GmbH. KAM Industries is a division of KAMIND Associates, Inc.

For more information on KAM Industries products, please contact

KAM Industries.
2373 NW 185th Ave., #416
Hillsboro, Oregon 97124

EMAIL: sales@kamind.com
WEB: <http://www.kamind.com>
FAX #: (503) 291-1221

Other names and brands are the property of their respective owners.

Revision: 10/4/2005

Copyright © 1991 -2005 KAM Industries. All rights reserved

Decoder Commander® Software

Table of Contents

OVERVIEW OF TRAIN SERVER® SOFTWARE	4
TRAIN SERVER APIS	5
DECODER COMMANDER AND PROGRAMMING.....	5
GETTING STARTED	6
INSTALLING DECODER COMMANDER® SOFTWARE	6
CONFIGURING TRAIN SERVER	6
START TRAIN SERVER® MANAGER	6
CONFIGURE COMMAND STATIONS.....	6
DECODER COMMANDER BASIC OPERATIONS	8
START TRAIN SERVER® MANAGER	8
START DECODER COMMANDER®	9
CONNECT DECODER COMMANDER TO TRAIN SERVER.....	9
DECODER COMMANDER CONSOLE	10
USING DECODER COMMANDER THE FIRST TIME	10
TERMINOLOGY	11
PROGRAMMING A DECODER WITH THE WIZARD	13
CONFIGURATION OPTIONS AND TOOLS	18
DECODER TOOLS	18
ADDRESS SETTING	20
DRIVING ATTRIBUTES.....	21
ANALOG FUNCTIONS	22
FX OPTIONS.....	23
SPEED CURVE	24
FUNCTION MATRIX	25
VARIABLE OVERVIEW	26
DEFINE SINGLE VARIABLE.....	27
PROGRAMMING INFORMATION WINDOWS	27
STATUS MENUS	27
PROGRAMMER WINDOW.....	28
REPORT VIEW WINDOW	28
TRAIN SERVER MESSAGE WINDOW	28
CREATING A TEMPLATE	29
TEMPLATE VERIFICATION TOOL	29
USING THE VERIFICATION TOOL.....	30
TEMPLATE FORMAT	30
TEMPLATE ELEMENTS.....	30
Template Capabilities (templateList).....	30
Function Mapping (fmlist)	32
CV Definitions (cvlist)	32
TROUBLESHOOTING	34
HOW TO OBTAIN THE TCP/IP ADDRESS OF A COMPUTER	34
CANNOT CONNECT TO TRAIN SERVER	34

Overview of Train Server® Software

KAM has been producing professional model railroad software since 1991 and is the leading supplier of Professional DCC software for the model railroad industry. All of KAM's software is scale independent, internet enabled, and supports all popular command stations.

Train Server is a software program (middleware) that handles the day to day management of the command station and coordinates the information flow between other Train Server applications. Train Server Architecture is a multi-user architecture that is unique in the model railroad industry in managing the model railroad communications interface. Most other DCC control applications are simple single user, single command station programs which do not allow information to be exchanged between simultaneously users on a real time basis.

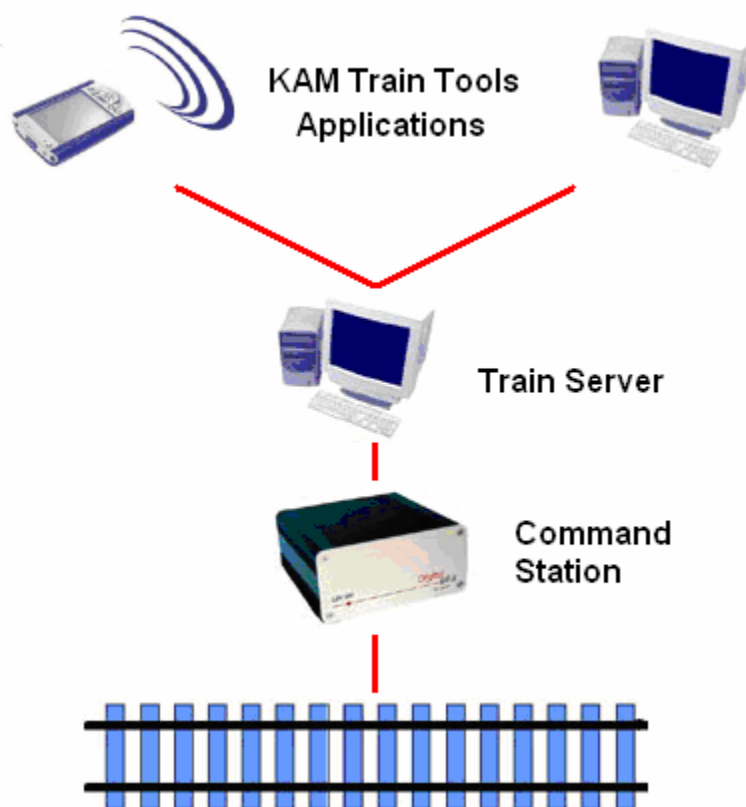


Figure 1: Train Server with Train Tools Applications

While the electrical signals sent through the layout to control DCC hardware is an industry standard, how application programs communicate to the command stations that generate these signals is vendor specific and often model specific. Train Server supports all popular command stations so that a single version of a DCC control application can be written that will work with all popular command stations now and into the future.

Decoder Commander® Software

All KAM Train Tools applications communicate with Train Server in order to control the layout. When using a single PC, you can run these applications on the same PC as Train Server. (Microsoft's loopback driver must be installed if there is no network hardware installed. See website for details.) In a networked PC environment, only one PC needs to be running Train Server. Other computers running applications designed for use with Train Server can operate the layout remotely over the network through the "server" PC running Train Server.

Train Server APIs

KAM supports and encourages open application programming interfaces (APIs). We publish all interface information on our web site, as well as on our CD-ROMS.

Train Server software supports 3rd party vendor, end user, and KAM developed applications. While Train Server is designed to run on a Windows PC, it supports a well published, open programming interface using XML data over TCP/IP networks so applications may be developed for any operating system and programming language that supports TCP/IP networking. Train Server also supports an open Windows COM/ActiveX-based API for applications that run on the same computer as Train Server. In addition, network aware applications can easily be built with a KAM code library that greatly simplifies DCC application development in a Windows environment using any .NET programming language. For more information on interfacing your own applications to Train Server, please review the documentation shipped on your CD-ROM, or published on our web site.

Decoder Commander and Programming

KAM has two different programming philosophies for programming decoders, a simple programmer (Engine Commander® software) and a fully function do everything programmer (Decoder Commander).

Engine Commander implements a basic programming capability for programming decoders. Engine Commander allows you to read and write CV's, program decoder address and speed curve. There are no templates to determine what decoder model you are running, nor support for advance decoder programming or file operations. Engine commander programming is a simple programmer, allowing you to read and write your decoder.

Decoder Commander is a serious decoder programmer that allows you to do everything that Engine Commander does plus configure advance decoder functions, clone decoders, customize speed curves, configure advance function mapping and create customize decoder templates. Decoder Commander allows you full access to the manufacture specific decoder information so you may customize the decoder as you see fit.

Getting Started

Installing Decoder Commander® Software

To install Decoder Commander insert the CD-ROM and follow the installation instructions supplied with your product. The default installation will only install a time limited demonstration version of the programs. To install licensed products, you will need to use your supplied serial number. This serial number is located on your CD-ROM or may be purchased on line from our web site. Your serial number will only install those applications that you have purchased. To install other demonstration applications you will need to use the custom installation option during your installation.

Applications designed to work with Train Server communicate only with Train Server to control the layout. For a single user, you can run these applications on the same PC as Train Server, but when multiple users would like to control the layout (or in the case of Loco CE you want to be able to control the layout wirelessly with a Pocket PC), only the PC with the command stations attached requires a running instance of Train Server.

If you wish to install only Decoder Commander on additional networked PCs, insert the CD-ROM and select the custom installation. Select Decoder Commander for installation and disable (uncheck) the other programs.

Train Server runs as a Windows process, not as an application with a user interface. Train Server Manager is an application used to start, stop, and configure Train Server.

Configuring Train Server

Make sure your command station is connected to the PC that has Train Server installed and make sure the command station has power.

Start Train Server® Manager

The first time you run any Train Server application you need to configure the command stations you want to use.

- Run the Train Server Manager. It is located in your start menu in the "Train Tools" program group.

Configure Command Stations

- Click the "Interface Wizard" button in the Train Server Manager application. This wizard will guide you through configuring your command station. (See Figure 3)

Decoder Commander® Software

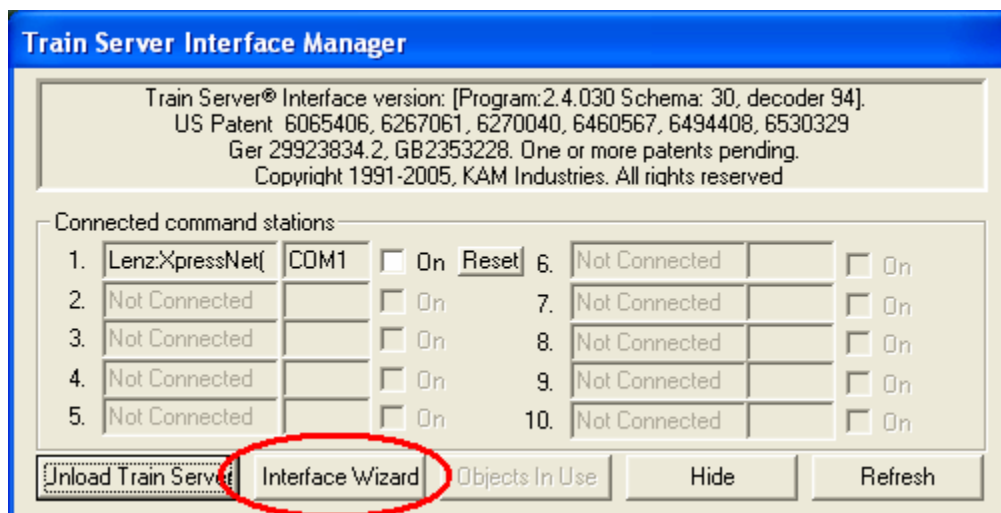


Figure 2: Train Server Manager

- The first configuration setting is the "**Logical Port**". (See Figure 4) Generally a logical port refers to where the command station is plugged into the PC. (which serial port, such as COM1.) Logical ports are a useful way to organize these physical connections rather than the using the Windows-specific device names. Your computer might have a COM1 but the next port might be COM3. Or your command station might be connected through a USB port. Logical Ports allow multiple command stations to be configured with simple number assignments, regardless of their physical PC port connection. They also allow you to move a command station to a different physical port without needing to reconfigure any applications other than Train Server since the logical port will still refer to the same command station, even on a different physical port.

In most cases, the default values shown in the wizard will be fine. The most important configuration setting is the specific brand and model of command station connected. (See Figure 5) The remaining screens are set to the factory default values that ship with the command station selected.

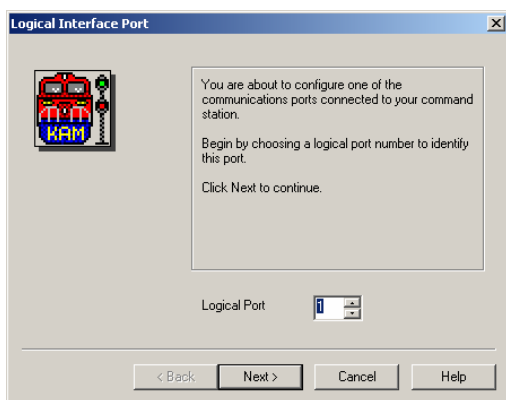


Figure 3: Port selection screen

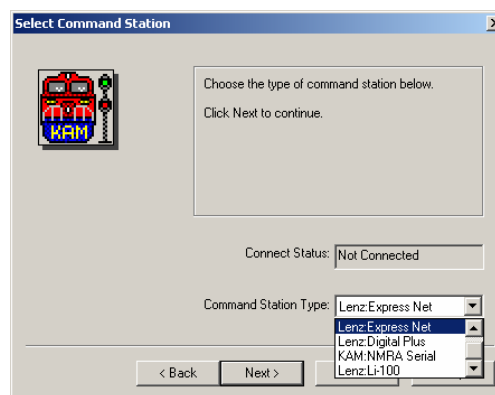


Figure 4: Command station selection

Decoder Commander® Software

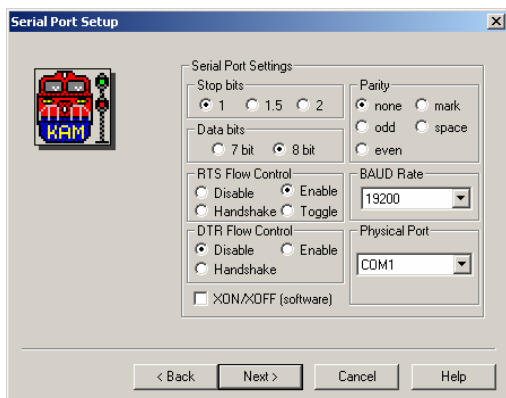


Figure 5: Default serial port selection

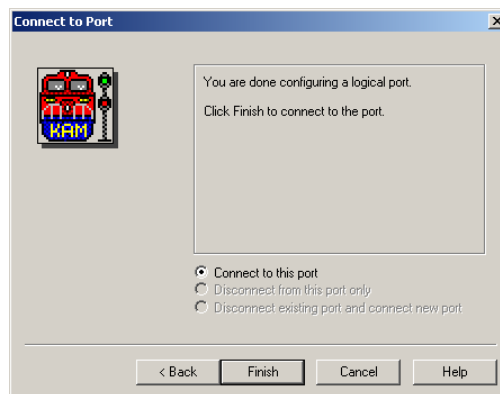


Figure 6: Final selection screen

At this point you are ready to run any Train Server application. When Train Server manager is running, the program acts as a communications gateway for Train Server local and network applications.

Below are descriptions for additional features support by Train Server Manager. (See Figure 3)

- **Unload Train Server:** This is used to terminate Train Server from running on your computer. Train Server Manager will stop Train Server's processes (enginterface.exe and locoevents.exe). It will not stop any application using Train Server, so it is recommended you shut down all Train Server applications before unloading Train Server.
- **On:** This allows you to turn on the power to the layout (checked), or turn it off (unchecked).
- **Reset:** This is used to send a hardware reset to the command station. In some command stations this will purge the command station's current settings and restore the factory default values.
- **Refresh:** This is used to update the status of the connected command stations in Train Server Manager.
- **Hide:** Reduces Train Server Manager to an icon in the system tray.

Decoder Commander Basic Operations

Start Train Server® Manager

If Train Server Manager is not yet running on the PC running Train Server, you must start Train Server Manager in order to start Train Server. If you have not yet configured any command stations, please read the previous section and complete this step first.

- Start Train Server Manger. It is located in your start menu in the "Train Tools" program group.

Train Server will remember the last command station configured, so you only need

Decoder Commander® Software

to enter the command station information one time.

TIP: Train Server must be running before you continue. If you have not selected a command station in Train Server, Layout commander will not control the layout.

Start Decoder Commander®

- Start Decoder Commander. It is located in your start menu in the "Train Tools" program group.

When decoder commander starts for the first time, there is a delay while the software searches your disk for the latest templates. This may take a few minutes depending on the size of your disk.

Decoder Commander uses XML data templates that conform to the manufactures based decoder definition. See the section on templates later in this manual.

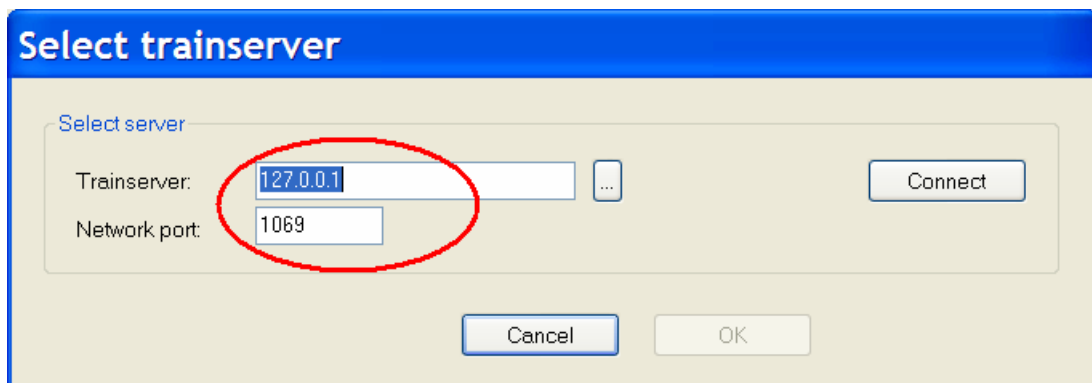


Figure 7: Decoder Commander main window

Connect Decoder Commander to Train Server

- Enter the TCP/IP address or the computer's network name that is running Train Server.

TCP/IP is the networking protocol used for Microsoft networks and the TCP/IP address is how each computer on the network is uniquely identified.

If your are running Decoder Commander on a different computer than the PC running Train Server you must enter the TCP/IP address or computer name of the PC running Train Server. See the Troubleshooting section of this manual for help in obtaining the TCP/IP address of another computer.

The first time Decoder Commander is run, it will use the TCP/IP address of the computer it is running on as the default value for the server address.

After Decoder Commander is run the first time, the default server address displayed will be the last address entered. Some networks use dynamically assigned TCP/IP addresses (DHCP) which can cause the address of a PC to change

Decoder Commander® Software

when rebooted which will cause the previously used address to be incorrect. See the troubleshooting section for more information.

- Enter the TCP/IP port number that Train Server is using.
This will always be port 1069 unless Train Server was intentionally reconfigured.
- Click the "Connect" button.

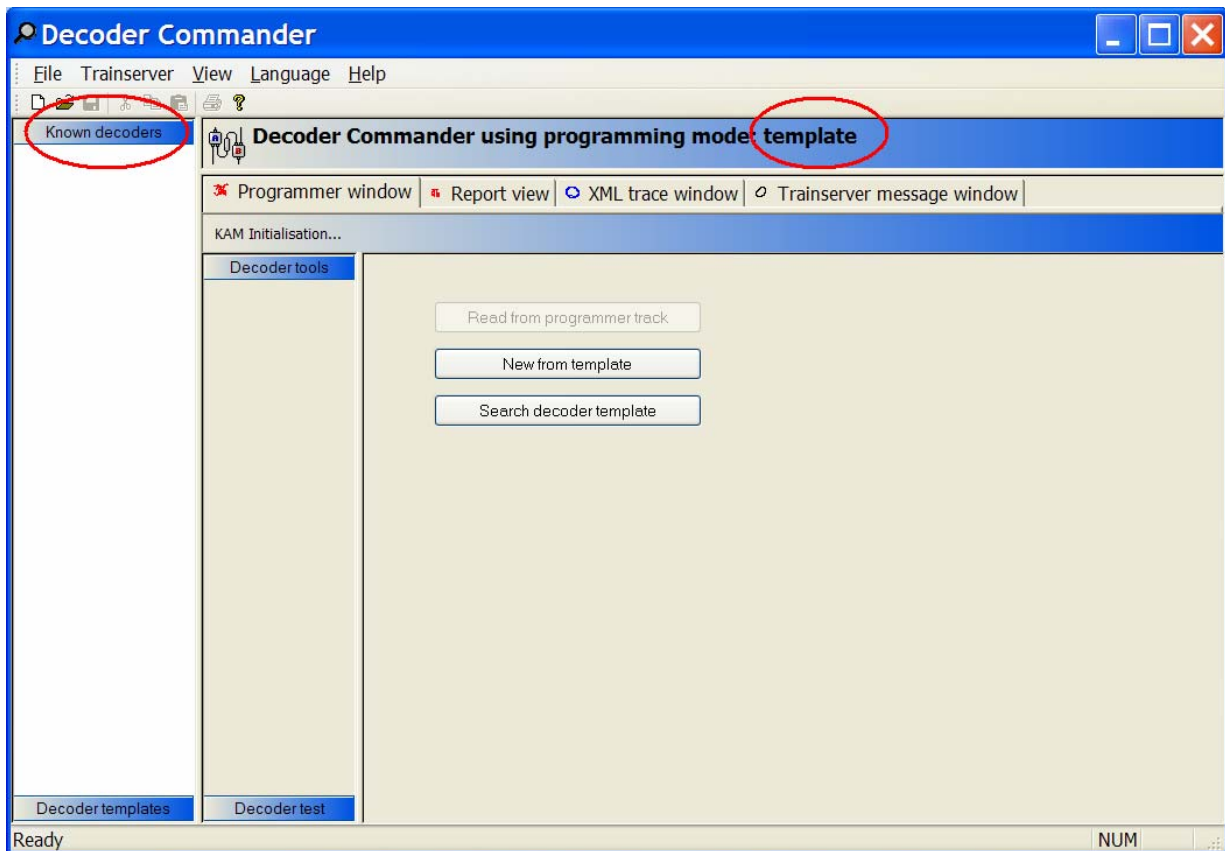
Decoder Commander Console

Figure 8: Decoder Commander startup screen

After you have selected the connect button, Decoder Commander, connects to Train Server and loads any know decoder that you have programmed, and sets up the programming mode form your last session.

Decoder Commander uses decoder templates, that represents the manufactures model railroad decoders. These templates are used as input to create a new decoder that represents on of the locomotives on your layout.

You can create your own templates, but you will need to run those template samples through the KAM's template verification tool to build a template that will work with Decoder Commander. See "Templates" later in this manual.

Using Decoder Commander the first time

Decoder Commander® Software**Terminology**

Decoder Commander is a decoder programmer tool designed for programming model railroad decoders. Before you begin to program your model railroad decoders, there are a few terms that describe the operation of Decoder Commander and the capabilities of the software.

Decoder	A hardware component in your model train that controls the operation of the model train. Decoders have a control variable specification that defines the operations characteristic of the decoder and how the model train will operate.
Control Variable	Control Variables or CV's contain configuration information on the operation of the decoder. The CV's are defined by the National Model Railroad Organization (NMRA), and conform to a standard that all model train manufactures support.
NMRA Standard	The nmra Digital Computer Control(DCC) standard is located at www.nmra.org . The individual decoder specifications are located at www.dcc.info . Manufactures who support the NMRA standard, display the NMRA conformance logo on their products.
Programmer	A software tool design to program the configuration variables on a mobile decoder. KAM's Decoder Commander is a software decoder programming tool.
Templates	All decoders have unique characteristics. KAM has created a set of Decoder Templates that has these characteristics in a XML configuration file.
Template Converter	A tool supplied by KAM that will process a users template and electronic build a decoder template that can be used by decoder commander. The template converter is described later in this manual.
Command Station	A hardware component used to control your model railroad layout. Command Stations support two different programming capabilities, Operation mode and service mode. Service mode requires a programming track for operation.
Programming	A command to change the CV in the decoder. Programming is either a "read" (service mode) or "write" (operations and service mode) for the desired CV.

Decoder Commander® Software

Read	<p>A decoder may only be read in service mode. This requires that the decoder is placed on the programming track.</p> <p>Decoder Commander will use the template programming mode unless the user overrides the operation.</p>
Write	<p>A decoder may be written in operations mode (on the mainline) or in service mode (on the programming track).</p> <p>Decoder Commander will use the template programming mode unless the user overrides the operation.</p>
Operations mode	<p>Operations mode is a programming mode that is used when the model train is operating on the layout.</p> <p>A decoder address can not be programmed in operations mode.</p> <p>Operations Mode only support an unverified write operation to the decoder. This is because the decoder is on the mainline operations track.</p>
Decoder Address	<p>A decoder address is an address that the computer uses to recognize the locomotive on the model railroad layout.</p> <p>Decoder addresses do not need to be unique. If two locomotives have the same address and are on the layout, then the both locomotives will respond to any computer command.</p>
Programming Track	<p>A set of track attached to a command station where you can program the decoder in service mode.</p>
Service Mode	<p>Service mode is a programming mode that uses the programming track on a command station. In service mode, there are three different programming operations support by the command station. These are page, direct and register.</p> <p>Service mode is used to program the decoder CV values and addresses. Service mode uses different types programming formats to program a decoder.</p> <p>Service Mode supports both "read" and "write" operations to the decoder. This is because the decoder is on the programming track.</p>
Programming formats	<p>There are three different types of programming formats used to program a decoder in service mode. These programming formats are page, direct and register.</p> <p>Different manufactures use differ formats in there command stations for programming. This is the same as different</p>

Decoder Commander® Software

decoders use different formats for programming as well.

Decoder Commander uses the decoder template programming value, and selects the correct programming mode for the decoder.

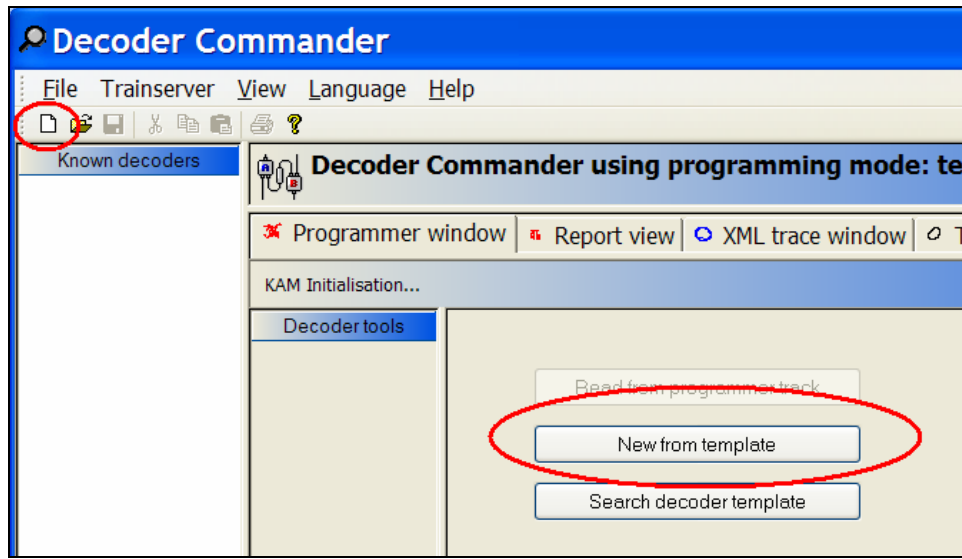
Programming a Decoder with the wizard

Figure 9: Programming your first decoder

Figure 10: Automatic Train control on a large layout

- Decoder Commander uses programming wizard when you program a decoder for the first time.

TIP: Before you start the decoder template wizard, create a directory called "My Decoders". Locate this directory in the "My Documents" directory. This will allow you to save your decoders to disk.

Decoder Commander uses both the NMRA service mode programming and the operations mode programming.

When you program a locomotive decoder for the first time, you will need to place the locomotive decoder on the service mode programming track to program the decoder CV's. After the decoder is programmed, you can program the decoder on the mainline.

- Select either the "new" file or "new from template" to begin the programming process for your locomotive decoder.

Decoder Commander® Software

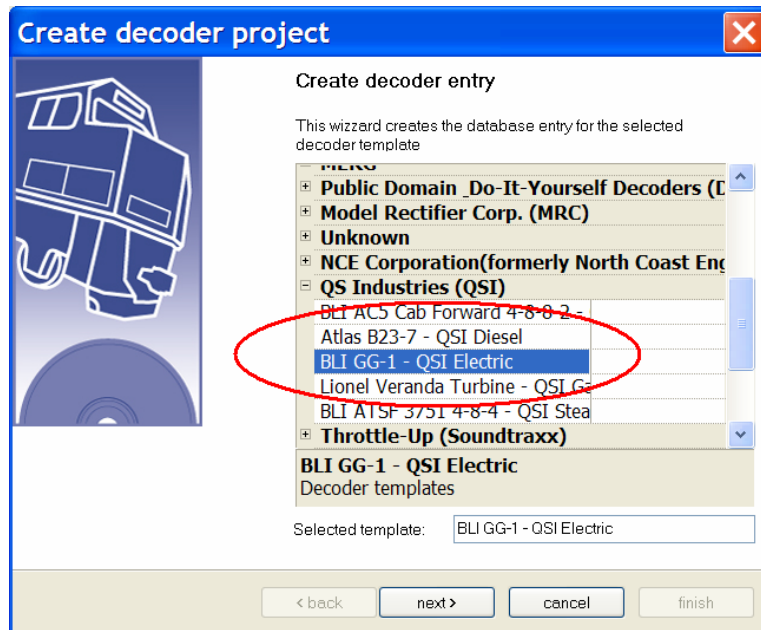


Figure 11: Running Decoder Commander Programming wizard

- After starting the wizard, select the decoder template that matches your decoder as close as possible. In our example, we have selected an QSI decoder from Baltimore Limited.

All of the information collected from the decoder will be stored in a readable data file. You can use the Decoder Commander to print out the decoder information, or any other third party tool that can read the Decoder Commander data format.

- Select next and enter information about the decoder project

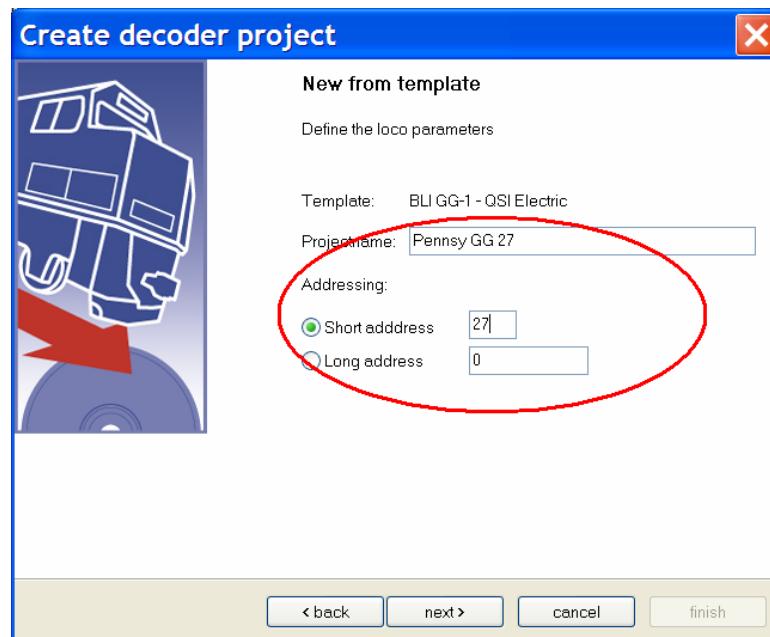


Figure 12: Entering project information for your decoder

Decoder Commander® Software

TIP: When you name a decoder project, name it with a value that is descriptive of the locomotive. This will make it easier to maintain the decoder over time.

- Select next after you have entered the decoder address and project name
- Select a directory where you would like to save the decoder information that you have created. You can save this information anywhere on your disk. Create a directory in My Documents called "My Decoders".

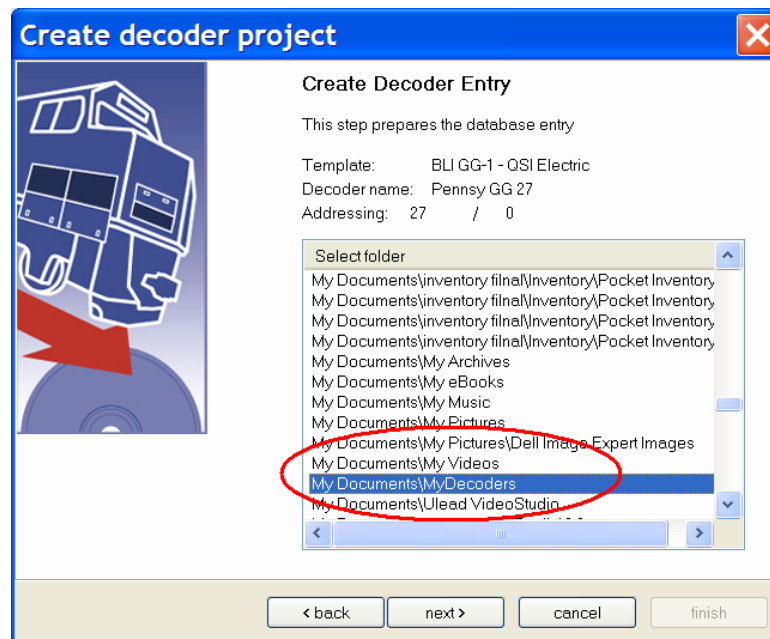


Figure 13: Entering project information for your decoder

- The Programming wizard allows you to specify the programming operation that you want to perform on the decoder.

There are three programming models that you can perform at this step, you can create a new template from the master templates (no programming) , you can clone a template from a previous locomotive, or not program anything and create your own template file for later modification. These options are described in the table below.

Clone decoder (optional) Select the decoder that you wish to copy. This will read one of your existing decoders, and either program the decoder, or create a new template file for use with the wizard

Create a template (optional) If this option is selected, a new template file will be created to use with Decoder Commanders template wizard. This is useful to create a decoder file that you can modify. If you modify a template file, you will need to run the template verification tool on your file.

Decoder Commander® Software

Read a decoder Read a decoder's CV's form the programming track and save the information to your decoder database for this locomotive

Write a decoder Write a decoders CV's to the programming track from the template file or a clone decoder (if selected).

It is best to clone a decoder on write operations, unless your club layout has a decoder standard, then you will use that decoder template.

TIP: When programming a new locomotive, place the locomotive on the programming track, before you decoder commander wizard. If you have purchased a sound decoder, check the manufactures instructions on programming. You may need to purchase a programming booster to program this decoder.

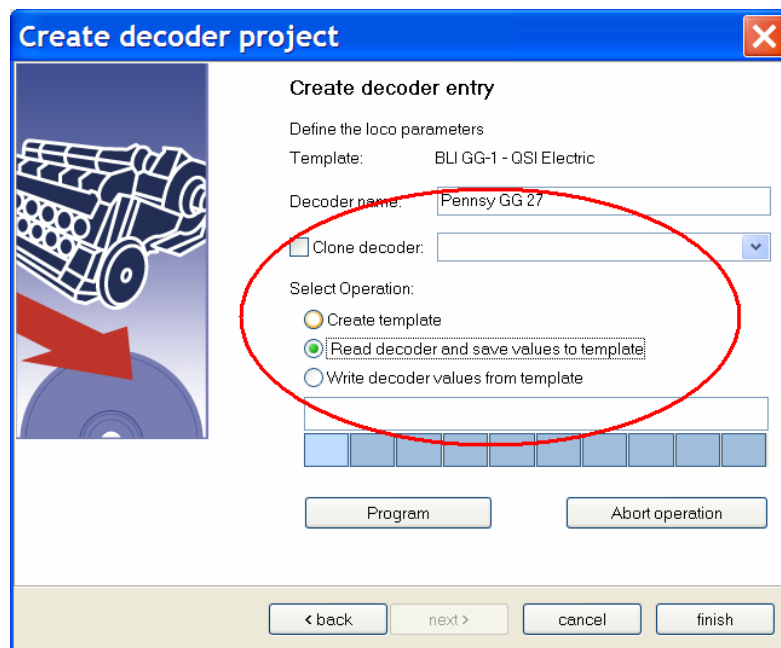


Figure 14: Entering project information for your decoder

Figure 14, shows that have decided to "read the decoder", and store the information into our decoder file, under the name of "Pennsy GG 27". To start the operation, we select program.

TIP: reading and writing a decoder takes time.. The read/write time may be upwards of 30 seconds per CV. Advance decoders typically have 50 CV's, so it is not uncommon to wait 10 minuets to read or write a mobile decoder.

- To begin the programming operation, select "Program". If you wish to abort the programming cycle, select "Abort". Stopping a command station programming sequence may take 30 seconds or more to stop the programming cycle.

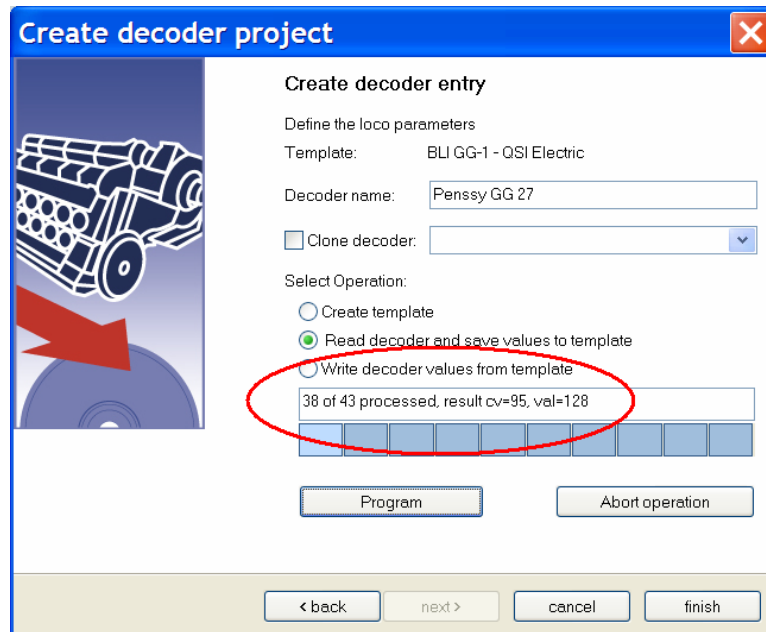
Decoder Commander® Software

Figure 15: Programming operation status update

- At the end of the programming cycle, you select finish. This will update the Decoder in your known decoder list.

Figure 15, shows the status update of the programming operation. This display will show you the number of CV's that are processed, and the value of each CV.

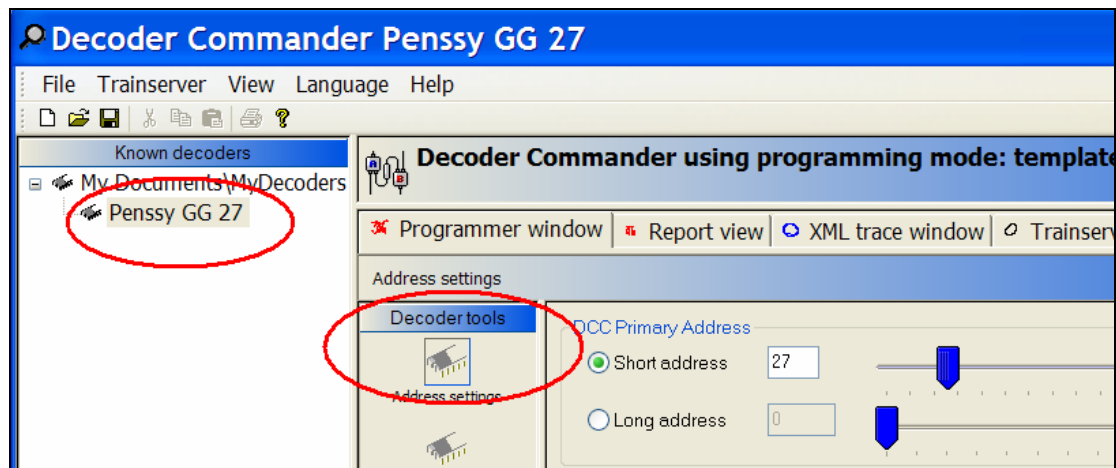


Figure 16: Status area updated with new Decoder information

After you have completed the programming operation, the Decoder Commander console is updated with the decoder information that we just read earlier.

Decoder Commander will scan your hard drive when Decoder Commander is started. The decoder information is stored in a XML data file in the directory that you specified.

Decoder Commander® Software

new decoder information. decoder tha tFigure 15, shows the status update of the programming operation. This display will show you the number of CV's that are processed, and the value of each CV.

Configuration Options and Tools

Decoder Tools

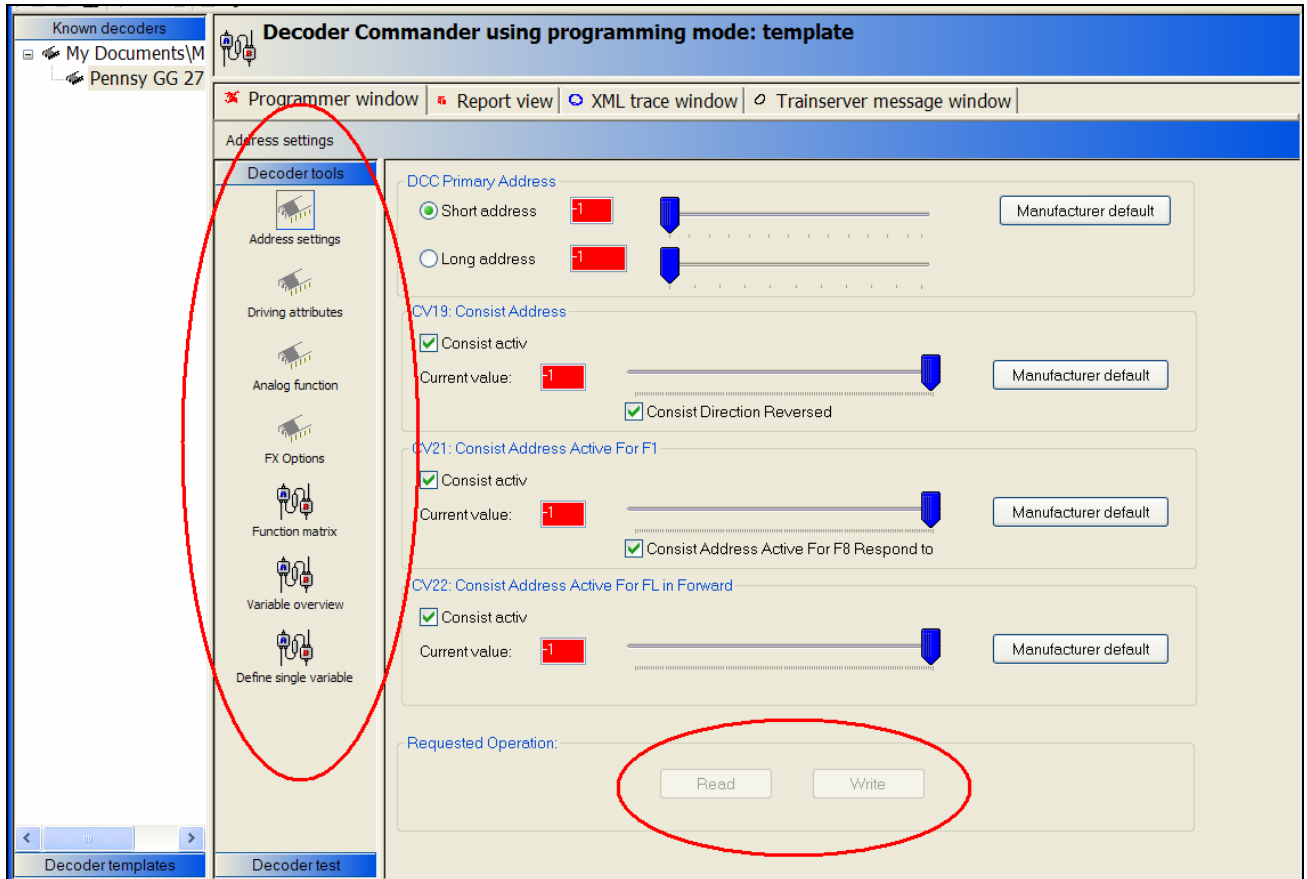


Figure 17: Programming with Decoder Commander Tools

You can use "Decoder Tools" to program your decoder. Decoder Tools allows you to program individual CV's, as groups of CV's. To use the Decoder Tools, select one of the known decoders (such as the "Penny GG 27"). Known decoders are decoder's whose data files are located on your disk. Known decoders are created using Decoder Commander programming wizard.

In figure 16, we have selected our decoder "Penny GG 27" to use. The Decoder Tools menu expands with the decoder tool options. In our example, we have started a programming operation (read was selected). The values that will be read, are changed to -1, and are colored red. Once the decoder values are returned from the command station these values are updated to returned values, and the background changes to white.

Decoder Commander® Software

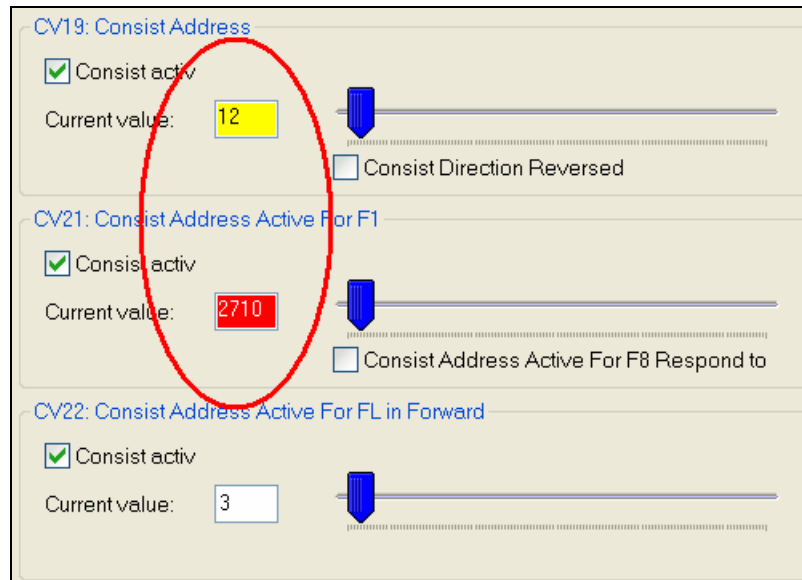


Figure 18: Information display on programming CV's

Decoder Commander allows you to program individual CV's in a group, and displays the status information about the programming operation. Decoder Commander uses KAM's Smart Decoder Technology to determine which CV needs to be written and updated to the decoder.

Yellow Shows a changed value for write operations. This value will be written to the decoder when the user depress the write operation.

Values that are in white are ignore on decoder write operation.

Red If the number displayed is in red and is not a -1, then this value is an error value. The value entered is out of range for the CV.

A -1 value in red, means that this information is being read from the decoder. The return value will be updated as clear text

TIP: If you wish to abort a programming operation, just select a different "decoder Tool". This action will abort the programming operation.

Decoder Commander® Software

Address Setting

The screenshot displays the 'Address Setting' configuration window in the Decoder Commander software. The left sidebar contains a menu with 'Address settings' circled in red. The main area is divided into several sections:

- DCC Primary Address:** Features a radio button for 'Short address' (selected) with a value of 27, and a radio button for 'Long address' with a value of 0. A slider is positioned between the two values. A 'Manufacturer default' button is on the right.
- CV19: Consist Address:** Includes a 'Consist activ' checkbox (unchecked), a 'Current value' of 0, a slider, and a 'Manufacturer default' button. A 'Consist Direction Reversed' checkbox is also present.
- CV21: Consist Address Active For F1:** Includes a 'Consist activ' checkbox (checked), a 'Current value' of 12, a slider, and a 'Manufacturer default' button. A 'Consist Address Active For F8 Respond to' checkbox is also present.
- CV22: Consist Address Active For FL in Forward:** Includes a 'Consist activ' checkbox (checked), a 'Current value' of 4, a slider, and a 'Manufacturer default' button.

At the bottom of the window, there is a 'Requested Operation:' section with 'Read' and 'Write' buttons.

Figure 19: Address setting CV's

Decoder addresses can only be program in service mode, and not in operations mode, with the exception of consist packet information (CV21 and CV22). All other CV's can be programmed on the main.

Decoder Commander® Software

Driving Attributes

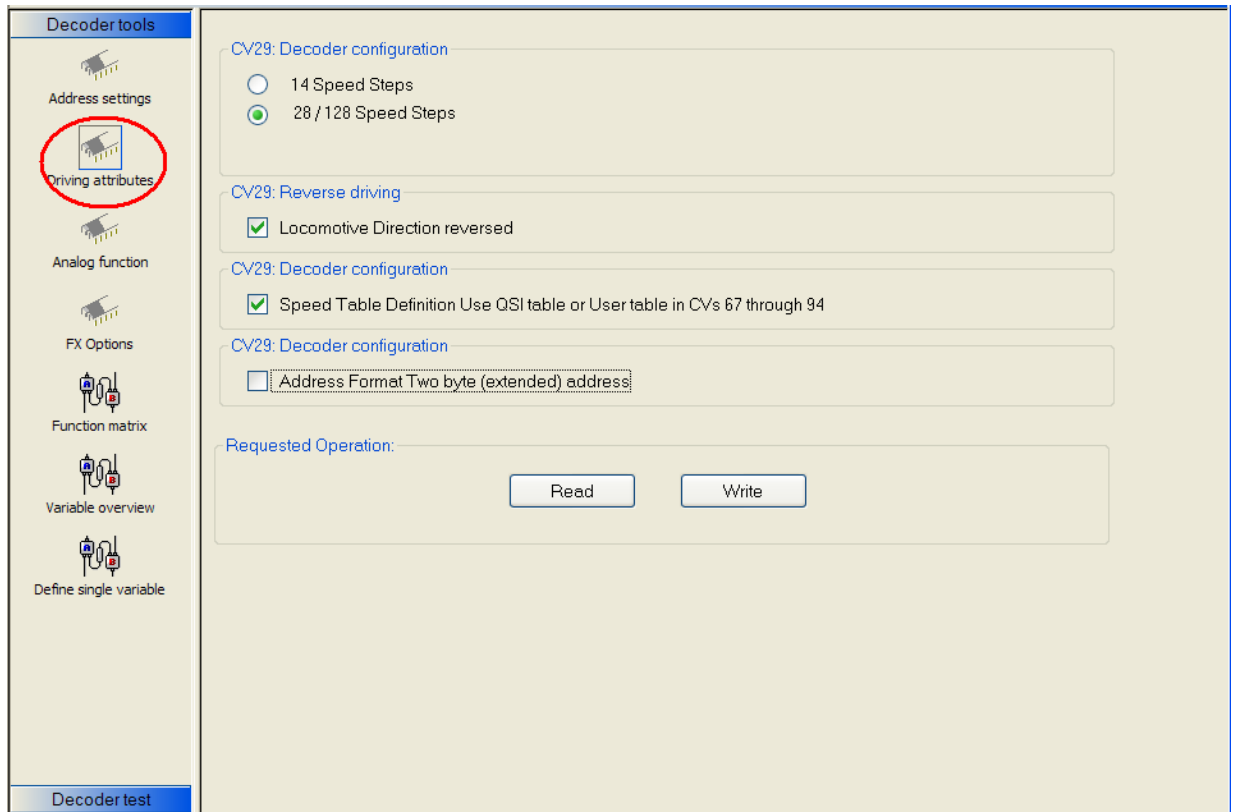


Figure 20: Driving Attributes

Driving attributes are those CV's that effect the operation characteristics of the locomotive decoder. Driving characteristics include speed step (14/28 mode), enabling a decoder speed table by use of the decoder, and to set the locomotives direction of operation (used for double headers)

Decoder Commander® Software

Analog Functions

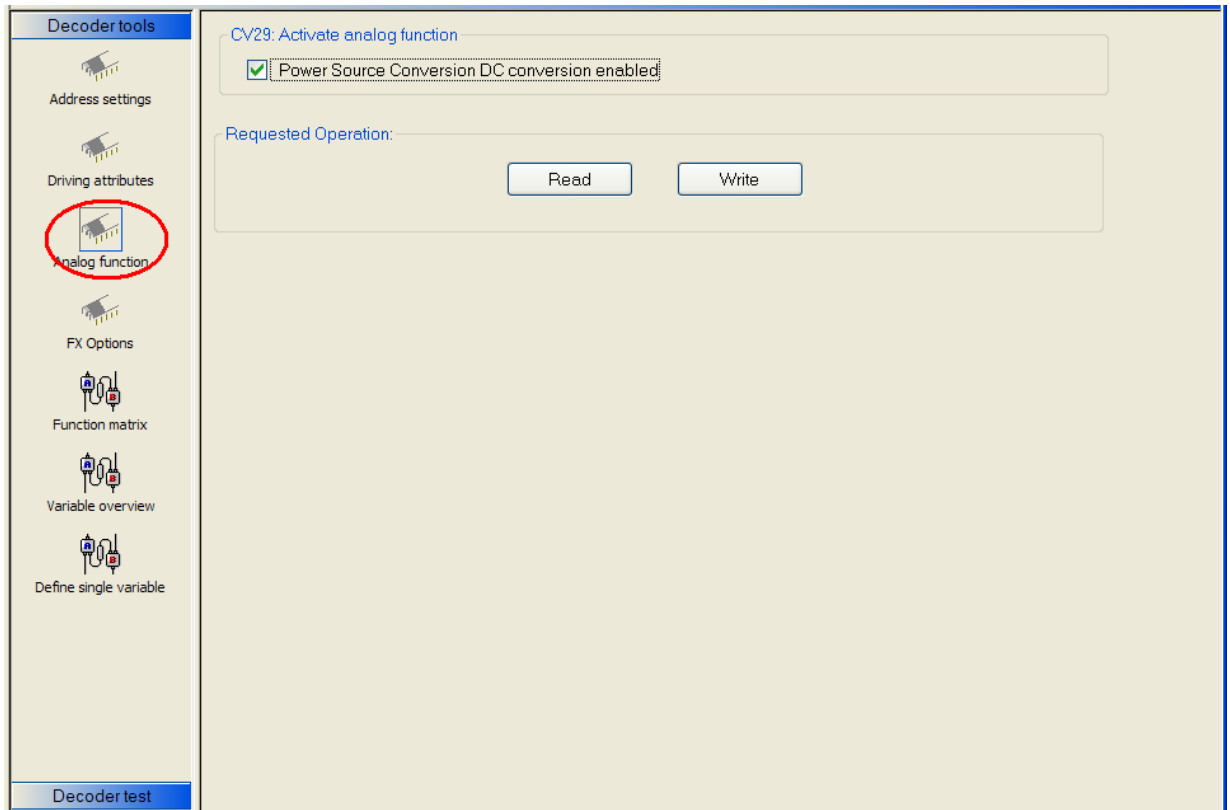


Figure 21: Analog Functions

Analog function support allows the locomotive to operate in analog mode, on a model railroad layout where a Digital Command and Control systems is not available. This capability is unique to each decoder, and is not universal across all decoders that are produced. If you wish to run your locomotive on a layout that does not support DCC, you will need to have CV29 analog mode enabled.

TIP: Analog mode does not always work with all Locomotives motors. You will need to check with the locomotive manufacturer to verify if the locomotive motor supports analog operation.

Likewise, all locomotive motors do not support DCC. Iron core motor typically will heat up and burn out on DCC layouts unless there is some electrical modifications made to the motor circuit.

Decoder Commander® Software

FX Options

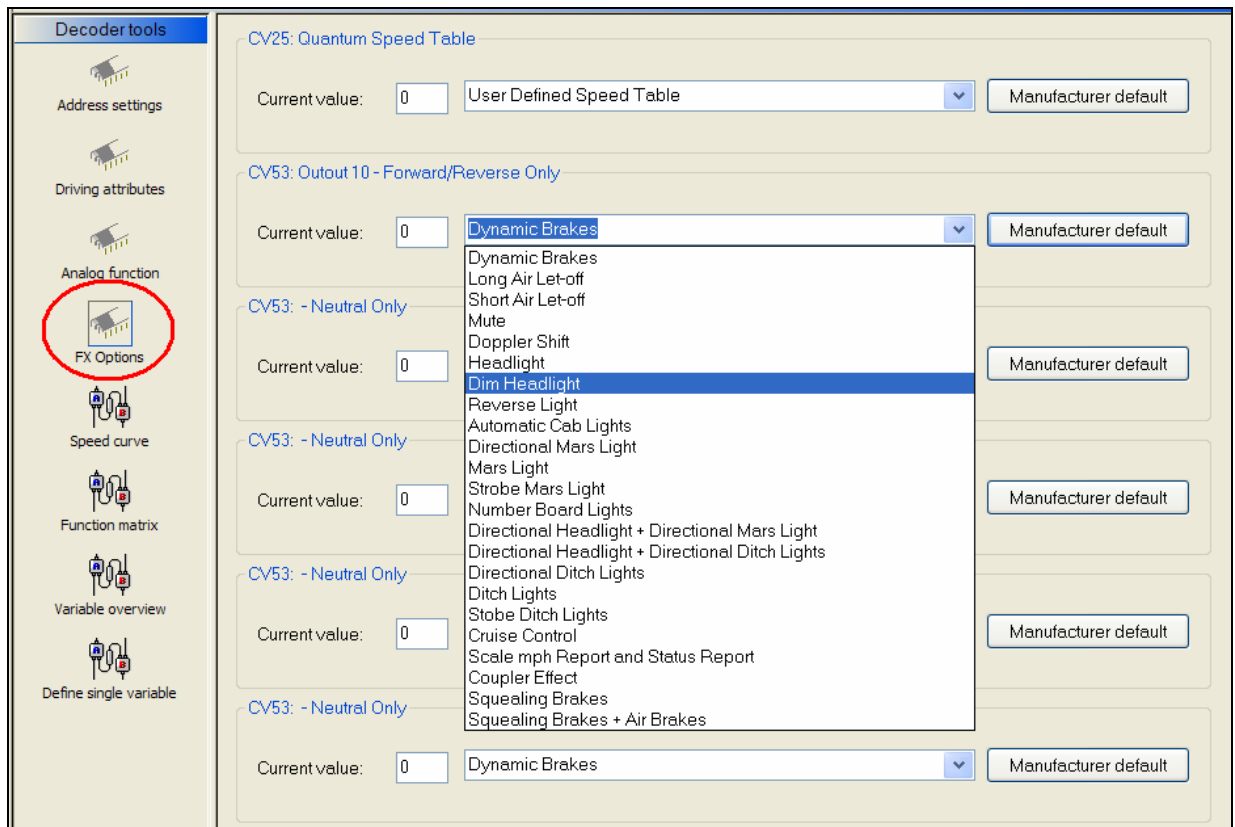


Figure 22: FX Function support

FX function support allows you to define the function bit modes of the decoder. These modes allow you different operating characteristics of the locomotive on the layout. Check with your decoder manufacturer to verify that bit mode is supported

Decoder Commander® Software

Speed Curve

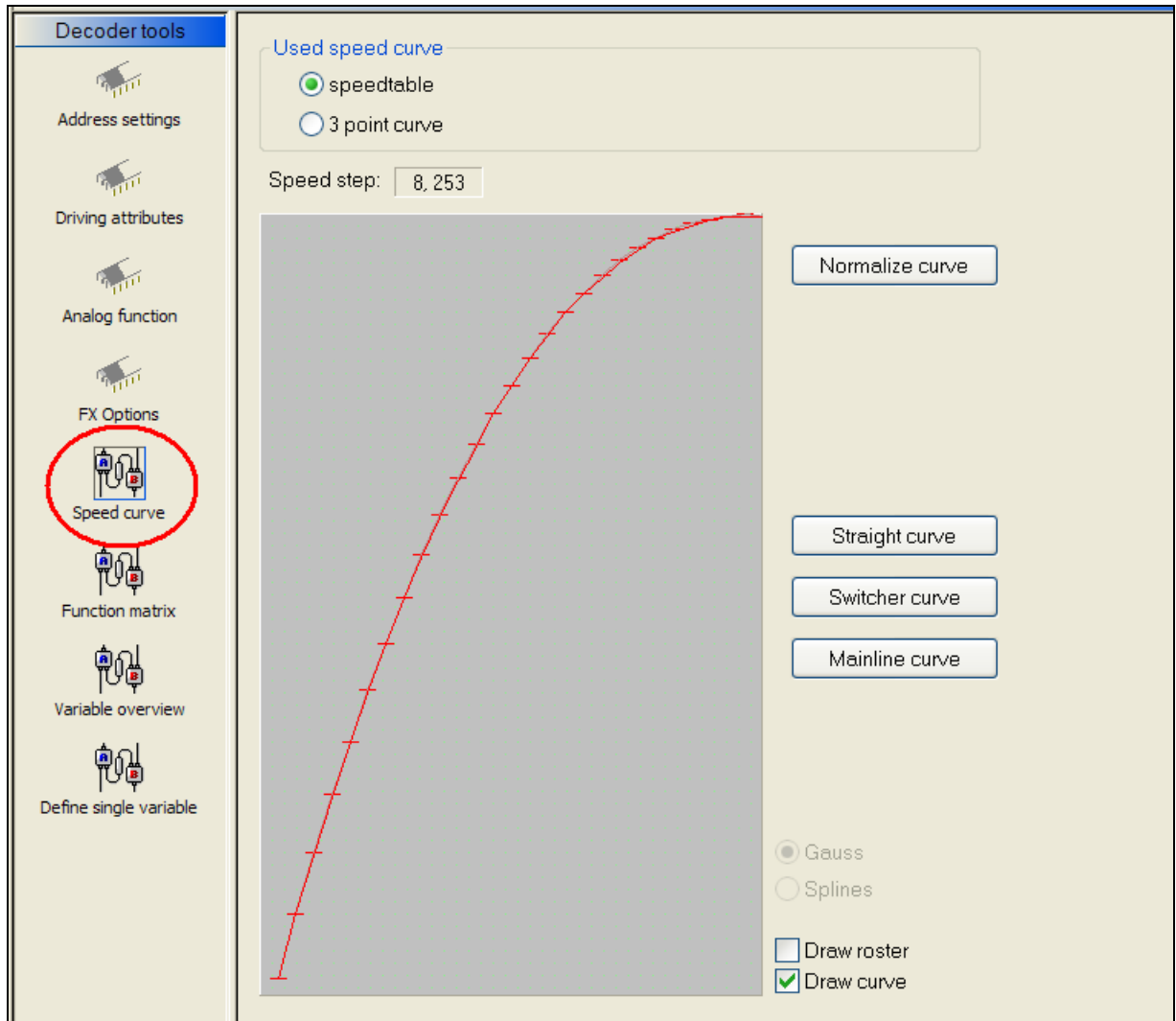


Figure 23: Speed Curve support

Decoder Commander supports speed, if the locomotive decoder supports speed curve. The decoder tools allows you to define any type of speed curve that makes sense for you locomotive. The tools support some preprogrammed speed curves, that would represent a switcher and mainline (freight/passenger).

On any speed curve, you need to be able to modify the speed curve. The decoder Tools, allow you to modify individual elements and normalized the curve.

If you choose a "read" operation, the decoder will be read, and the speed curve will updated based on the information from the locomotive.

TIP: Before you modify a speed curve, always read the speed curve from the locomotive. This will give you a base line to make your speed curve modifications.

Decoder Commander® Software

Function Matrix

	Fwd light	Rev light	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
Fwd light	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Rev light	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F12	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Manufacturer Configuration
Value:

Operation Requested:

Figure 24: Function Matrix mapping

Some decoders allow you to cross map the decoder functions, this is useful if you want to have a different operation sequence. For example, if you may desire to configure ditch lights of the locomotive when the head light is turned on. Another example, would be the operation of an uncoupler in a switcher. In running a switcher, you may desire to map the uncoupler to the rev light, so when a switch changes direction you can drop the cars behind it..

In both of these cases, you would map the decoder functions in the function matrix as shown above.

Decoder Commander® Software

Variable Overview

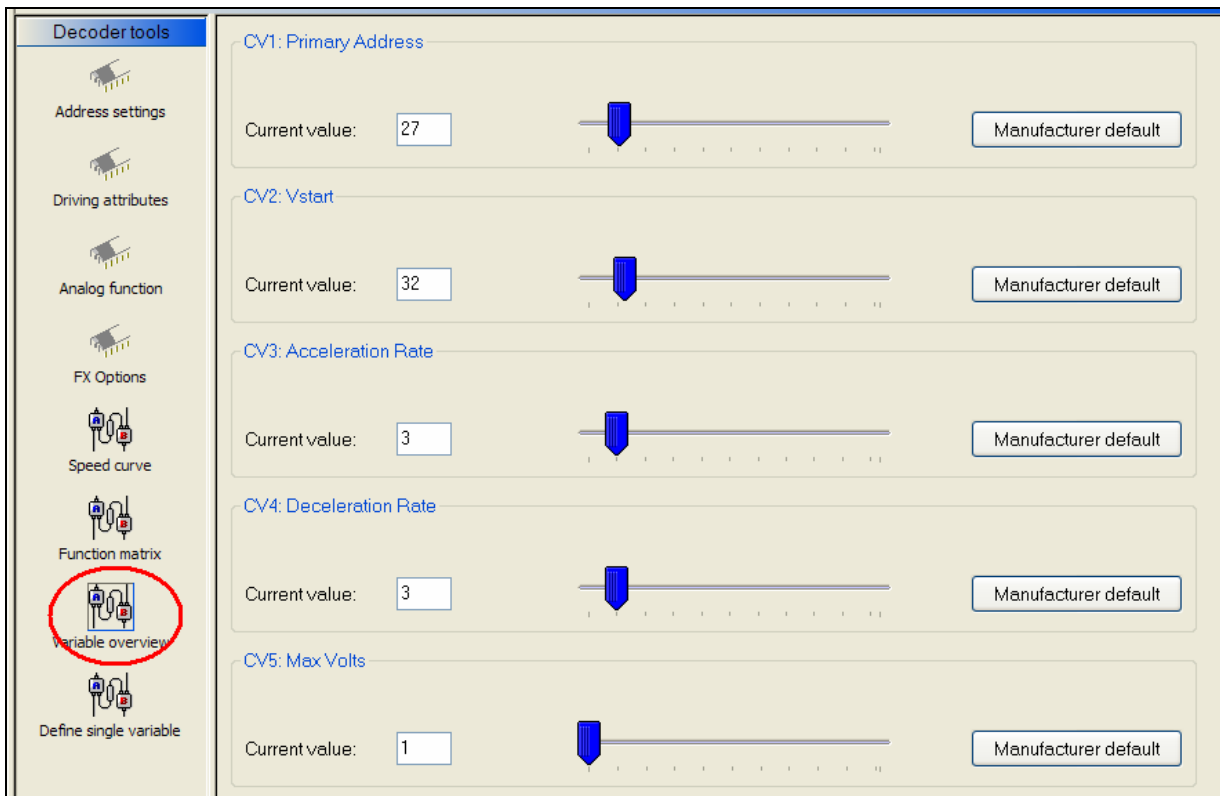


Figure 25: Decoder CV Overview

Variable Overview allows you to read and write a single CV variable or multiple CV variables to the decoder. The CV values and descriptions are from the decoder template. The CV's are programmed based on the template that is used, or Decoder Commander programming override.

To program a CV, find the correct CV that you wish to program, and move the slider till the appropriate value. Change one or all values. When it is time to program, Decoder Commander will program the CV's to the desired state.

Select the desired programming operation, either read or write. You can view the programming status in the status bar or the message window.

Decoder Commander® Software

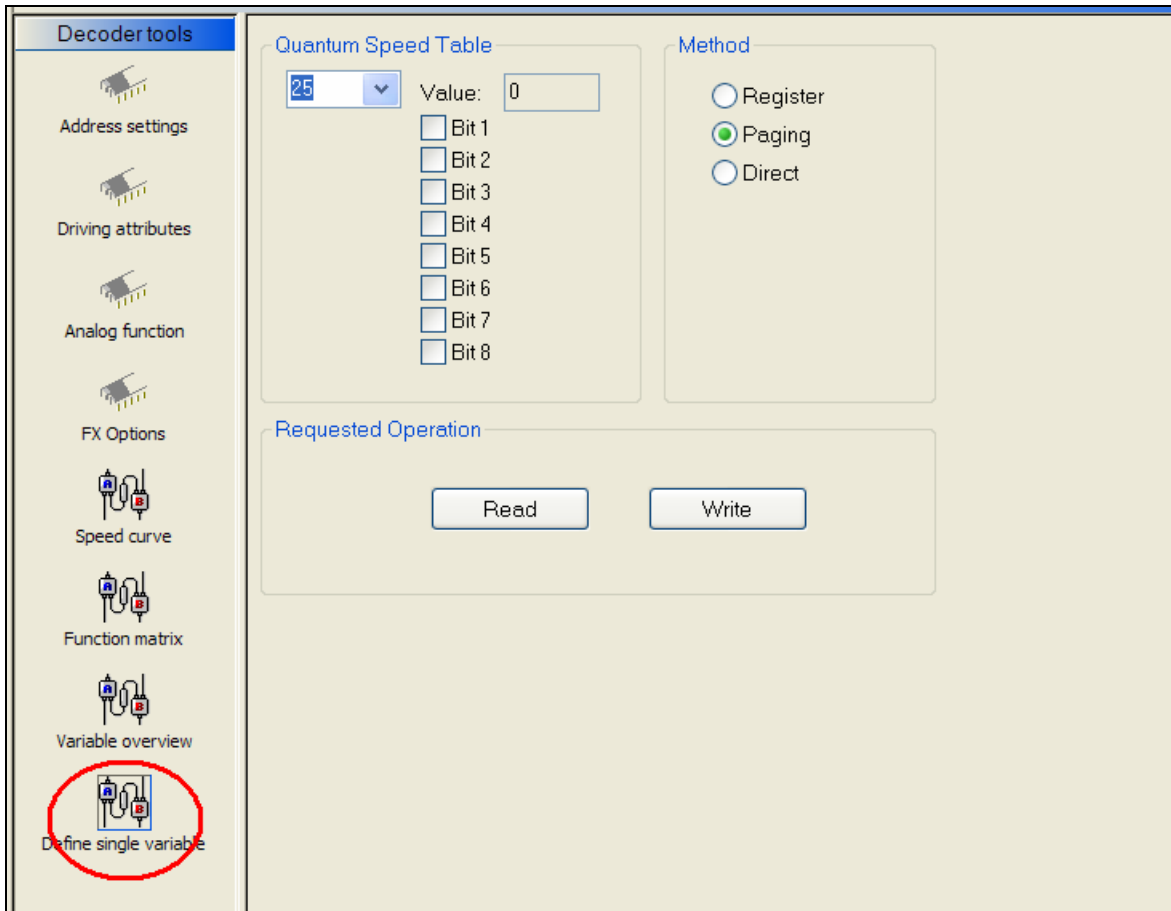
Define Single Variable

Figure 26: Single CV variable programming

Define single variable allows you to read and write a single CV variable to the decoder. The decoder template contains the necessary information about the decoder's CV, and displays that information. The programming operations displayed are derived from the decoder template.

Programming Information Windows**Status Menus**

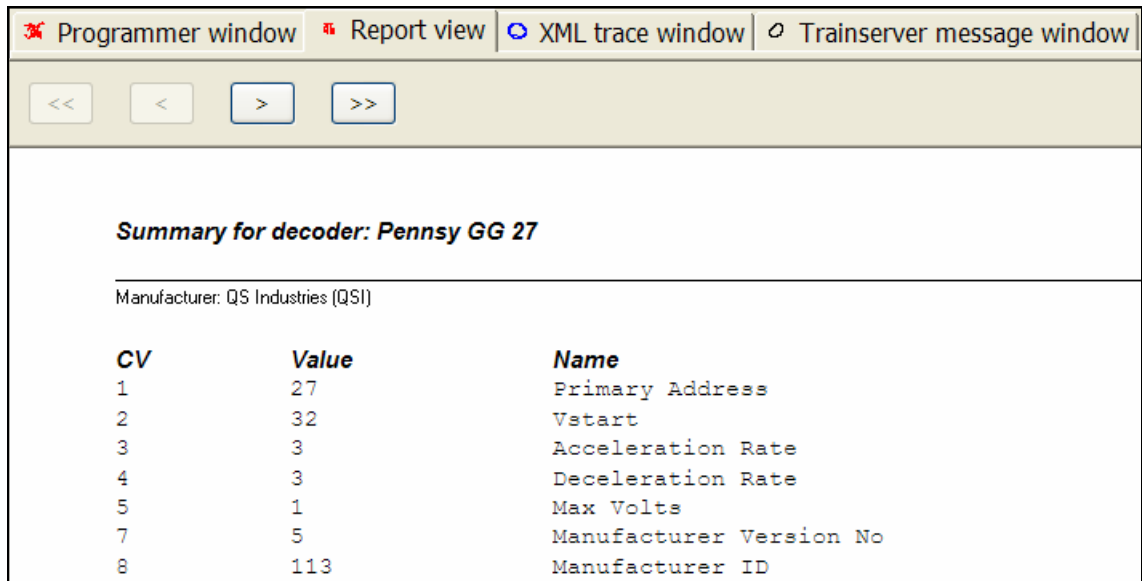
Decoder Commander maintains a runtime control of the decoder operations. There are four different status menus, Programmer window (default), report view, XML trace window and Train server message window.

Decoder Commander® Software**Programmer window**

The programmer window is the default view that allows the user to control the programming operations in decoder commander.

Report view window

The report view window is a print view of the decoder characteristics. This view is updated real time when the changes are reported by train server. If you desired to print a decoder summary, display the report view first, then select the print operation from the file menu.

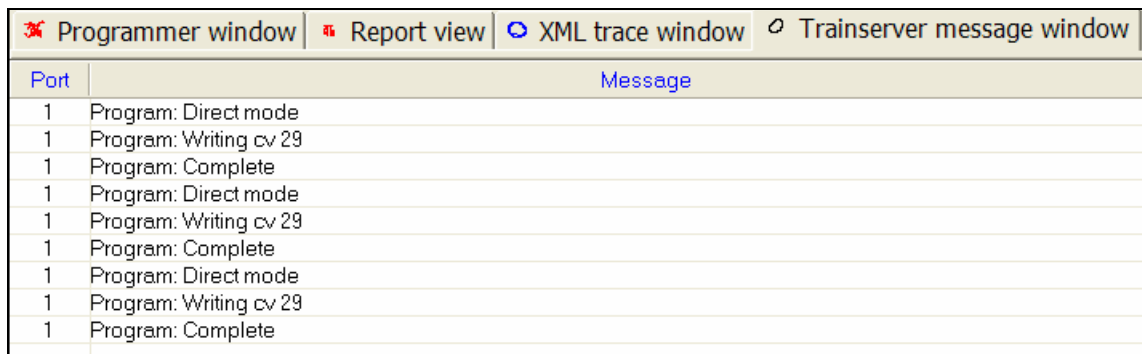


Summary for decoder: Pennsy GG 27

Manufacturer: QS Industries (QSI)

CV	Value	Name
1	27	Primary Address
2	32	Vstart
3	3	Acceleration Rate
4	3	Deceleration Rate
5	1	Max Volts
7	5	Manufacturer Version No
8	113	Manufacturer ID

Figure 27: Report View print window

Train Server Message Window


Port	Message
1	Program: Direct mode
1	Program: Writing cv 29
1	Program: Complete
1	Program: Direct mode
1	Program: Writing cv 29
1	Program: Complete
1	Program: Direct mode
1	Program: Writing cv 29
1	Program: Complete

Figure 28: Train Server status window

Train Server message window displays the message feedback information from the command station.

Decoder Commander® Software

Decoder Commander Template

Creating a Template

Decoder Commander uses a set of smart tags for template formats. These smart tags control new features displayed in Decoder Commander Console. You can create new template files of your own

TIP: If you wish to create a new template, always store the template in a directory other than the installed directory. This way your custom templates will not be overwritten by a software update.

Creating a new template is simple, follow these steps

1. Run the Decoder Programming wizard
2. Enter a Template name, and a directory to store it
3. Enter address "3" to as an address for the template
4. Select "Create Template" option in the last step of the wizard

Your new template will be created in the directory that you specified. You can now modify this template, and make changes. When you are completed with the changes in your template, you need to perform two steps.

1. Load the template file into Internet explorer and verify that your XML data format is correct
2. Run the Template verification tool to verify the template data and to make sure the template is compatible with Decoder Commander.
3. Start up Decoder Commander and begin to use your new template to program your decoders.

Testing your template is extremely important. The most minor change, can break your template. Always test any change that you make with the above three steps. The final step is to run the template through the Template verification tool. This tool will make sure that your template is valid for use with decoder Commander.

TIP: Detailed information about the decoder configuration variables may be found on the <http://www.nmra.org> web site, and the technical group website at <http://www.dcc.info>. Please refer to these web site for the latest information about the NMRA standard for Digital Command Control (DCC) advance attributes, Control Variables (CV's). Decoder Commander uses the NMRA standard when ever possible for decoder information and characteristics.

Template Verification Tool

Decoder Commander® Software

The Template Verification Tool is a tool that KAM has released to allow you to create your own template file, and use third party templates, and convert them into a format that is useable by Decoder Commander. The Template Verification Tool properly configures the XML syntax, and adds the missing syntax information that is required to use the various "Decoder Tools" in Decoder Commander..

Using the verification Tool

TBD

Template Format**Template Elements**

```

<?xml version="1.0" encoding="utf-8" ?>
- <DECODERS xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <DATE_CREATED />
  <DATE_MODIFIED>20050914</DATE_MODIFIED>
- <templateList>
  - <DECODER_TEMPLATE LANGUAGE="English">
    <TPL_VERSION>1</TPL_VERSION>
    <DECODER_VERSION>3</DECODER_VERSION>
    <DECODER_HIGH_VERSION>0</DECODER_HIGH_VERSION>
    <MANUFACTURER_ID>999</MANUFACTURER_ID>
    <MFG>Use My Decoder, Inc</MFG>
    <NAME>Special Demo Decoder</NAME>
    <PROG_MODES DIRECT="YES" PAGED="NO" REGISTER="YES" OPS="YES" />
    <RESET PRESENT="1" CV="8" VAL="8" />
    <SPEED_CURVE STARTING_CV="67" ENDING_CV="94" />
  + <fmapList>
  + <cvList>
  </DECODER_TEMPLATE>
</templateList>
</DECODERS>

```

Figure 29: Template List parameter on Decoder Commanders XML Template file

Template Capabilities (templateList)

The <templatelist> describes the primary features of the decoder template file. These features are described in the table below

DECODER_TEMPLATE LANGUAGE	This is the language of the template. All text descriptions will conform to the template language.
----------------------------------	--

Decoder Commander® Software

TPL_VERSION	When a modification is made to the template file, the template version number should change to the new version number.
DECODER_VERSION	Manufacturer version number that this template file corresponds to. When the manufacturer released the template file.
DECODER_HIGH_VERSION	Extended manufacturer version information about the decoder
MANUFACTURE_ID	The NMRA manufacturer ID
MFG	The NMRA manufacturer's name that corresponds to the Manufacturer ID
NAME	The descriptive marketing name of the decoder
PROG_MODES	Programming modes that the decoder can support. This is different the command stations programming support. The common subset is the programming mode that is used by Train Server to program the decoder.
RESET_PRESENT	New decoders have the capability to be reset to the Manufacturers default value. This is the reset value and CV that is used to reset the decoder to the Manufacture default state.
SPEED_CURVE	<p>Most decoders support a speed curve. This element describes the starting and stopping CV range of the speed curve. If this value is not present, then the speed curve will not be displayed.</p> <p>The decoder speed characteristics are displayed in the <i>Decoder Tools</i> "Speed curve"</p>
fmlist	<p>Some decoders support function mapping. This parameter enables the function mapping of the decoder. This capability is not present in all decoders.</p> <p>The decoder function matrix is displayed in the <i>Decoder Tools</i> "Function Matrix"</p>
cvlist	<p>This a listing of the CV values and description.</p> <p>The decoder CV characteristics are displayed in the Decoder Tools "Variable Overview"</p>

Decoder Commander® Software

Function Mapping (*fmlist*)

Function mapping list is the mapping (*fmlist*) of the decoder functions or the function matrix. These function attributes are located in the "Decoder Tools", function matrix. The function mapping contains 4 values that describes how the function is mapped as it conforms to the NMRA DCC specifications.

```
- <fmapList>
  <FMAP CV="33" DEFVAL="1" MASK="255" FNAME="Fwd light" />
  <FMAP CV="34" DEFVAL="2" MASK="255" FNAME="Rev light" />
  <FMAP CV="35" DEFVAL="4" MASK="255" FNAME="F1" />
  <FMAP CV="36" DEFVAL="8" MASK="255" FNAME="F2" />
  <FMAP CV="37" DEFVAL="16" MASK="2040" FNAME="F3" />
  <FMAP CV="38" DEFVAL="32" MASK="2040" FNAME="F4" />
  <FMAP CV="39" DEFVAL="64" MASK="2040" FNAME="F5" />
  <FMAP CV="40" DEFVAL="128" MASK="2040" FNAME="F6" />
  <FMAP CV="41" DEFVAL="256" MASK="2040" FNAME="F7" />
  <FMAP CV="42" DEFVAL="512" MASK="2040" FNAME="F8" />
  <FMAP CV="43" DEFVAL="1024" MASK="16320" FNAME="F9" />
  <FMAP CV="44" DEFVAL="2048" MASK="16320" FNAME="F10" />
</fmapList>
```

Figure 30: Function Mapping parameter on Decoder Commanders XML Template file

CV	The CV number for the function mapping
DEFVAL	The default value for the function mapping
MASK	The mask used for the function mapping. The mask is used to define how granular the function mapping is. On earlier decoder, the function mapping was only a few bits. 255 is an 8 bit function mask.
FNAME	The function mapped name

CV Definitions (*cvlist*)

The *cvlist* contains the CV definitions that Decoder Commander uses to program the decoder. This information is displayed in the "Decoder Tools" menu, as "Variable Overview" and "Define Single Variable"

The *cvlist* data format corresponds to the NMRA organization definition for control variables. However, the NMRA has only defined a limited number of CV variables. There are a large number of CV variables that are available for users to use. Decoder Commander is designed to be extended by the user and to display these variables in the Decoder Commander Console.

Decoder Commander® Software

```

- <CV INDEX="29">
  <DEFAULT>3</DEFAULT>
  <MIN>0</MIN>
  <MAX>255</MAX>
  <NAME>Normal direction of motion</NAME>
  <BITS BITS_DEFINED="YES" />
- <BIT_DEFINITION>
  <BIT_0>Normal direction of motion reverse</BIT_0>
  <BIT_1>Speed steps 28 speed step format</BIT_1>
  <BIT_2>Analog (DC) Operation DC conversion enabled</BIT_2>
  <BIT_3 />
  <BIT_4 />
  <BIT_5>Address Format Two byte (extended) address</BIT_5>
  <BIT_6 />
  <BIT_7 />
</BIT_DEFINITION>
<DATE_MODIFIED />
<BitMask>00000001</BitMask>
<stringList />
</CV>

```

Figure 31: CV list parameter on Decoder Commanders XML Template file

CV INDEX	The CV number that that CV corresponds to.
DEFAULT	Default value of the CV
MIN	The minimum value of the CV
MAX	The max value of the CV
NAME	The descriptive name of the CV. This descriptive name should be in the language of the template.
BITS BITS_DEFINED	This identifies if the bit position in the CV have a unique format.
BIT_0 ... BIT_7	If the BITS_DEFINED value is yes, then these are the corresponding bits for the CV register. Each value contains a description of the bit. This description is in the language of the template.
DATE_MODIFIED	An text string that used for modification information for the CV template.
BitMask	The bit mask used to define the bit parameters.
stringlist	A list of strings that corresponds to different values of the CV contents. For example, a value of 1, would correspond to the first element, a value of 2 would correspond to the second element.

Decoder Commander® Software

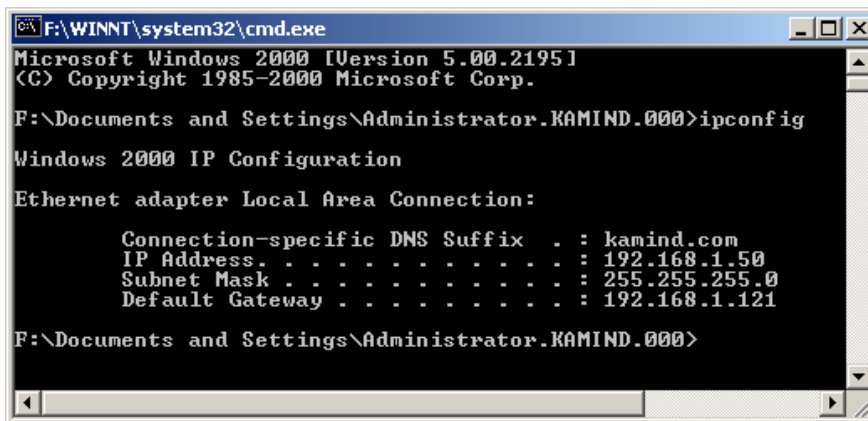
The stringlist is used when the BITS_DEFINED is set to "NO". and example of the parameter could be the following

`<BitName>Dynamic Brakes</BitName>`

Troubleshooting

How To Obtain The TCP/IP Address Of A Computer

- From the Windows Start Menu, select "Run..."
- Enter "cmd" (no quotes) and click the OK button.
- In the command window that opens, type "IPCONFIG" (no quotes) and hit the enter key on the keyboard.
- The IP address of this computer will be displayed. Use this value for the server address in Decoder Commander.



```

F:\WINNT\system32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

F:\Documents and Settings\Administrator.KAMIND.000>ipconfig

Windows 2000 IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : kamind.com
    IP Address . . . . . : 192.168.1.50
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.121

F:\Documents and Settings\Administrator.KAMIND.000>
  
```

Figure 32: Running IPCONFIG to determine the TCP/IP address

Cannot Connect To Train Server

- This is most commonly caused because Train Server has not been started using Train Server Manager.
- If Train Server is running on another computer, you must enter that server's name or TCP/IP address, not the name or address of the computer running Decoder Commander.
- The TCP/IP address of the PC running Train Server may be incorrect.

Most networks use dynamic TCP/IP address assignments (DHCP). This can cause TCP/IP addresses to change each time a computer is rebooted. The easiest way to prevent addresses from changing is to configure the router to "lease" TCP/IP addresses for a longer period of time. With TCP/IP address leasing, a computer gets to keep the previously used address as long as it logs in again within the expiration period of the lease (usually measured in days).

Another way to prevent addresses from changing is to manually assign static TCP/IP addresses and either not use DHCP or assign these static addresses outside

Decoder Commander® Software

of the range of addresses DHCP manages. Use IPCONFIG (described above) to determine the assigned TCP/IP address of a PC.

You can also use the name of the computer running Train Server. Even if DHCP assigns a new TCP/IP address to the computer, the computer name won't change unexpectedly when it reboots. A computer's name is easier to remember than its TCP/IP address but on some networks looking up a computer by name can be less reliable than using its TCP/IP address. Sometimes it may become necessary to help the name lookups by adding another computer's name and TCP/IP address into a computer's LMHOST file. The LMHOST file is located in the "\Windows\System32\drivers\etc\" folder. Of course, since the TCP/IP address of each computer must be typed into the LMHOST file, the lookup will fail if a computer's TCP/IP address changes.

- TCP/IP port 1069 communication may be blocked.

Train Server listens for connections on TCP/IP port number 1069. Firewall programs (such as Windows Firewall with XP) could be blocking that port from being able to communicate over the network. All computers running Engine Command and the computer running Train Server must have this TCP/IP port unblocked.

To unblock TCP/IP port 1069 in Windows Firewall, follow these steps: (For other firewalls refer to the manufacturer's documentation.)

- Open the Windows Control Panel. (Start menu -> Control Panel)
- Double click on Windows Firewall.

To verify if Windows Firewall is the problem you can temporarily disable Windows Firewall protection by clicking the "Off" radio button and then the OK button. Next try to connect Decoder Commander to Train Server. *Be sure to reactivate Windows Firewall afterwards to protect your computer!* If this solves the problem, turn Windows Firewall back on and continue these steps to unblock port 1069.

- Click the Advanced tab.
- Select your Local Areas Connection and then click the "Settings..." button.
- In the "Services" tab, if there is not already an entry for "tcp 1069" click the "Add..." button. Then enter either this computer's name or TCP/IP address, 1069 for the external and internal ports, and make sure TCP is selected. Then click the OK button.

Decoder Commander® Software

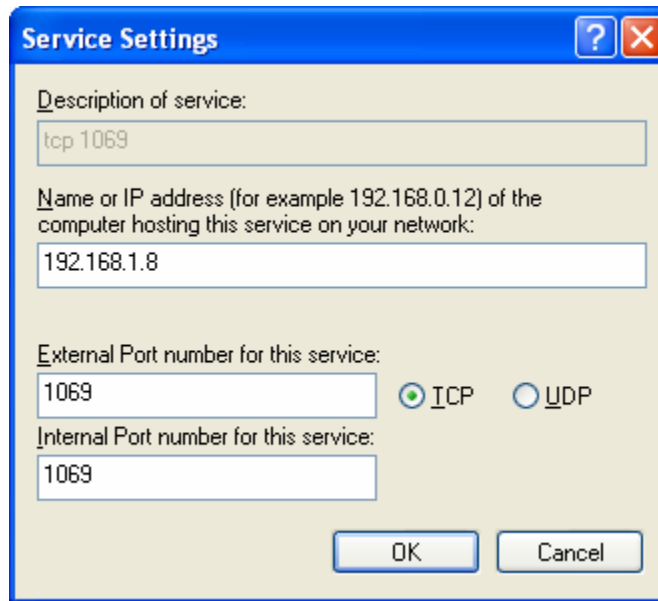


Figure 33: Unblocking TCP/IP Port 1069