

Installation Guide ***Revision-a***

MotorMan

DCC or Simple Serial Bus ***8 line Output Driver*** ***plus 8 line Input/Output Auxiliary Port***

This PDF is designed to be read on screen, two pages at a time. If you want to print a copy, your PDF viewer should have an option for printing two pages on one sheet of paper, but you may need to start with page 2 to get it to print facing pages correctly. (Print this cover page separately.)

Copyright

This document is Copyright © April 2014 by **RR-CirKits, Inc.**. You may distribute it under the terms of either the GNU General Public License, version 3 or later (<http://www.gnu.org/licenses/gpl.html>), or the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0/>), version 3.0 or later.

All trademarks within this guide belong to their legitimate owners.

Authors

Dick Bronson

Feedback

Please direct any comments or suggestions about this document to:

dick@rr-cirkits.com

Contact Information

RR-CirKits, Inc.
7918 Royal Ct.
Waxhaw, NC USA 28173

<http://www.rr-cirkits.com>
sales@rr-cirkits.com
support@rr-cirkits.com
1-704-843-3769
Fax: 1-704-243-4310

Publication date and software version

First published April 2014. Based on MotorMan Rev-a.

WARNING: This product contains a chemical known to the state of California to cause cancer, birth defects or other reproductive harm. Do not ingest.

*You can download an editable version of this document from
<http://www.rr-cirkits.com/manuals/MotorMan-manual-a.odt>*

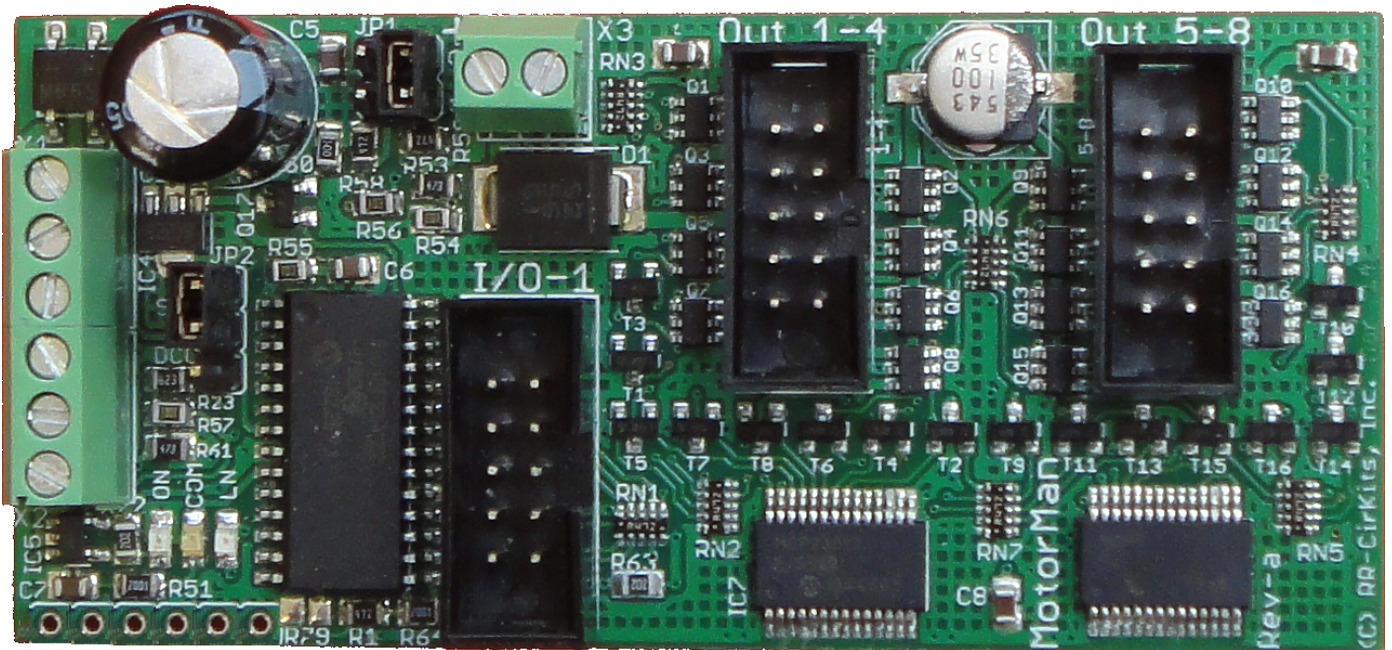
Contents

Copyright.....	2
Overview.....	4
1.0 Features.....	5
2.0 Getting Started.....	5
2.1 Create a Default Roster Entry as a template.....	5
2.2 Create a New Roster Entry for each MotorMan.....	6
2.3 Edit the new Roster Entry.....	6
3.0 Power and Serial Connections.....	7
3.1 LocoNet® Compatible Simple Serial Bus Connector.....	7
3.2 Power Connections.....	8
3.3 Status Indicators.....	9
3.4 JP1 JP2, and JP3 Mode Selection jumpers.....	9
4.0 Output Connections.....	10
4.1 I/O Ports.....	10
4.1.1 I/O Port Configuration.....	10
4.1.2 Input-Output mode.....	10
4.1.3 Context Sensitive Display.....	11
4.1.4 Composite Type.....	11
4.1.5 MotorMan I/O Connector Wiring.....	12
4.2 MotorMan Output Ports.....	12
4.2.1 MotorMan Output Power Supply Wiring.....	13
4.2.2 MotorMan Stall Motor Drive Wiring.....	13
4.2.3 MotorMan Dual Coil Wiring.....	13
4.2.4 MotorMan Single Coil Bi-Polar Wiring (Kato).....	13
4.2.5 MotorMan Stepper Motor Wiring.....	13
4.2.5.1 Five wire (4 Phase) Stepper Motor Wiring.....	14
4.2.5.2 Four wire (2 Phase) Bi-polar Stepper Motor Wiring.....	14
4.3 MotorMan Secondary Messages.....	14
5.0 MotorMan Logic Statements.....	14
5.1 Logic Blocks.....	14
5.2 Variables.....	15
5.3 Functions.....	15
5.4 ABS and APB Signal examples.....	15
6.0 MotorMan compatible Input/Output Cards.....	15
6.1 RB-4 (Relay Board - 4 x SPDT).....	16
7.0 Trouble shooting.....	16
7.1 Sanity Test.....	16
7.2 Activity Test.....	17
7.3 Default Address.....	17
8.0 Boot Loader.....	18
9.0 Grounding and Isolation.....	19
10.0 Warranty Information.....	19
11.0 FCC Information.....	19

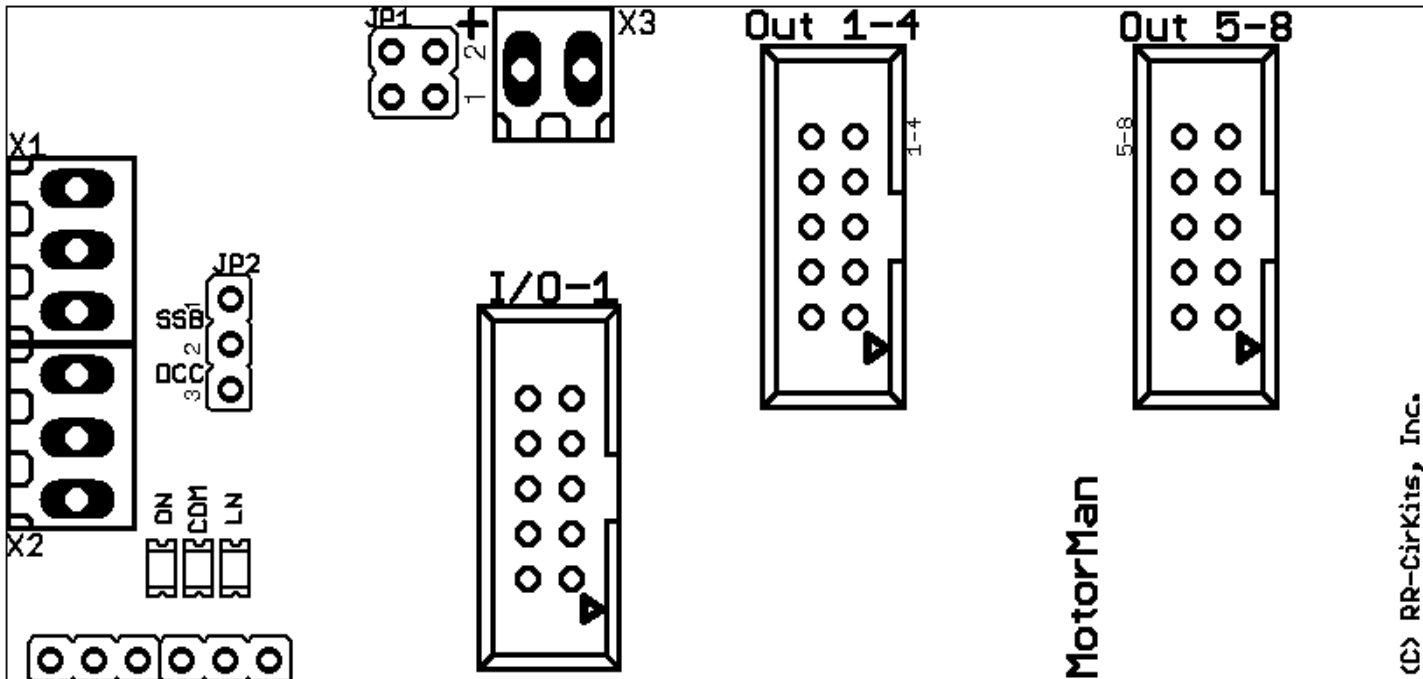
Overview

The MotorMan Simple Serial Bus interface provides a simple and easy way to connect between a LocoNet® bus and the layout. The MotorMan may be connected at any convenient point on the Simple Serial Bus or else directly to the DCC bus. (DCC mode is Output only)

LocoNet® is a registered trademark of Digitrax. www.digitrax.com



MotorMan



MotorMan Connectors

1.0 Features

- Support for 8 Output line driver pairs in any combination of:
 - 8 'H' Bridge drivers.
 - 8 Dual Coil drivers.
 - 8 Single Coil Bi-Polar drivers.
- 4 Stepper Motor drivers. **Note: Initial shipping firmware does not support stepper motors. An update to firmware revision 1.1 is required.**
- Support for 8 Input/Output lines in any combination of:
 - Up to 8 Input Lines.
 - Up to 8 Output Lines.
- Internal Logic Blocks with up to 32 conditional statements.
- Communicates over the Simple Serial Bus which is a 3 wire LocoNet[®] compatible serial interface.
- Alternatively will connect directly to a DCC bus. Input lines and internal logic not externally available in DCC mode.
- CV controlled programming via Software. (e.g. JMRI DecoderPro 2.4 or later.) Both reading and writing of CV values is supported in OPs mode over the LocoNet[®] compatible Simple Serial Bus connection. OPs mode 'Write Only' mode over DCC.
- Automatically saves input/output states during power down.
- Boot Loader allows firmware upgrades over the LocoNet[®] compatible Simple Serial Bus connection.
- Logic power is supplied by either the Simple Serial Bus or the DCC connection.
- External power input for output lines allows matching the load requirements.

2.0 Getting Started

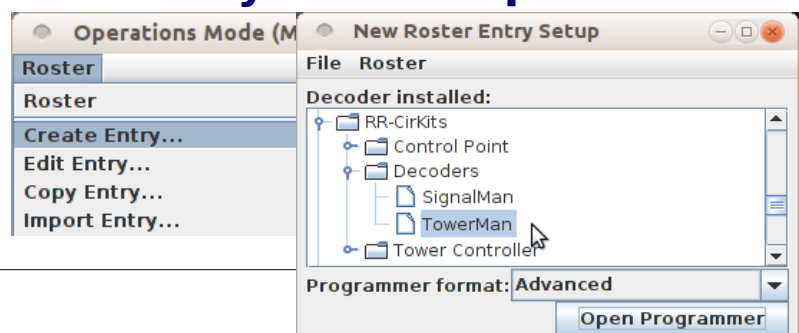
We suggest that you use a computer program such as the JMRI DecoderPro (3.4 or later) to setup the MotorMan. The "point and click" interface will save you much time and frustration while setting the many possible options that you will need to configure.

OPS Mode Address: Each MotorMan has a single address that is used for OPS mode programming on the layout. Each individual MotorMan on your layout may have its own address, (recommended) or each MotorMan may use the jumper selectable pre-programmed default address of 9999.

2.1 Create a Default Roster Entry as a template

Follow the following steps:

- a) Open the '**OPs Mode Programmer**' and select 'Roster' - 'Create Entry'.



- b) Next Select '**RR-CirKits**' - '**Decoders**' - '**MotorMan**' as the decoder type. Set the format: to 'Advanced' then Click 'Open Programmer'.
- c) Once the Decoder File opens, go to the 'Basic' tab, check '4 digit addressing' and enter an address of '9999' in the 'Long Address' box.
- d) Go to the 'MotorMan' tab.
- e) Set 'Master mode' as follows:

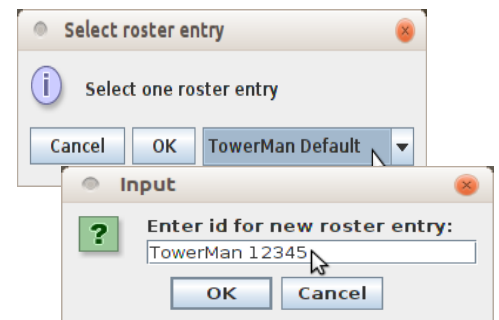
Master Mode No ▼ OPS Mode Programming Enable ▼

 - For use with a LocoNet® compatible command station select 'No'.
 - For use with any other, or no command station, select 'Enabled'.
- f) Set 'OPs Mode Programming' as follows:
 - To only use the default address of 9999 with JP1 1-3 on all MotorMan units select 'No'.
 - For user defined individual addressing select 'Enable'. (recommended)
- g) Go to the 'Roster Entry' tab, enter an ID (e.g. 'MotorMan Default') and then click on the 'Save to Roster' button.
- h) Close the new default decoder file. (template)

2.2 Create a New Roster Entry for each MotorMan

Follow the following steps:

- a) Open the 'OPs Mode Programmer' and select 'Roster' - 'Copy Entry...'.
- b) Select 'MotorMan Default' as the original.
- c) Enter a new roster id for each new MotorMan.
- d) Close each new entry by clicking on OK.



2.3 Edit the new Roster Entry

- a) This Ops mode address is only used to configure the MotorMan. It has nothing to do with the addresses that each line will use to respond to input or output commands.
- b) Connect the unit's LocoNet® compatible Simple Serial Bus to any point on your system's bus. The yellow command indicator will light each time the MotorMan initiates or responds to a LocoNet® command.
- c) Apply Power to the MotorMan. The green power ON status indicator should illuminate.
- d) Once the MotorMan is operating, all actual configuration, including any changes to the address, must be done with the JMRI 'Operations Mode Programmer'.
- e) When a MotorMan is powered up for the first time it will have a default address of 9999 to match the roster entry that you have just created from the 'MotorMan Default' template.
- f) If this MotorMan has an unknown address, Install the option select jumper JP1 at 1-3. The JP1 1-3 jumper will force the MotorMan into OPs mode and it will respond to both address 9999 and also what ever address was set by the CVs. This always allows you to take control of any MotorMan and correct



any setup errors. The JP1 1-3 jumper also forces the MotorMan into Master Mode Enabled for use on a stand alone bus.

- g) Any changed address will only take effect after the power has been cycled on the MotorMan.
- h) The MotorMan decoder file is large. You must use JMRI version 3.4 or later. (3.7.4 test or later) You also must not attempt to open two different programming windows at the same time. This may exceed the JMRI internal memory allocations.
- i) It is recommended that you give the MotorMan a long address in the range of 10000-16383 to avoid potential conflicts with locos if your command station allows it. This address must not conflict with that of any Locomotives or any other devices in use on the layout, as no distinction is made between mobile and fixed decoders in OPS mode.
- j) Change any other info you want to at this time, write all changes to the MotorMan, then save to the roster.
- k) Remove the JP1 1-3 jumper if used, then power cycle the MotorMan.
- l) You should now be able to interface to your Railroad from your software and make changes using its assigned address.

Tip

Remember, to modify, or restore an unknown address, you first power up the unit, then set the program option (JP1) to 1-3 and use the default 9999 address to re-program the unit. The unit will not power up with the jumper 1-3 in place. Always power it up first, then add the jumper.

3.0 Power and Serial Connections

The MotorMan (8 Line Driver Board) has six connectors and three status indicators. Two of these are terminals for connections to the system. One is for supplying the output power, and the other three are used as connections to the I/O lines. This section covers the system connections consisting of the serial port connectors, power connections and Status indicators.

3.1 LocoNet® Compatible Simple Serial Bus Connector

The serial port connection is made to the MotorMan via the Simple Serial Bus (SSB) cable connected to either of the two screw terminals. (3 pin) Simple Serial Bus cables are normally wired straight through, but no harm comes from an accidental reversal. The SSB wiring passes straight through both connectors.

Pin outs for the Simple Serial Bus data connector:

Pin	Description
1	DC or DCC
2	LocoNet® Serial line
3	DC or DCC

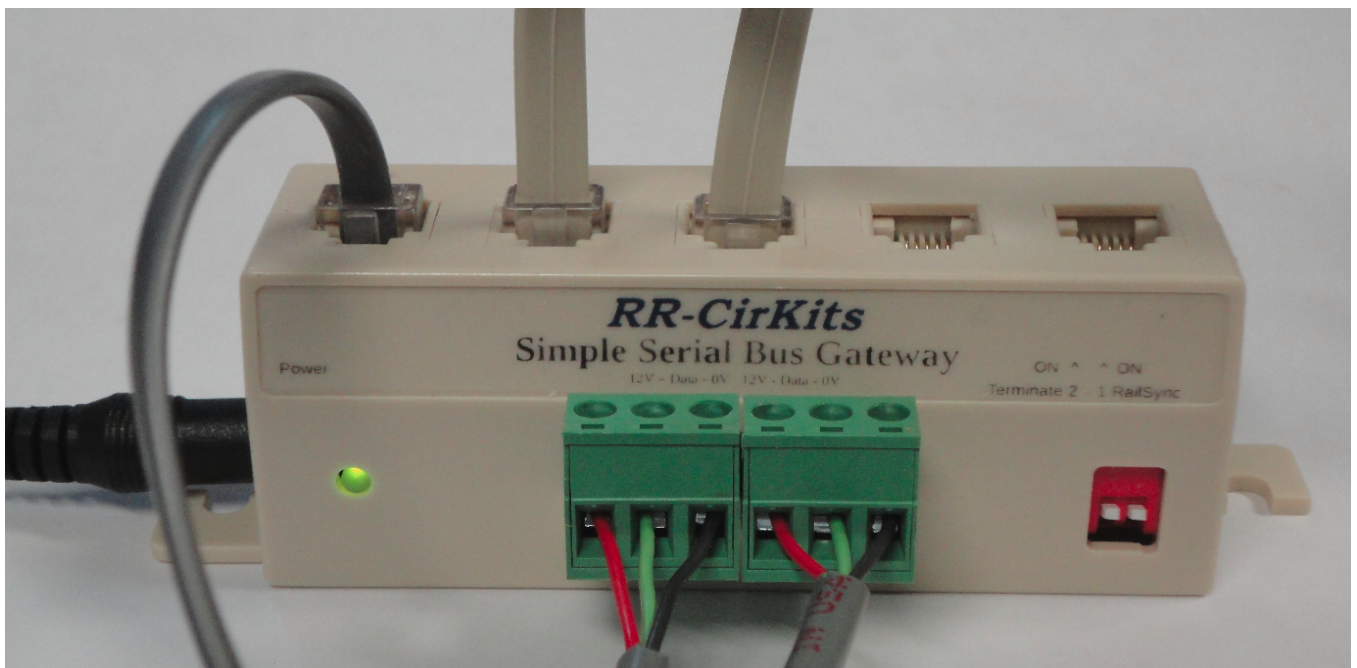
Simple Serial Bus power is supplied on Pin 1 and Pin 3. Power can be from +12V DC or DCC. Serial port Pin 2 is the LocoNet® compatible Simple Serial Bus signal line.

The Simple Serial Bus connectors will accept wire from 16 AWG to 22 AWG. For larger wire sizes simply attach a tail of smaller gauge wire to extend from the main bus wires to the MotorMan connectors. It may help to tin stranded conductors to prevent any stray strands from causing short circuits.

3.2 Power Connections

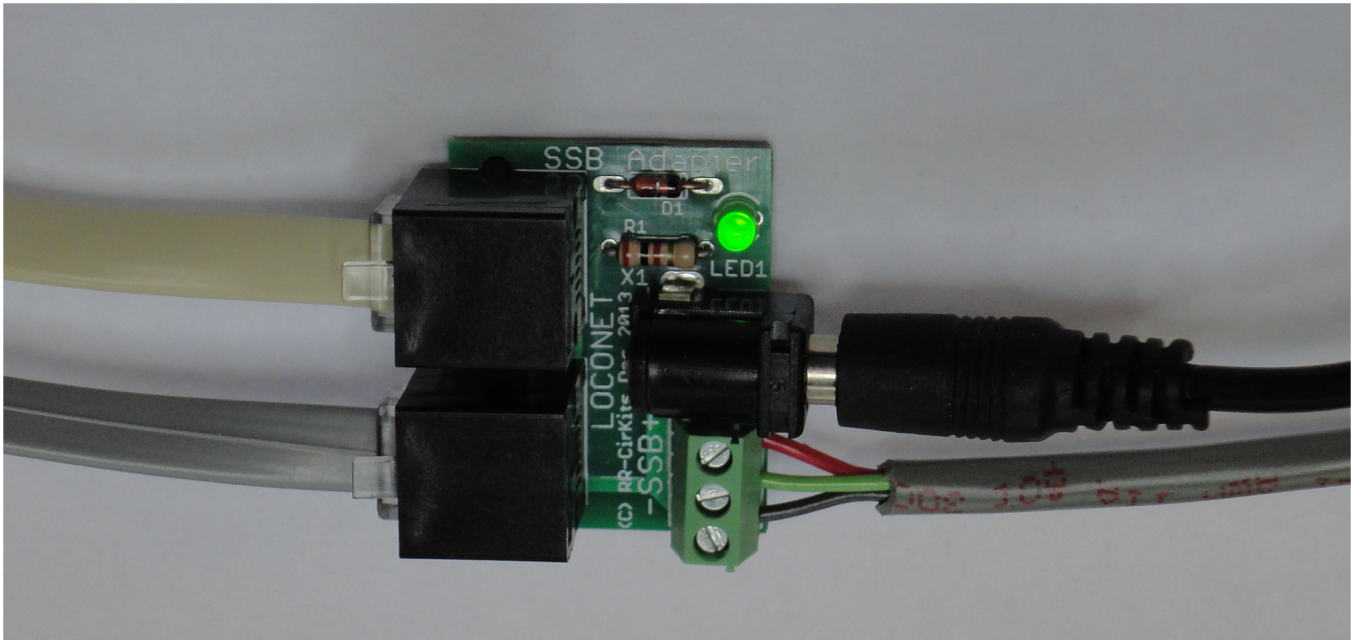
The MotorMan requires an external power source of between 11 and 15 volts DC or DCC supplied on the Simple Serial Bus. It also requires an external power supply of 5 - 25 volts DC for the loads. (see section 4.2.1)

The Simple Serial Bus Gateway unit is a convenient way to supply the required power to the MotorMan and other Simple Serial Bus boards over single or dual 3 wire cables. It also includes both a LocoNet® terminator and power to the RailSync lines as options for stand alone LocoNet® users. (i.e. not Digitrax) In addition it provides an easy point to connect up to 5 LocoNet® cables for compatible devices such as the LocoBuffer-USB computer interface



Simple Serial Bus Gateway

The SSB Adapter is the simple way for users with an existing LocoNet (e.g. Digitrax users) to add power to the SSB. (Simple Serial Bus)



SSB Adapter

3.3 Status Indicators

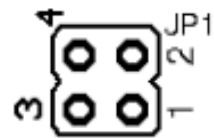
The MotorMan has three status indicators located near to the Simple Serial Bus connectors. The green ON status indicator shows the power status of the MotorMan itself. The yellow COM (command) indicator normally shows all data activity addressed to, or coming from, this specific MotorMan, and also any activity/error status during a boot loader firmware upgrade. (see section 8.0) The red LN indicator shows all LocoNet activity seen by this unit. It is useful for trouble shooting purposes.

3.4 JP1 JP2, and JP3 Mode Selection jumpers

The MotorMan has two Mode Jumpers, JP1 and JP2. These select the electrical interface mode and options.

JP1 has 4 valid positions:

- a) **None** = Serial mode. (LocoNet® data) Simple Serial Bus mode. (use with JP2 1-2) Read only at power up.
- b) **2 - 4** = DCC mode. (DCC data) DCC accessory decoder mode. (use with JP2 2-3) Read only at power up.
- c) **1 - 3** = OPs Mode Programming set to defaults. Read after power up.



Note This **1-3** jumper is not allowed during power up. It causes a rapid flash error warning if it is in place at power up.

- d) **3 - 4** = Boot Loader mode. Serial Mode only. Read only at power up.

Warning

For proper Boot Loader error status reporting be sure to remove Boot Loader jumper **3 - 4** after power up. The Boot Loader status is maintained until power is cycled or the boot process has completed.

JP2 has 2 positions.

- a) Serial = Serial mode. (1 - 2) Simple Serial Bus mode. (use with JP1 = none)
- b) DCC = DCC mode. (2 - 3) DCC accessory decoder mode. (use with JP1 = 2 - 4)



4.0 Output Connections

We suggest that the user take advantage of JMRI DecoderPro or a similar program to set the MotorMan configuration values, rather than attempting to use a hand held controller due to the large number of items to set, and the inability of a hand held to access 5 digit addresses. Current versions of DP also have the capability of setting groups of values with one selection.

4.1 I/O Ports

The MotorMan has one auxiliary Input/Output port for a total of 8 lines. The I/O port is normally configured as either all Inputs or all Outputs to be compatible with the various RR-CirKits I/O modules. However each line may be individually set as either input or output for special purposes.

Any special effects may be applied differently for each line. E.g. one line may be held steady while another sends a pulse or is blinking.

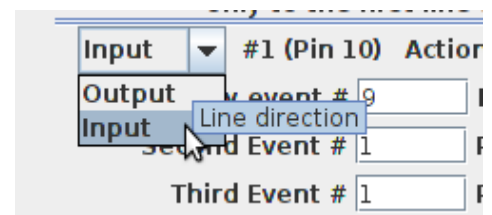
4.1.1 I/O Port Configuration

The following examples are using JMRI DecoderPro files for the MotorMan.

To prevent any confusion the MotorMan decoder file presents all input and output events in their common formats. For example LS9 is simply entered as 'Sensor Message 9' not 'unit #2, AuxA' or some such board oriented notation. This allows assigning any desired event to any line, no matter which board it is connected to.

4.1.2 Input-Output mode

The first selection seen on each line is its direction. Select the appropriate one. Normally a complete port will be assigned as either input or output lines, but this not required if your application allows. One example may be direct driving LEDs (with external resistors) and monitoring contacts in a control panel.



4.1.3 Context Sensitive Display

The MotorMan decoder file attempts to show a context sensitive display of the options presented.

Output	#1 (Pin 10)	Paired	Normal	Type	Short Pulse	Timing	Steady	Type	Driver
Primary event #	9	Polarity	respond inv	Switch Request	message	on Both Transitions			
Second Event #	1	Polarity	send normal	Switch Request	message	No Second Message			
Third Event #	1	Polarity	send normal	Switch Request	message	No Third Message			

Input	#2 (Pin 9)	Action	Normal	Debounce	Default	Type	Block Detector
Primary event #	10	Polarity	send inverted	Sensor Message	message	on Both Transitions	
Second Event #	1	Polarity	send normal	Switch Request	message	No Second Message	
Third Event #	1	Polarity	send normal	Switch Request	message	No Third Message	

For example in the above snapshot I have selected line #1 as a 'Driver' and line #2 as a 'Block Detector'. (Note the '**Type**' selection box at the end of each row) The items displayed for the output line include its type (Pulse, Blink) and its timing values. A 'Short Pulse' with 'Steady' length is interpreted as a continuous output, not a pulse. In the same location in the 'Input' line we find the contact 'De-bounce' timing value. For example if your block detectors do not include built in noise suppression for dirty track you might use a long value here to compensate for that. Note that the default value is sufficient to de-bounce noisy mechanical contacts. The MotorMan would not normally be used to drive signals, so the Signal mode is not made context sensitive.

4.1.4 Composite Type

JMRI decoder files include the option of a combo box that presets multiple options at one time. In this example we see some of these pre-configured types available. If you make changes to individual settings then the 'Type' box may change to 'Custom' indicating that JMRI does not recognize the new configuration. 'Custom' does not mean that your configuration will not work for your purposes. For example using a longer pulse timing for a stubborn dual coil switch machines will work, but it will show as 'Custom' to

serve as a warning against possible coil over heating.

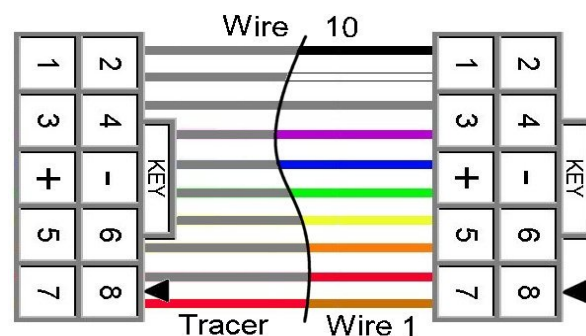
- **Block Detector** - Many block detectors have a low output when occupied. The 'Block Detector' setting compensates for this by sending an inverted response.
- **SPST On-Off Sensor** - Sensors may be of either polarity, so this setting allows you to set either 'send normal' or 'send inverted' as the message type. If your block sensors do not have an inverted output, then use this option.
- **SPST Alt Button** - Sometimes a single button can be used to send alternating commands. One example would be when controlling a local turnout from the fascia. The operator can easily see the position, and press the button if it needs to be changed. To do this set the 'SPST Alt Button' option, which customizes the MotorMan to send a 'Switch Request Message'. This option only sends as the button is closed.

- **Paired Button Inputs** - When you need two buttons that send paired events, then this setting does the job. It reserves two lines and uses the event ID that you assign to the first line. Each button only sends when it is closed. One sends normal and the other sends inverted outputs. It may be customized to send 'Switch Request' messages to directly control turnouts.
- **Driver** - This option is used with simple outputs such as stall motor drivers, relays, or direct drive lamps. It normally responds to 'Switch Request' messages. It may be customized to respond to sensors, for example to light occupancy lamps that follow block sensors. It also may respond to either the normal or inverted commands.
- **Dual Coil** - This specialized output option pairs lines and sends a short pulse as an output, closed on one, and thrown on the other. Of course the MotorMan does not have enough power to drive the coils of a turnout motor, so a board such as the SCSD-8 needs to be used to actually power the coils. As in other paired options, just the event ID entered in the first option box is used. Any event entered in the second line of the pair is ignored.
- **Dual Inverted** - This is simply an easy way to reverse the direction of the Dual Coil option without crawling under the layout.
- **Signal** - The signal option also pairs output lines. It also changes the interpretation of the two primary events to emulate the Digitrax SE8c behavior. For most users the better option would be to use a SignalMan. However for driving 4ASD-4 or SDB-4 boards this option does the job. Also it is possible to connect the SMD-8 board with TAB-3 options to drive Semaphore signals with Tortoise actuators.

4.1.5 MotorMan I/O Connector Wiring

The two port connector's wiring is as follows. Note that the pin numbers and I/O line numbers are NOT the same, and actually run opposite to each other.

Pin number	Connection name
1	line 8
2	line 7
3	line 6
4	line 5
5	Ground
6	+5VDC
7	line 4
8	line 3
9	line 2
10	line 1



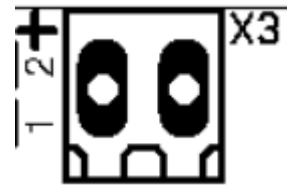
10 position IDC cable

4.2 MotorMan Output Ports

The MotorMan has dual output drive connections. The two output connectors **Out 1-4**, and **Out 5-8** are setup to drive 4 pairs on each connector as well as provide connections to the output power and ground common lines.

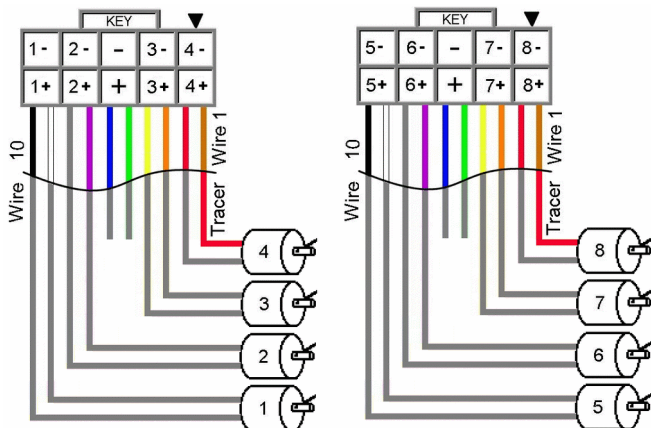
4.2.1 MotorMan Output Power Supply Wiring

The MotorMan is a high power output drive device and as such expects a local power supply for its loads. If your load is power limited such as Tortoises, then you can optionally use the SSB power wiring as a power source. The input power is polarity sensitive, so be sure to connect the power appropriately. There is a 3A protection diode provided to prevent damage.



Match the external power supply to the expected load requirements. For example if driving 5V stepper motors, then supply 5-6VDC. If driving Tortoise machines, then 9-12V works better. For dual coil machines such as Kato or Atlas then 12-15V may be required.

4.2.2 MotorMan Stall Motor Drive Wiring



Stall motors are connected to pairs of the output wiring. The ribbon cable may simply be split into pairs, or a board such as the FOB-A or FOB-C may be used. Another option is to use the TAB-2 or TAB-3 boards to plug directly to the Tortoise connectors.

The stall motor speed will be controlled by the power supplied to the MotorMan power terminals.

One option would be to supply 12V to the power terminals and then drive the stall

Stall Motor Connections to MotorMan

motors as Bi-Polar outputs. This would give fast action to the turnouts, but then remove power from the motors once they have changed positions to limit total power draw and motor heating.

4.2.3 MotorMan Dual Coil Wiring

Dual Coil machines such as Peco or Atlas may be driven by the MotorMan. Do not attempt to drive more than one coil per output line to avoid damage to the MotorMan output driver chips.

4.2.4 MotorMan Single Coil Bi-Polar Wiring (Kato)

Single Coil Bi-Polar machines such as Kato may be driven by the MotorMan. Do not attempt to drive more than one coil per output line to avoid damage to the MotorMan.

4.2.5 MotorMan Stepper Motor Wiring

Small stepper motors such as the 28BYJ-48 series of geared 4 phase motors are readily available on the Internet. Even smaller bi-polar steppers from digital cameras, floppy drives and similar sources are available as well. These can be used for complex automation as well as simple tasks such as turnout activation.

4.2.5.1 Five wire (4 Phase) Stepper Motor Wiring

Initial shipping firmware does not support stepper motors. An update to revision 1.1 is required.

4.2.5.2 Four wire (2 Phase) Bi-polar Stepper Motor Wiring

Initial shipping firmware does not support stepper motors. An update to revision 1.1 is required.

4.3 MotorMan Secondary Messages

In these examples there are no second or third messages being sent. However additional messages may be sent if desired, for example to activate a 'Next' turnout whenever the 'first' is sent in order to sequence a yard ladder. These additional messages, if enabled, are sent whenever the primary event occurs. Another simple example would be to allow two or even three different messages to be sent when a single button is pressed. A bit more complex use would be to send one message when a button or sensor is activated, and to send a different message when it is released. If a secondary message is configured as 'respond norm' or 'respond inv' then the selected event will control the output just as the primary event does. For example a single event could be master reset for many turnouts.

5.0 MotorMan Logic Statements

In addition to the optional secondary messages, the MotorMan contains 32 logic statements. Each statement may be used to create a new command based on one or two conditionals.

5.1 Logic Blocks

A logic 'block' consists of several statements all related to a single group. To calculate this logic, each logic block is treated as a group of statements independent of any other groups. The logic in any group will only be evaluated when some variable contained in that block changes state. This logic is event driven, not state driven. The logic blocks are used to do signaling logic or other animations such as controlling grade crossings. Each statement of this table consists of an identifier and two memory locations that remember the last command given for any items they are 'watching'. These memory locations remember the event information from the last seen commands to allow decision making. Any changes to these memory locations that cause a block to become 'true' will generate a new evaluation of the logic block, possibly sending out a new command.

Logic block statements for signals are required to be grouped together from the most restrictive aspect to the least restrictive aspect. The last entry of every block is marked by checking the 'End Logic Block' check box. Each entry that is to be evaluated must have its 'Enable' box checked. Each individual block evaluation is

Enable End Logic Block

done from top down in the list, and all evaluation for that block is terminated with the first matching statement or when the "End Logic Block" check box is reached. If no

enabled Mast-Aspect evaluates to 'true' then no Mast-Aspect message is sent, and evaluation ceases until some watched variable in that block changes again.

5.2 Variables

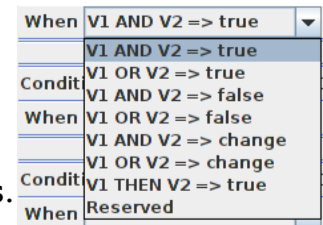
Each statement contains zero to two variables for identifying the required sensor, turnout, and/or following signal's aspect information in order to do simple signal

interlocking. Any logic statement evaluates to true only if its function and variables are true. A blank variable entry is considered to be true. Thus a blank entry at the end of a logic block will cause a default message to be sent if no previous statement with the same group number has been evaluated to 'true'.

The state of all variables is remembered internally in order to compare them. A change in either variable's state will trigger a new comparison. This dual variable capability makes it easy to calculate a signal aspect or train direction based on turnout position and occupancy.

5.3 Functions

The 'Function' column has options for **AND**, and **OR** statements. The 'When' selection entry expands this capability. Especially note the **V1 THEN V2** option. This allows you to only trigger on V2 if V1 was previously true. This makes it easy to determine train direction simply by using two occupancy sensors.



A statement may optionally send an event when true. To send an event define the event to be sent and check the 'Enable' check box. Events sent may be LocoNet sensor, switch, or switch feedback messages. Events may also be NMRA DCC Signal Mast commands. (a mast number plus its aspect)

5.4 ABS and APB Signal examples

For examples of using logic to control signals see the SignalMan manual. The MotorMan logic may be used to expand a SignalMan table if more statements are required.

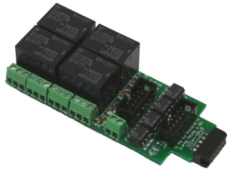
6.0 MotorMan compatible Input/Output Cards

The RR-CirKits MotorMan and its compatible I/O modules are designed to be clipped into Tyco 3-1/4" Snap-Track® mounted to the bench work. (Snap-Track® is a plastic channel designed to mount PC cards to a chassis, not something to run trains on.)

Each I/O module is equipped with one auxiliary connector to facilitate these connections. A single MotorMan or compatible I/O module fits into the 3TK2-1 (single) mounting track. Other widths are available for compact installations using multiple boards.

Any of the standard RR-CirKits I/O cards may be used with the MotorMan I/O connector. Of special interest is the RB-4 shown below. It may be used to switch frog power in situations where the turnout motor has none, or insufficient contacts to do the job.

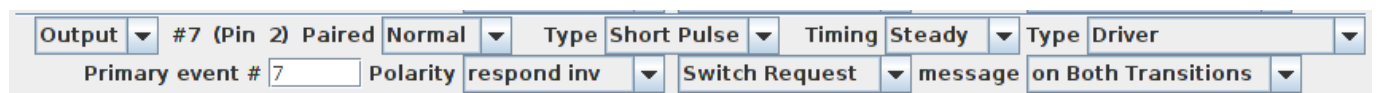
6.1 RB-4 (Relay Board - 4 x SPDT)



Relay Board - 4 is a Quad 10A SPDT relay board with logic level drivers. It is suitable for use with MotorMan, TC-64, LNCP, or other logic output devices. It requires 12V auxiliary power to drive the relay coils. Auxiliary power is optically isolated from the logic inputs for double isolation. LED indicators for each relay make it easy to monitor activity.

In some cases it may be suitable to use the 12V SSB power to supply the RB-2 relays, but be sure that you do not exceed the available current on the SSB line. Each RB-4 may draw up to 150ma. at 12VDC.

Includes dual ribbon connectors with offset lines to allow easy connection as output 1-4, or output 5-8, of the MotorMan, or other driver.



The RB-4 input lines are active low so all lines on this MotorMan port should be configured as "Driver". This inverted input mode matches most types of driver outputs, and the drive polarity may be easily switched either in the MotorMan setup or by reversing the RB-4 output contacts.

7.0 Trouble shooting

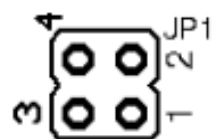
7.1 Sanity Test

To perform a very basic MotorMan sanity test perform the following steps:

- Power up the MotorMan by plugging it into the power supply.
- The green power LED should come on.
- The previous output states should be automatically be restored.

Or

- Apply the JP1 1 - 3 option jumper.
- Power up the MotorMan by plugging it into the power supply.
- This causes an error and the COM LED should flash rapidly if the unit is operational.



If the green power LED does not light, be sure that a power supply is connected to the Simple Serial Bus, or the DCC track power is on, and provides at least 11V to the MotorMan. The MotorMan includes internal power monitoring circuits, and will not restart operation until the supply voltage stabilizes at higher than 10VDC. The green power LED will initially light at much lower voltages, so it is not a reliable indicator of suitable power.

- If a volt meter is not available you can do a quick test by removing the power, applying the JP1 1-3 option jumper, then restoring the jumper. If the COM LED flashes rapidly, then the power up sequencing has completed OK. Be sure to restore the jumper to its normal position after the test.

7.2 Activity Test

The MotorMan's input circuit and code sends data directly to the unit's processor, so if you send any command to the unit it should immediately be seen on the command (COM) LED. This test uses the free software available from the JMRI project to watch the test commands. (www.jmri.org)

Steps:

- Open the JMRI LocoNet® Monitor window. Using the turnout control send a command to any output line on this MotorMan. The command should appear in the LocoNet® monitor window and the both MotorMan command (Y) LED and LocoNet activity (R) LED should blink.
- The connected output should respond.

If there is activity at the LocoBuffer-USB red LED, but no activity light at the MotorMan red LED when switch commands are sent, check the Simple Serial Bus wiring. If the command is seen in the LocoNet® monitor, but not in the command light, be sure that the command you are sending is addressed to respond on this MotorMan. If there is no activity shown in the LocoNet® monitor window, check that you have the correct interface selected in the JMRI preferences, and that you have the correct COM port selected.

7.3 Default Address

If the MotorMan's address settings get lost, reset the address as shown in section 2.0. The factory default address with JP1 1-3 in place is 9999. The MotorMan is also forced into Master Mode by the JP1 1-3 jumper, so it may be re-programmed on a stand alone bus.

Steps:

- 1) Power the unit.
- 2) Apply jumper at JP1 1-3
- 3) Use the MotorMan Default roster entry to restore the proper address for this unit in the "Basic" tab, then switch to the "CVs" tab and write just CV17 and CV18.
- 4) Remove the JP1 1-3 jumper

- 5) Remove then restore power to return the MotorMan back to its normal operating mode and enable the new (or restored) address.

8.0 Boot Loader

If an update to your MotorMan firmware is needed, a program such as "Download Firmware" in JMRI version 2.4 or later is required.

To enter Boot loader mode:

- 1) Remove power from the MotorMan.
- 2) Add a jumper at JP1 3-4.
- 3) Apply power to the MotorMan. The yellow COM light should light.
- 4) Remove the JP1 3-4 jumper.
- 5) The yellow COM LED will continue to light steadily indicating that the boot loader is now ready to accept messages from the JMRI Down-loader.
- 6) Start JMRI and select "LocoNet" then "Download Firmware".
- 7) Select the latest MotorMan upgrade and click "Read file" then "Download".
- 8) The COM LED will blink very faintly each time it receives a data packet from the JMRI down loader.
- 9) When programming is successful, the yellow COM LED will completely extinguish automatically, and normal operation will resume automatically.

Error codes:

During boot loading, several possible error codes are shown by the yellow LED:

- 1 blink: manufacturer code or product code mismatch; - Be sure that you have chosen the proper file to upload.
- 2 blinks: hardware version mismatch; - You may need to select "Accept later hardware versions".
- 3 blinks: hex file software version equal or lower; - You may need to select "Don't check software version".
- 4 blinks: unused (no LocoNet® data).
- 5 blinks: internal programming error detected. - The operation has failed, see below.

Each of the above errors will halt the boot loader process. To recover from these error types, remove power from the MotorMan, add the jumper at JP1 3-4, apply power again, then remove the jumper as before. Correct the error, and then re-run the JMRI program.

To recover from the last error type, it must be assumed that the firmware is in an undefined state, most probably damaged or incomplete. If the process is interrupted for any reason you will also end up with corrupted firmware which must be reloaded. The boot loader code itself is protected against being written over.

9.0 Grounding and Isolation

Unlike the LocoBuffer-USB and the Tower Controller, the MotorMan is not optically isolated from the Simple Serial Bus. This allows for possible ground loop problems between the LocoNet® and your layout power supplies, so be sure to keep the ground connection to the Terminator unit, either isolated, or else in common with your booster's power source.

Normally all MotorMan connections originate or reference to the MotorMan board itself. There is danger of ground loops with this high power output board, so it should use an isolated power source.

Properly ground your boosters, your power supplies, and your desktop computer through a 3 wire cable, and isolate them from each other via isolated equipment where necessary.

10.0 Warranty Information

We offer a one year warranty on the MotorMan. This device contains no user serviceable parts. If a defect occurs, please contact RR-CirKits at:

support@rr-cirkits.com for a replacement.

11.0 FCC Information

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and

2. this device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Any modifications to this device voids the user's authority to operate under and be in compliance with these regulations. The actual measured radiation from the MotorMan is much lower than the maximum that is permitted by the FCC Rules, so it is unlikely that this device will cause any RFI problems.

RR-CirKits, Inc.
7918 Royal Ct.
Waxhaw, NC USA 28173

<http://www.rr-cirkits.com>
sales@rr-cirkits.com
service@rr-cirkits.com
1-704-843-3769
Fax: 1-704-243-4310