

NMRA 2022
St. Louis
Applications and the future
(Layout Command & Control)

Compiled by: Dick Bronson
RR-CirKits, Inc.

LCC.

Part 1 (What is it, who is it for?)

www.rr-cirkits.com/clinics/NMRA-2022-LCC-A.pdf

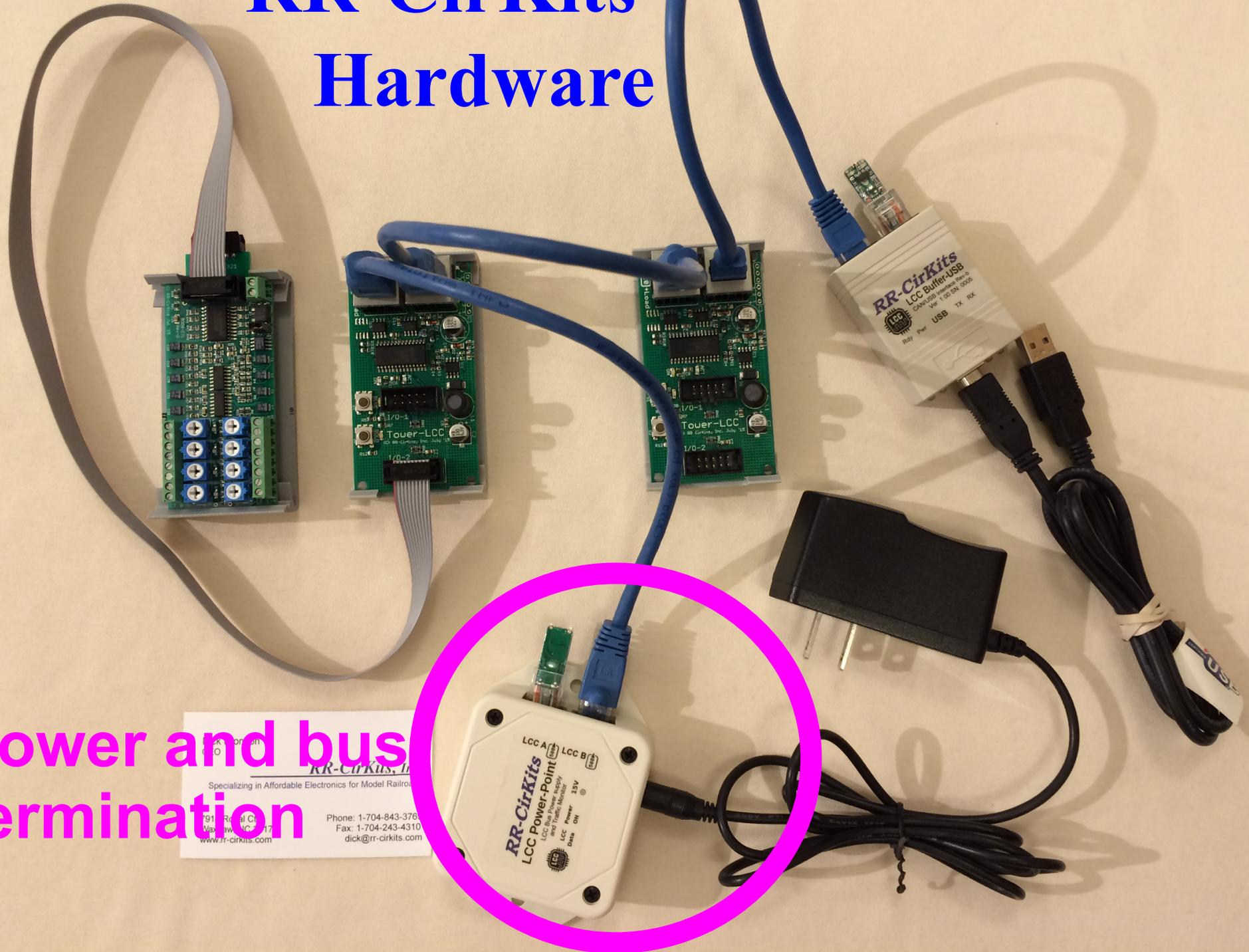
Part 2 (Applications and the future)

www.rr-cirkits.com/clinics/NMRA-2022-LCC-B.pdf

Applications

- The first task is to create an LCC network.
 - Power
 - Termination
 - One or more LCC nodes

RR-CirKits Hardware

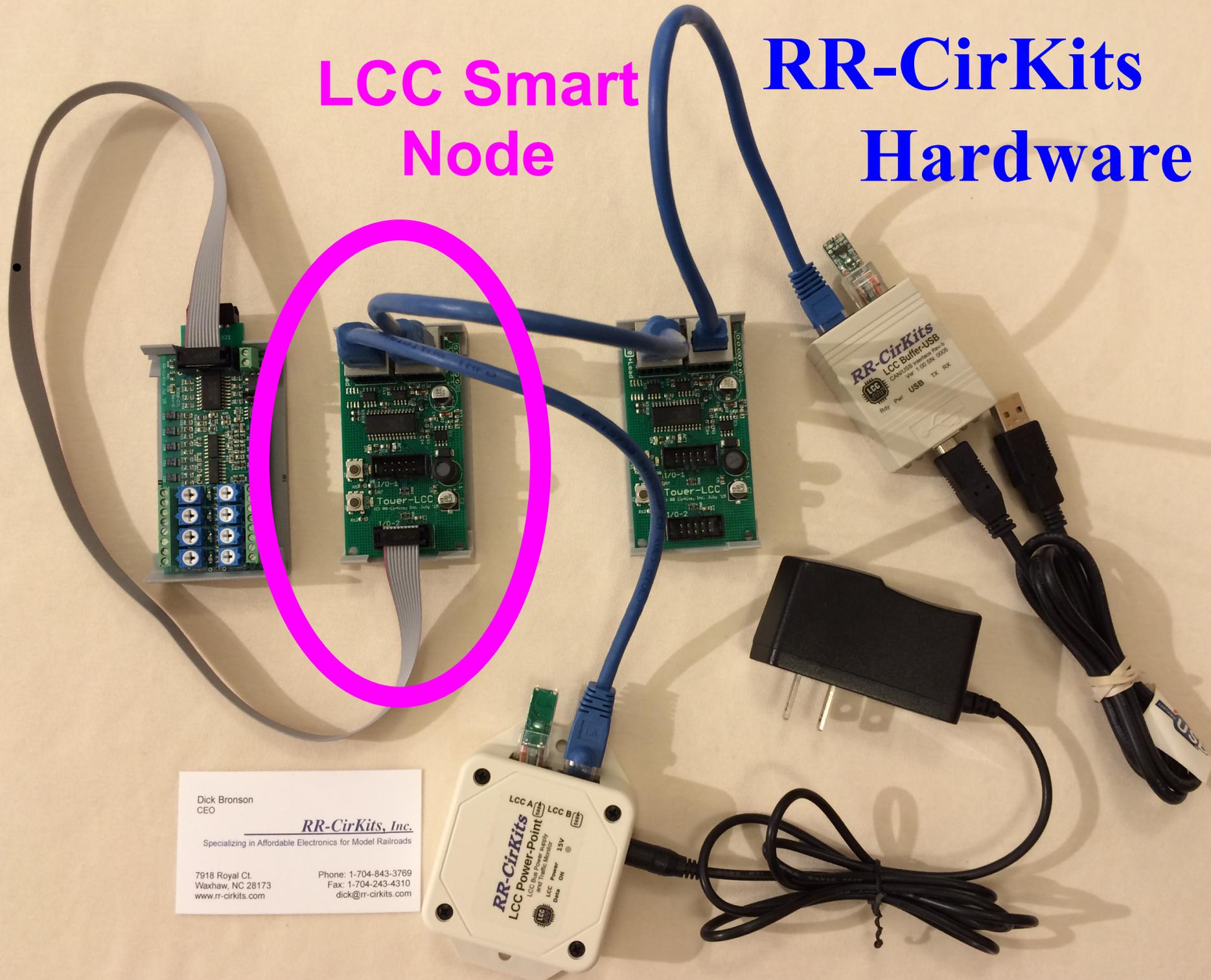


power and bus
termination

RR-CirKits, Inc.
Specializing in Affordable Electronics for Model Railroaders
911 Railroad Ct.
Mason, OH 45040
www.rr-cirkits.com
Phone: 1-704-843-3763
Fax: 1-704-243-4310
dick@rr-cirkits.com

LCC Smart Node

RR-CirKits Hardware



Dick Bronson
CEO

RR-CirKits, Inc.
Specializing in Affordable Electronics for Model Railroads

7918 Royal Ct.
Waxhaw, NC 28173
www.rr-cirkits.com

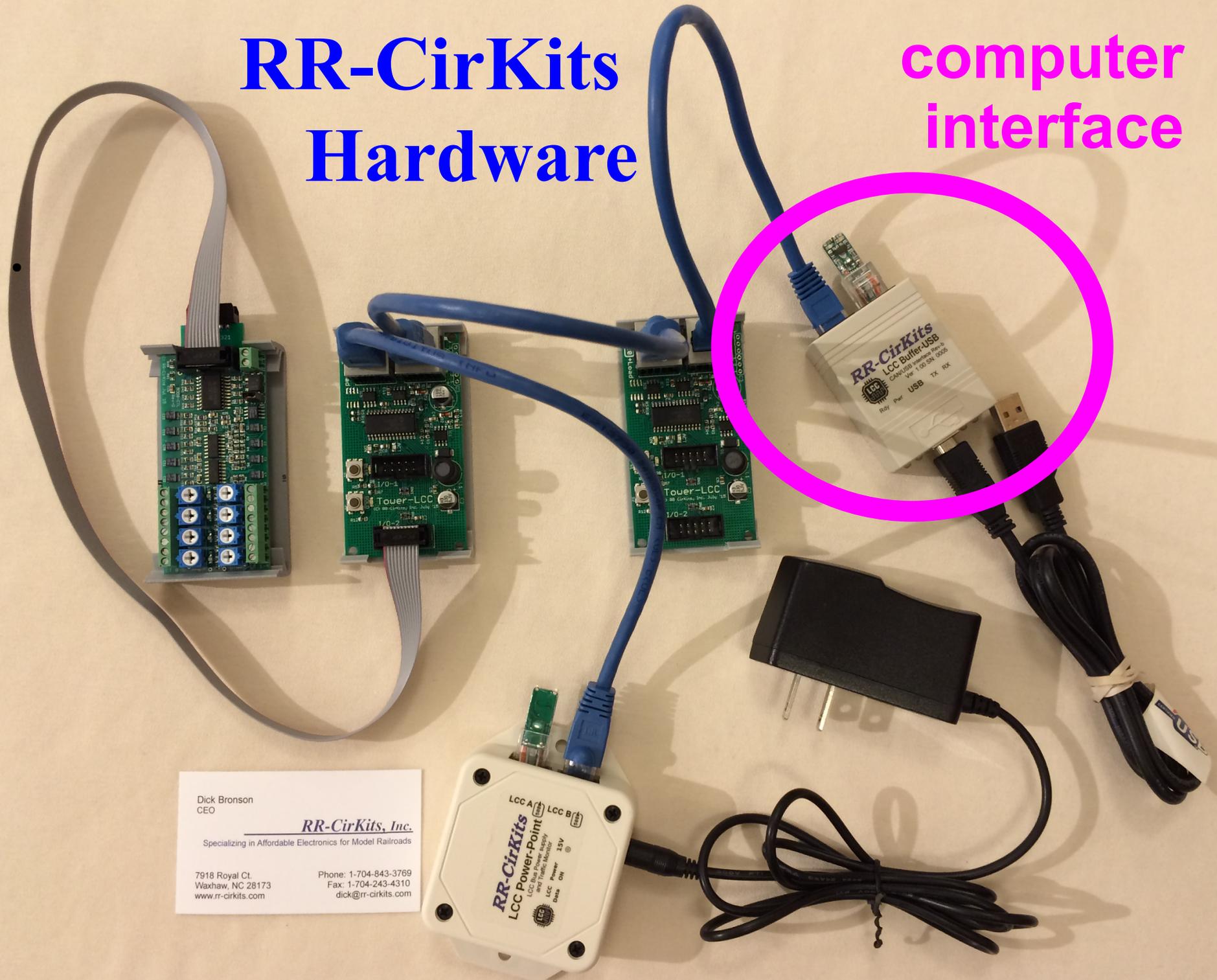
Phone: 1-704-843-3769
Fax: 1-704-243-4310
dick@rr-cirkits.com

Applications

- The next task is to connect LCC to JMRI or MRS
 - A LCC Buffer-USB
 - B LCC LocoNet Gateway
 - C TCS Command Station (Using CAN via GC Network Interface)
 - D Some other Wifi Hub such as the OpenMRN Hub.

RR-CirKits Hardware

computer
interface



Dick Bronson
CEO

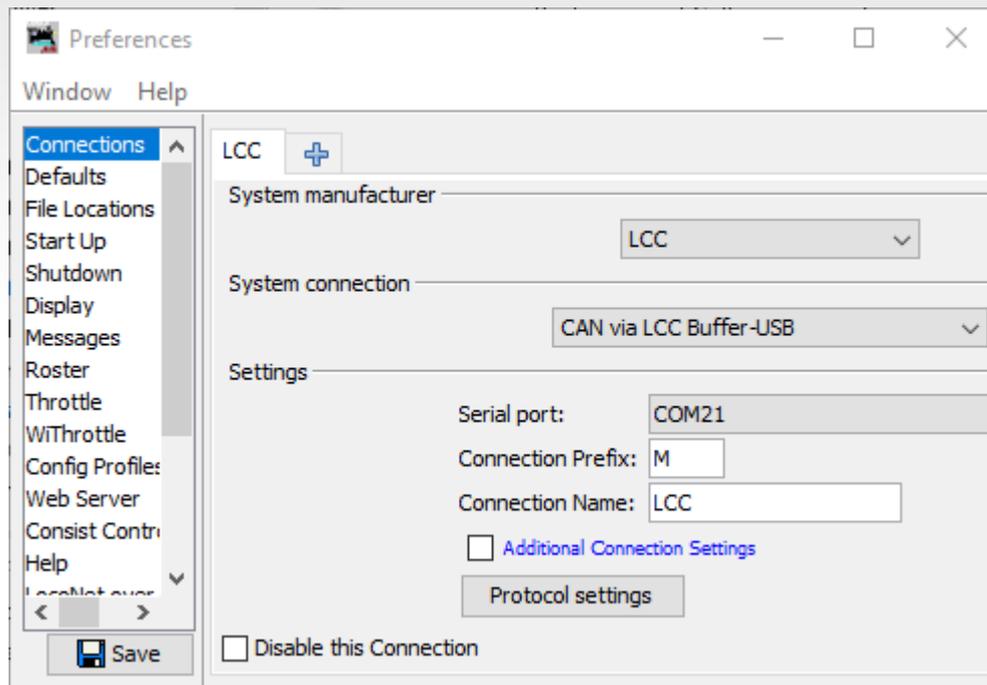
RR-CirKits, Inc.
Specializing in Affordable Electronics for Model Railroads

7918 Royal Ct.
Waxhaw, NC 28173
www.rr-cirkits.com

Phone: 1-704-843-3769
Fax: 1-704-243-4310
dick@rr-cirkits.com

Configure JMRI Preferences

- Select JMRI Edit → Preferences → Connections

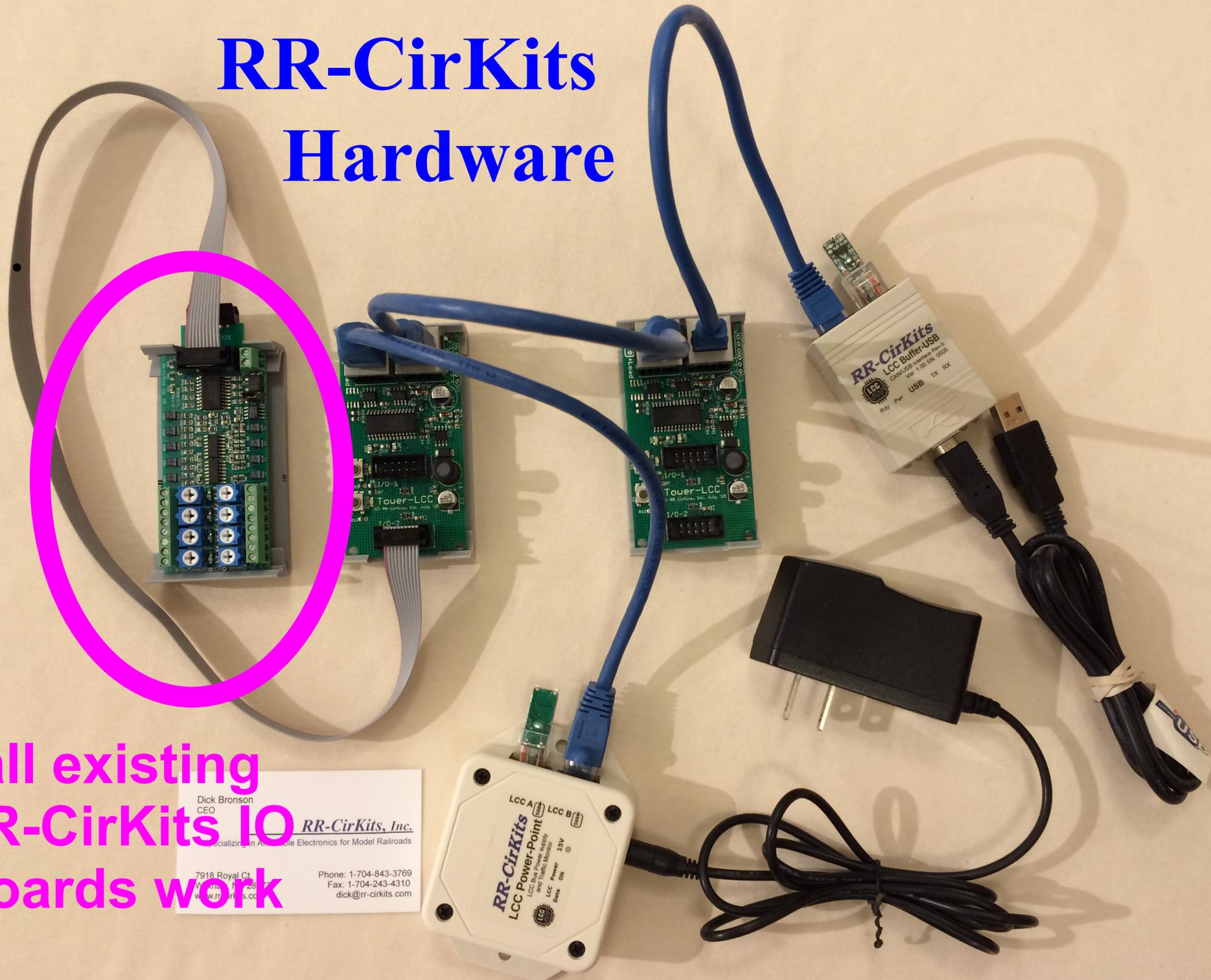


- System manufacturer: LCC
- System Connection:
CAN via LCC Buffer-USB
- Serial Port: to match
- Save this configuration
- Restart JMRI

Applications

- Finally we make some connection to simple layout hardware such as:
 - Detectors
 - Buttons
 - Turnouts
 - Cross bucks
 - Indicators
 - Signals
 - Lights
 - Etc.

RR-CirKits Hardware



all existing
RR-CirKits IO
boards work

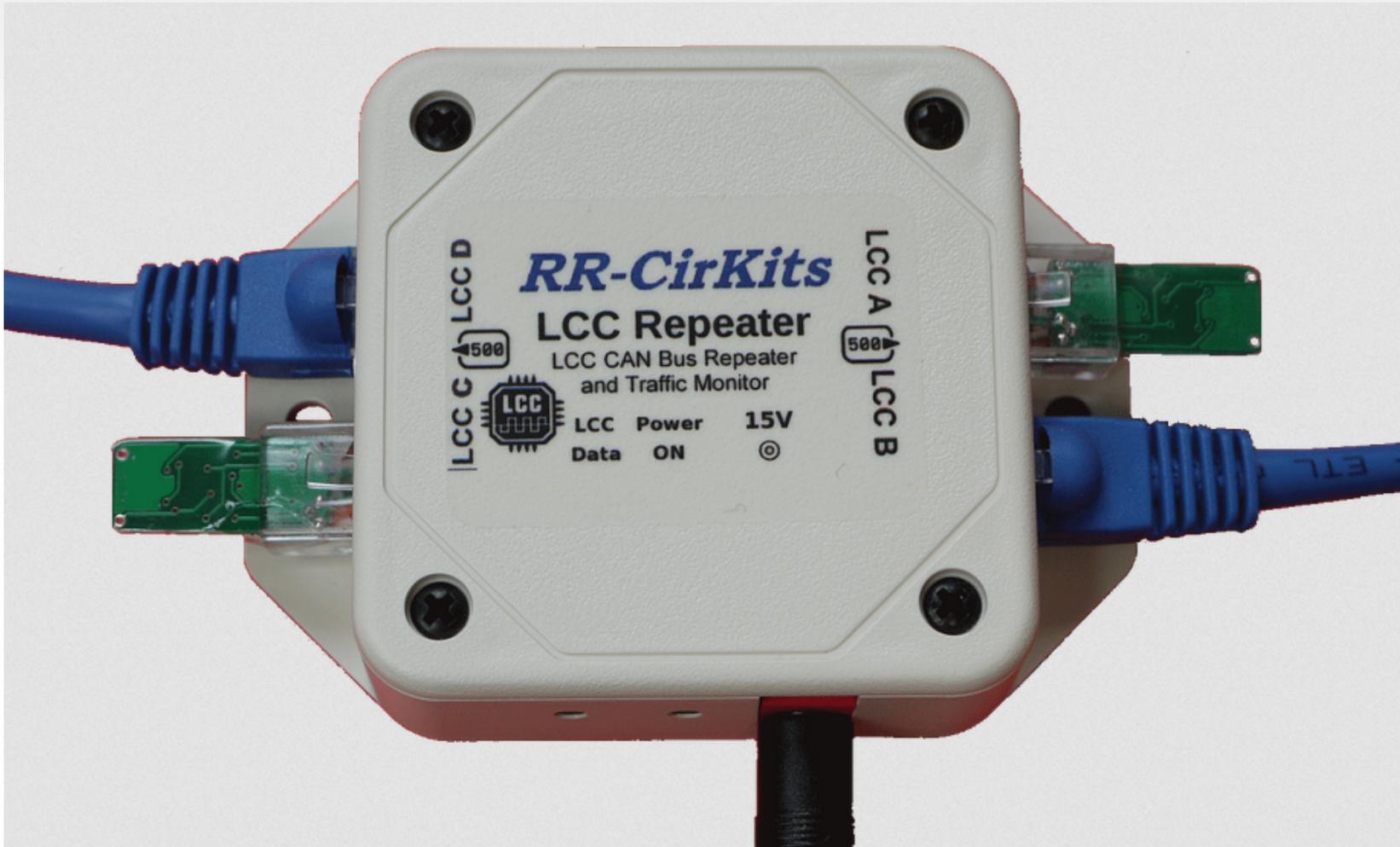
Dick Bronson
CEO
RR-CirKits, Inc.
Specializing in Affordable Electronics for Model Railroads
7918 Royal Ct.
Waukegan, IL 60087
www.rr-cirkits.com
Phone: 1-704-843-3769
Fax: 1-704-243-4310
dick@rr-cirkits.com

Applications

- Or more complex devices such as:
 - Repeaters
 - Fast clocks
 - Gateways
 - Command Stations
 - Boosters
 - Throttles
 - Power Managers
 - Trains
 - Computers

Applications

- Repeater



Applications

- Fast Clock

Logic Rail

Technologies has just released their LCC Fast Clock.

This unit takes full advantage of the LCC protocol to

support vastly improved capabilities over their similar products for legacy systems. Their NCE version is a repeater for the NCE cab fast clock. Their LocoNet version adds the capability to be configured to trigger up to four different LocoNet commands. The LCC version turns that option on its head, and allows any LCC node to be configured to be triggered at any of the 1,440 possible fast minutes in a fast day. Any LCC Fast Clock attached to the LCC bus may be a master or a slave. The LCC Fast Clock is configured over the bus.



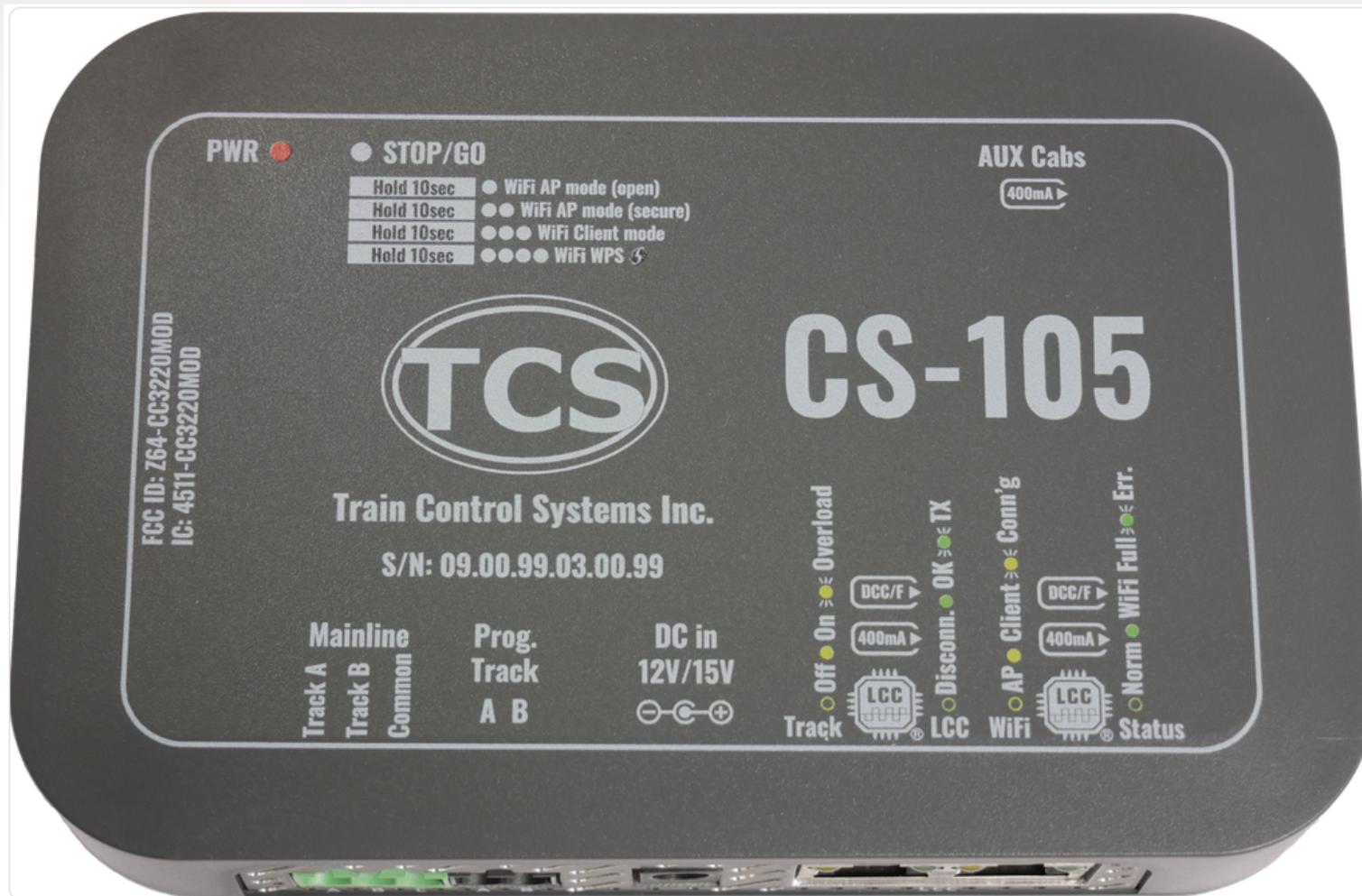
Applications

- Gateways



Applications

- DCC Command Station



Applications

- Wireless Throttles (LCC over Wifi)

The TCS UWT-100 can operate in two different WiFi modes. Most current users use it to communicate with their existing command stations using the JMRI WiThrottle protocol. I suspect that many owners don't even realize that it also supports direct LCC traction protocol over WiFi, and that it can be configured over the LCC network.



Applications

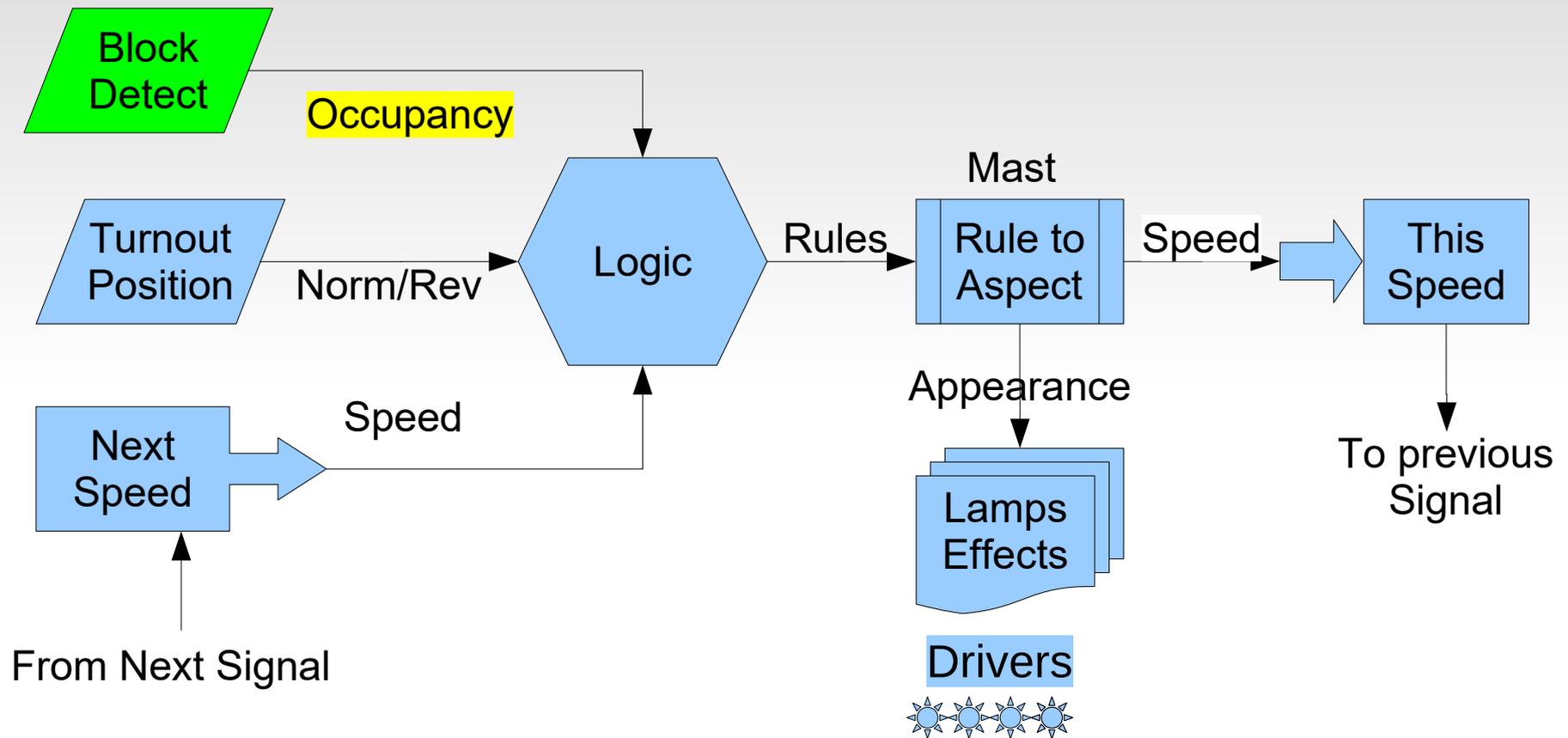
- JMRI Sensors

JMRI includes a handy tool at the bottom of the CDI window to make sensors or turnouts from events. LCC Nodes may use two (or more) EventIDs to control sensors and turnouts, so you must use cut/paste to choose the pair that you want for JMRI. For this sensor we use Event 1 and Event 2 that we just defined.

Screenshot of the JMRI Sensor/Turnout creation dialog box. The dialog has a title bar "Sensor/Turnout creation". It contains a "User name" field with "Whithead West Main 1". Below it are two "Event ID" fields: "Event ID for Active / Thrown" with "02.01.57.10.00.00.06" and "Event ID for Inactive / Closed" with "02.01.57.10.00.06.00.07". Each event ID field has "Copy" and "Paste" buttons. At the bottom of the dialog are "Make Sensor" and "Make Turnout" buttons. Below the dialog is a row of buttons: "Refresh All", "Save changed", "Backup...", "Restore...", "Make All Sensors", and "Make All Turnouts". A green arrow points from the "User name" field to the "Make Sensor" button. A pink arrow points from the "Event ID for Active / Thrown" field to the "Make Sensor" button. Another pink arrow points from the "Event ID for Inactive / Closed" field to the "Make Sensor" button.

Enter the JMRI user name for this sensor, (or turnout) then click on the Make Sensor button. This item will automatically be added to your JMRI Sensor (or turnout) table. Be sure to save the table for future use. Normally this data will become part of a 'Panels' file, and be synchronized with the node when loaded.

Block Detect Example



The BOD4 and BOD4-CP cards each include 4 block detector circuits for easy connection to the Signal LCC board. These boards use CT coils to prevent track voltage drop and provide 100% isolation.

Block Detector Variables

- Variables

Variables are used to follow the state of objects of interest such as block detectors, turnout positions, etc. Normally two events will allow the variable to follow the state of some object, true/false, normal/diverging, clear/occupied, etc.

Select Input/Output line.

Line 1 (Whithead West Main 1) | Line 2 (Whithead West Main 2) | Line 3 | Line 4 | Line 5 | Line 6

I/O

Line description
Whithead West Main 1 | Refresh | Write

Output Function
No Function | Refresh | Write

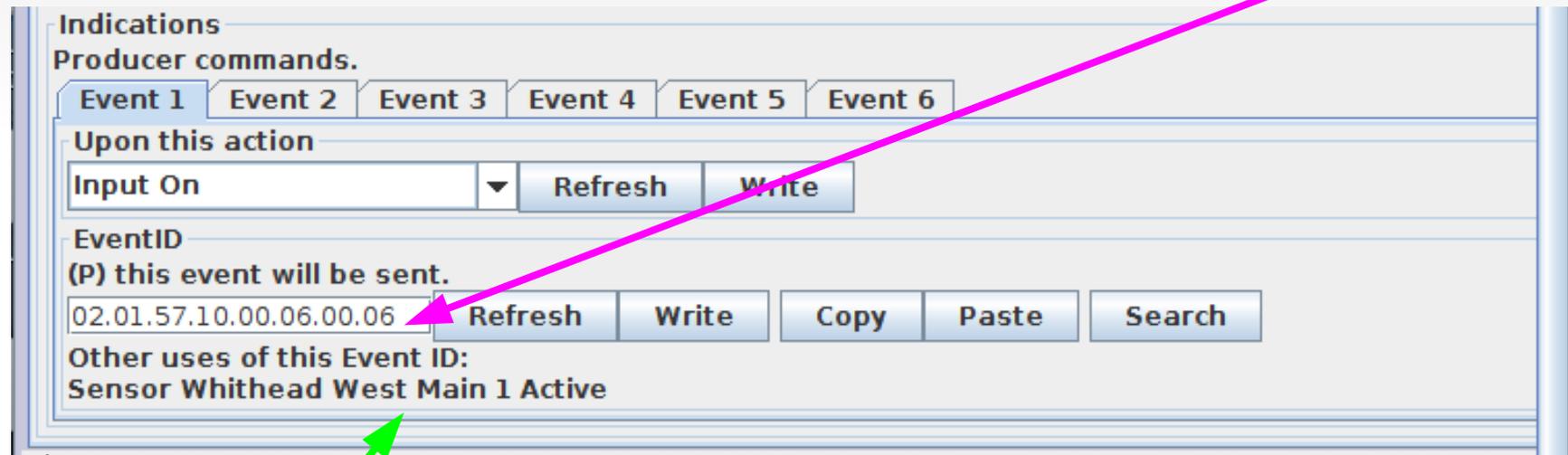
Input Function
Active Lo | Refresh | Write

Lets start by connecting a block sensor to an input. Its description is 'Whithead West Main 1' so we enter it in the description block and 'Write' it to the node. Detectors are Input Functions with 'Active Lo' so we set that and write it. For a normal Input be sure that the Output Function is set to 'No Function'. Of course 'Line 1' is the one connected to our first block detector.

Block Detector Variables

- Input (Producer) Events

We now go to the Indications (Producers) for this line, and enable two events. The first (Event 1) will be sent when the Input is 'On' (goes low in our application) When we need to know if the block goes occupied, we will use this EventID.



utilize the magic numbers (EventID) for these events. Its description 'Whithead West Main 1' is noted here to remind you of its function. This information is known because I made a JMRI sensor that follows it. This is a JMRI feature available in the JMRI CDI tool.

Block Detector Variables

- JMRI Sensors

JMRI includes a handy tool at the bottom of the CDI window to make sensors or turnouts from events. LCC Nodes may use two (or more) EventIDs to control sensors and turnouts, so you must use cut/paste to choose the pair that you want for JMRI. For this sensor we use Event 1 and Event 2 that we just defined.

Sensor/Turnout creation

User name
Whithead West Main 1

Event ID for Active / Thrown
02.01.57.10.00.06 00.06 Copy Paste

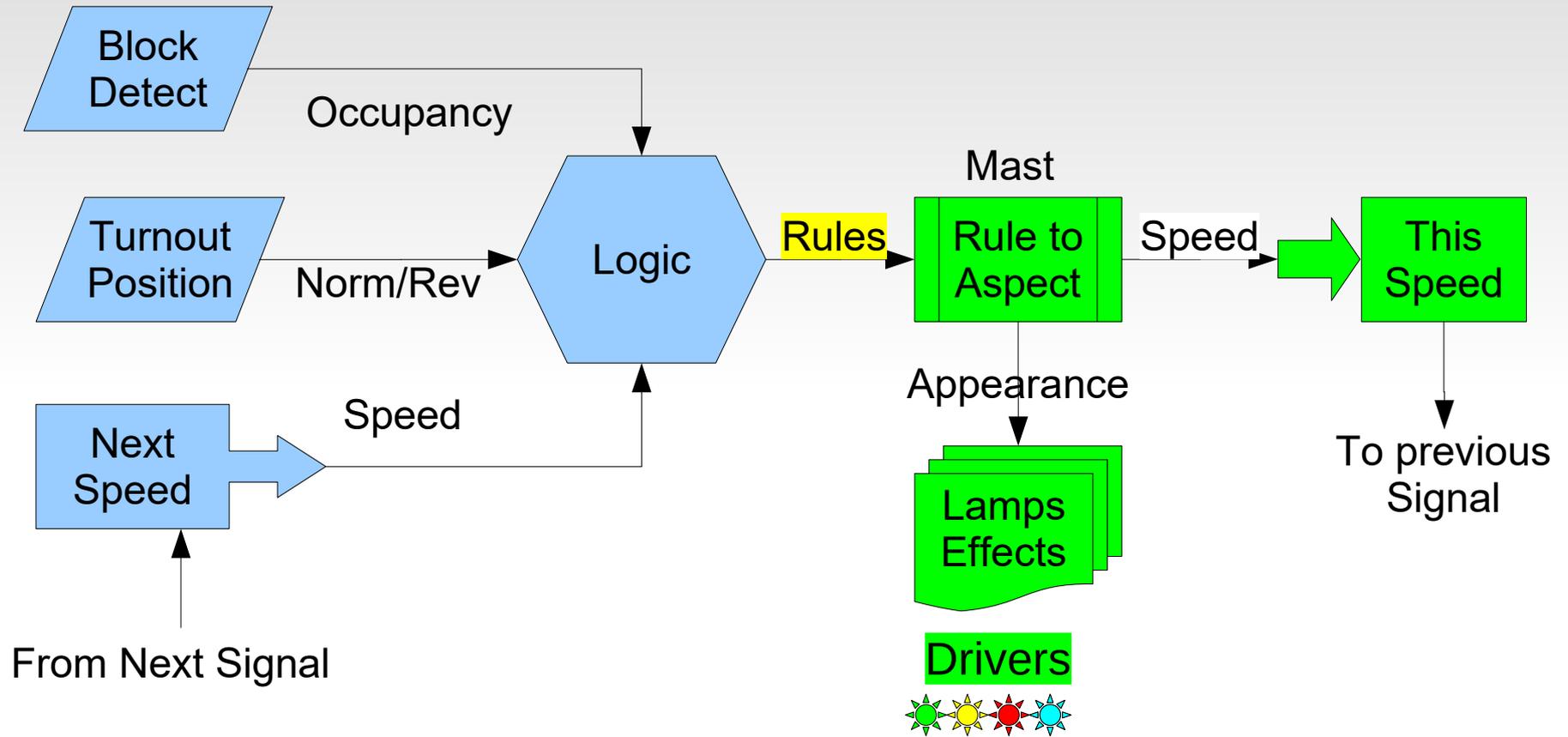
Event ID for Inactive / Closed
02.01.57.10.00.06.00.07 Copy Paste

Make Sensor Make Turnout

Refresh All Save changed Backup... Restore... Make All Sensors Make All Turnouts

Enter the JMRI user name for this sensor, (or turnout) then click on the Make Sensor button. This item will automatically be added to your JMRI Sensor (or turnout) table. Be sure to save the table for future use. Normally this data will become part of a 'Panels' file, and be synchronized with the node when loaded.

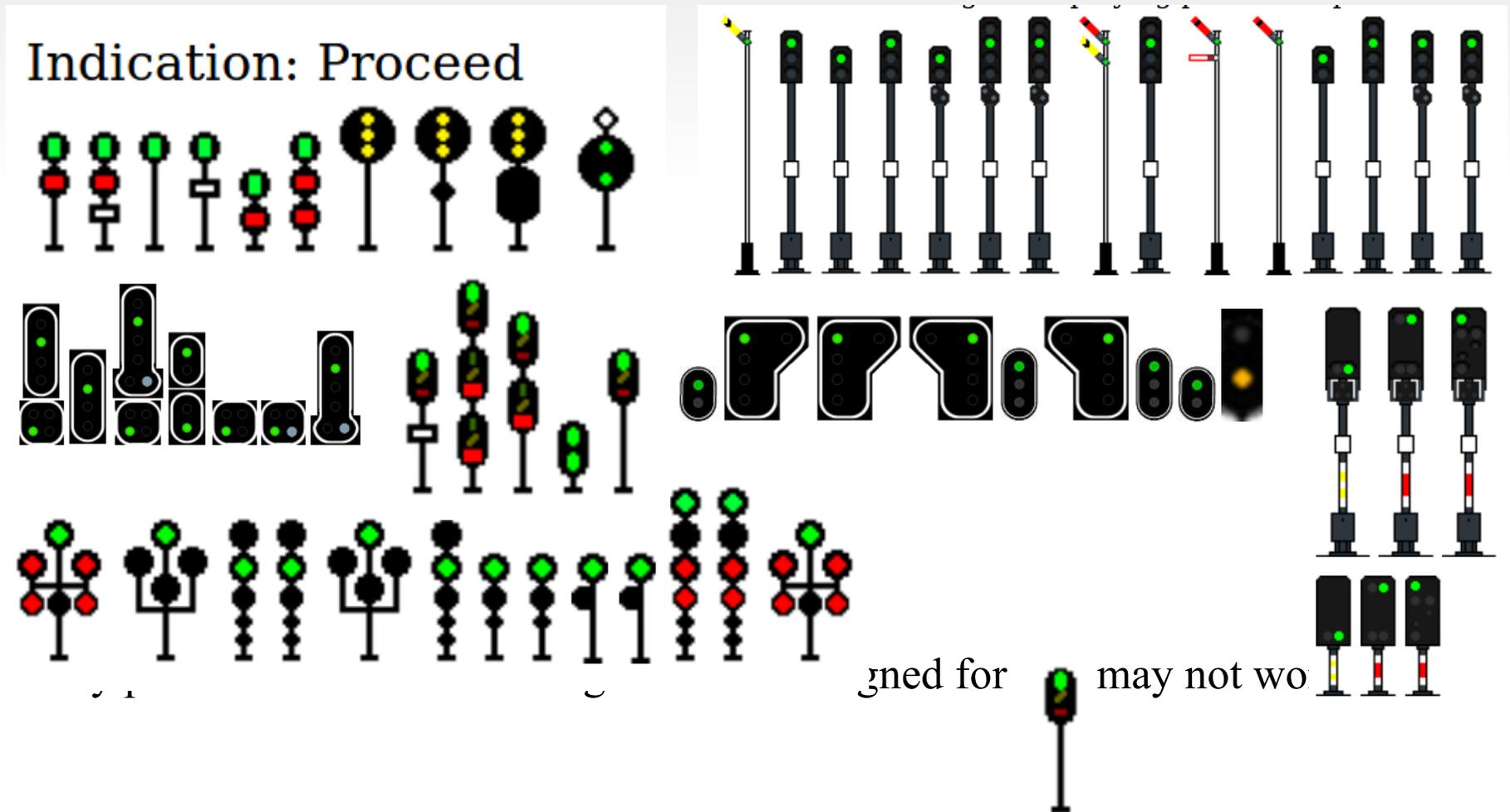
Rule to Aspect



Signaling Requirements

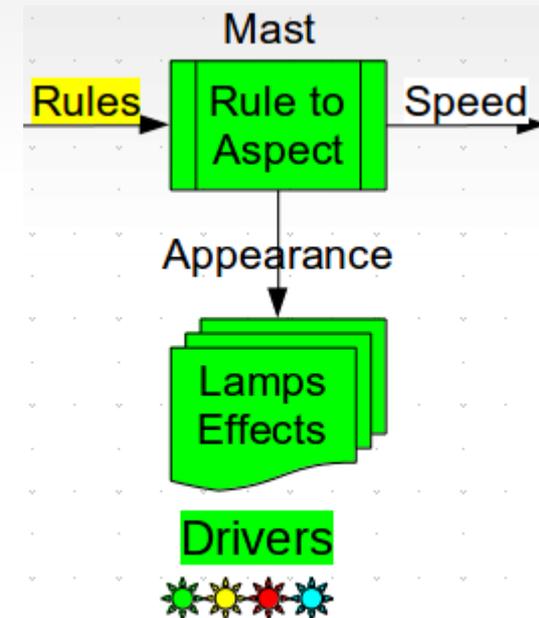
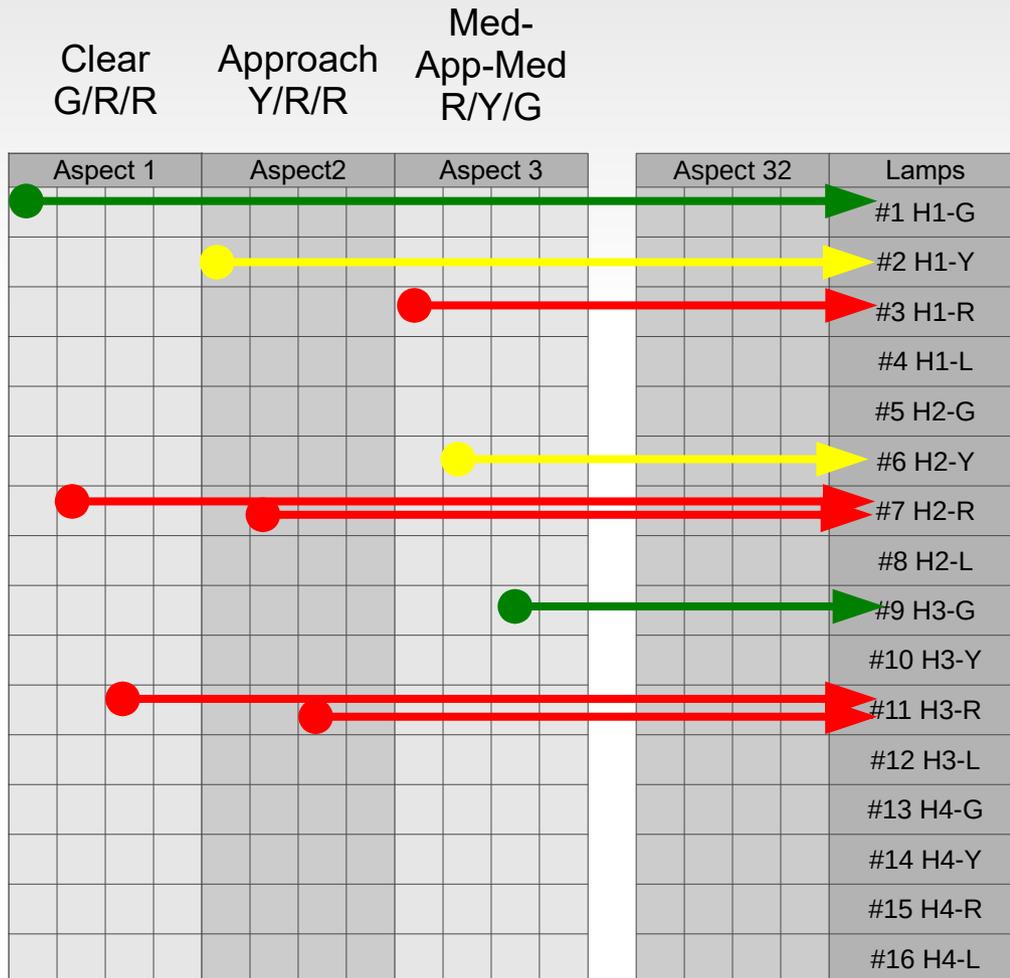
- Rule to Aspect conversion

Signal rules such as 'Stop', 'Approach', 'Clear', Etc. are displayed differently on different types of signals. A simple way to make these conversions is needed.

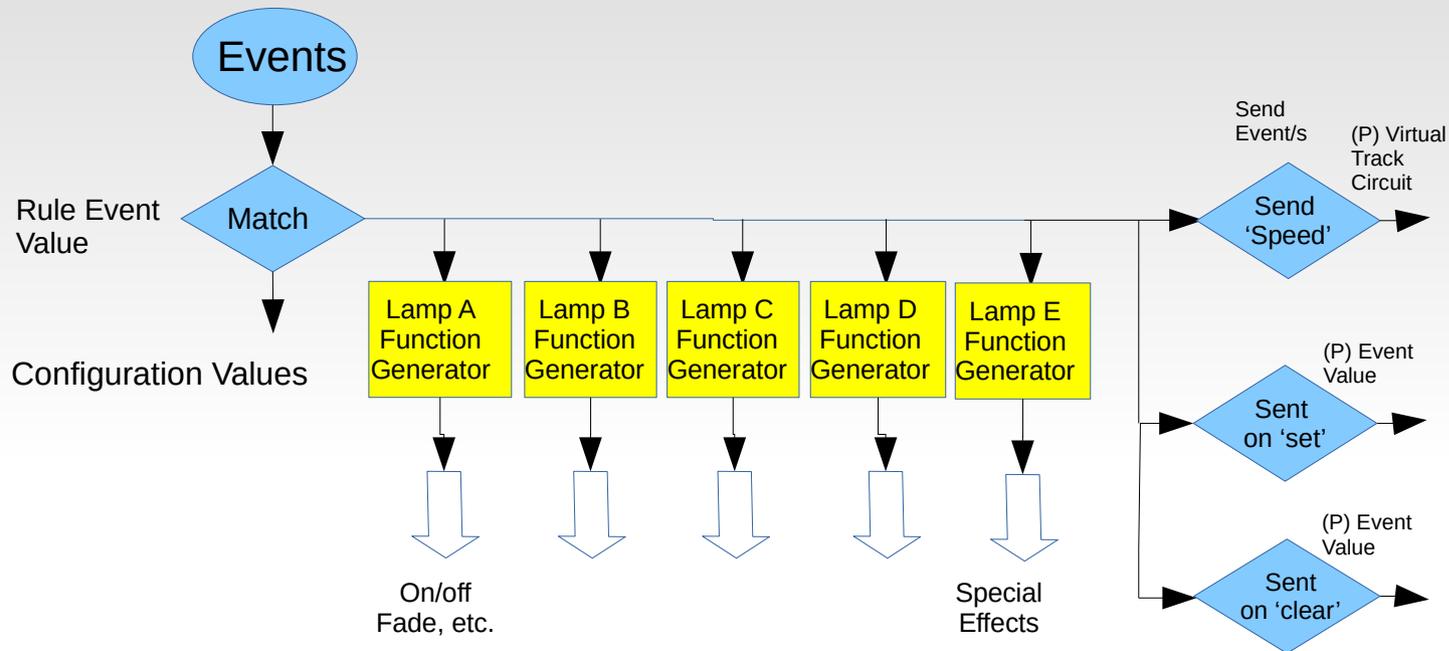


Aspects to Appearances

- Our solution may be visualized as a grid. Each (of 32) aspects has 4 columns that can each control any one of the 16 lamp drivers.



Rule to Appearance



■ Signal Masts

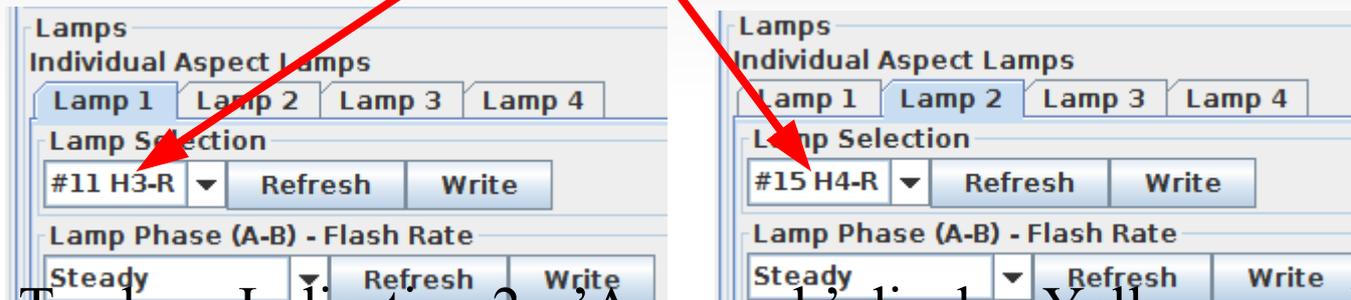
This flow chart shows the Mast functionality in a different way. Any signal rule that is seen (matched) can send up to 4 lamp control messages, plus an optional special effect. Speed is sent by the virtual track circuit, and it also sends 2 optional events. These optional events may be used to send indications back to a CTC panel, or allow some other process to monitor aspect changes.

Signal Masts

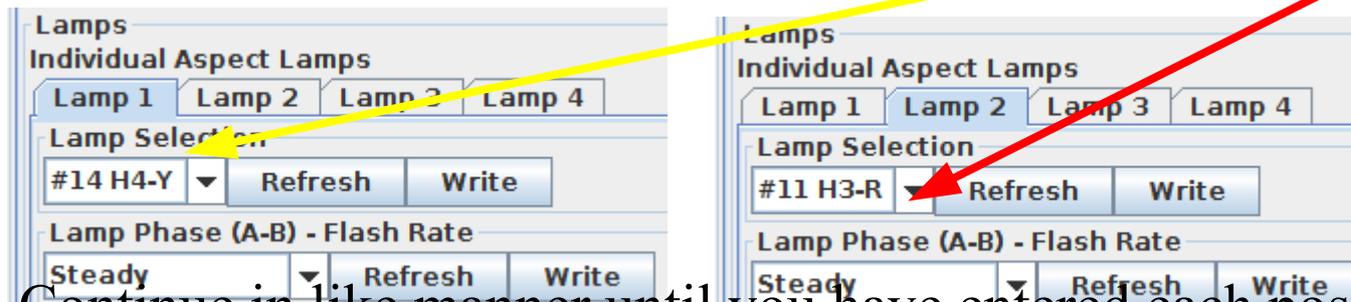
- The EventIDs sent and responded to by each rule are also controlled by the MASTS segment. Because this Rule to Aspect conversion is actually the links between the 'Rule' events and the actual hardware we call it all 'MASTS' and treat it as one segment in the CDI.
- The Signal LCC supports 8 Masts, each of which supports 8 indications. If a single mast requires more than 8 aspects, then any mast may be logically linked with a previous one. (Signal-Mini is 16 x 4)
- A 'Mast' definition makes two assumptions.
 - 1) Only one aspect may be shown at a time. Setting any aspect automatically cancels any previous aspect.
 - 2) A mast may only have a single speed limit at a time. This 'Speed' is the speed allowed when passing the mast.
- Making a mast 'Linked to Previous' carries the above assumptions over from any previous mast/masts. Speed is always taken from the first mast.

Lamps

- The bottom line in displaying each aspect is to choose what lamps are lit. After all, that is what the crew (and the visitors) actually see.
- The mast on the signal bridge is dual head searchlights. This means 'Stop' will display Red over Red. Pick the appropriate lamps to show this.



- To show Indication 2 - 'Approach' display Yellow over Red.



- Continue in like manner until you have entered each possible aspect.

Signal Lamp Drivers

- Different colors of LEDs have different voltage drops. This drop is subtracted from the drive voltage when calculating the series resistance. A typical red LED operates at 1.9V and a green LED operates at 3.3V. At 5V the red resistor drops 3.1V and the green resistor drops 1.7V. With equal values, the red LED will draw nearly twice the current as the green. Using a 12V source, the resistor voltage drops are 10.1V and 8.7V respectively, or just a 15% difference in current.
- Sometimes it is easiest to wire 2 LEDs in series for a signal. The voltage drops of green and yellow LEDs make it difficult or impossible to drive these in series with 5V supplies. As a result all of our RR-CirKits signal driver boards have always supplied 10V or more to the drive circuits.

Signal Lamp Drivers

- Each Indication (Aspect) can be displayed with as many as four lamps. If you have a rare signal aspect that can not be shown with just 4 lighted lamps you can make a duplicate mast to light any additional lamps. Remember dual lamps that light together only count as a single lamp. (e.g. in Position Lights and Color Position lights) Only lighted lamps count. Only controlled lamps count. A marker that is always lighted can simply be powered full time.
- Lamp Phase – Flash Rate may be used to flash signals automatically. One common example is 'Advance Approach' which is commonly displayed with a flashing yellow lamp. Setting an appropriate Flash Rate means that the signal logic doesn't need to worry about flashing the signal or overloading the bus with unnecessary traffic. Providing both A and B phase options is handy for grade crossing flashers or other alternating lamp situations.

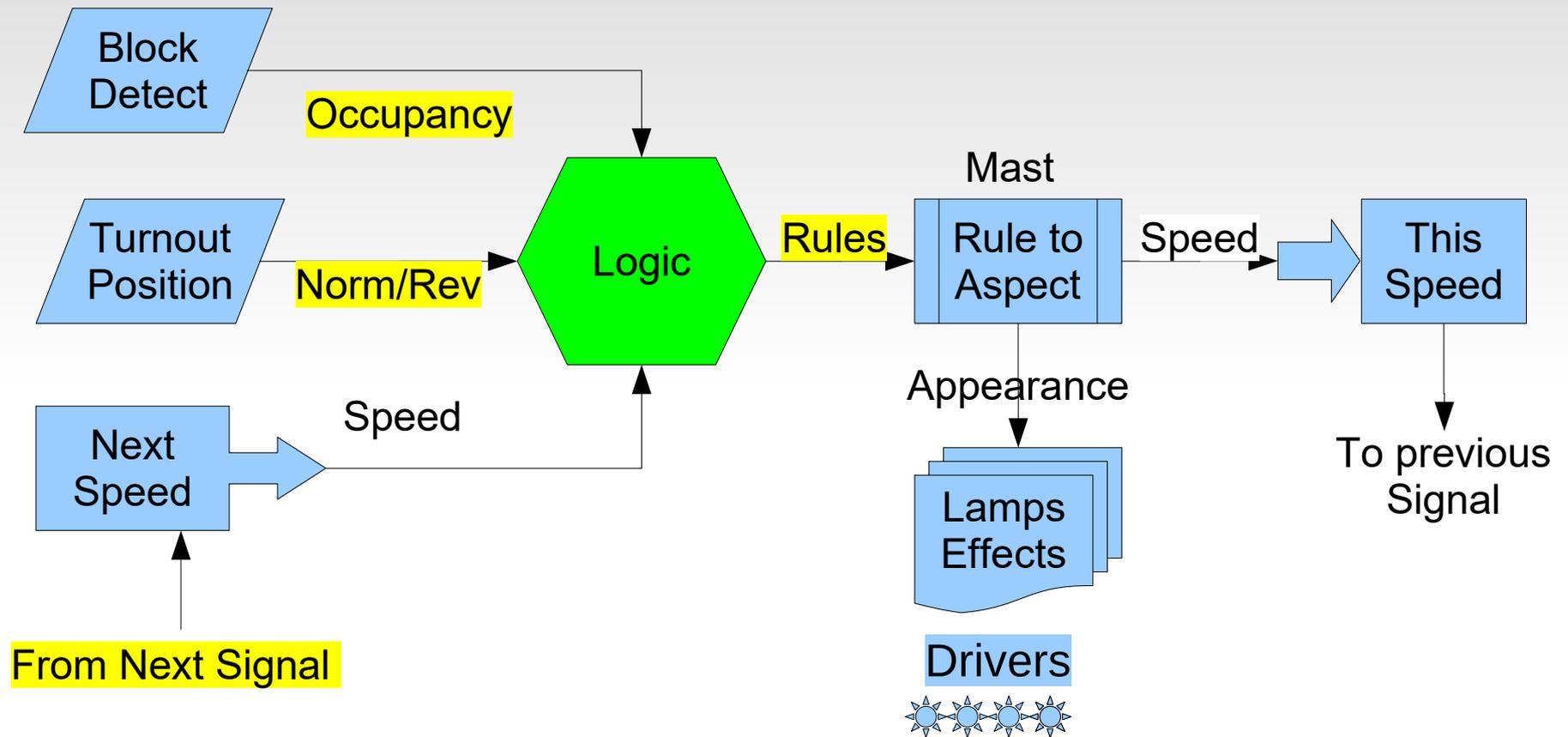
Lamps - Special Effects

- Incandescent fade. Signal lamps are wired differently than standard household lamps. They include a ballast resistor in series with the lamp. This ballast serves two purposes. One is simply to set the brightness of the lamp. More importantly when a cold lamp is first powered up it prevents the normal inrush current by dropping most of the voltage across the ballast until the lamp warms up. The visual result of this is that a signal does not blink on instantly. In fact signals lamps fade on slowly enough to be noted. Of course even household incandescents fade off slowly as the lamps cool down again.
- Transition effects. The B&O signal clip we saw earlier shows an interesting transition between Clear and Stop. Not only does it show the fade up and down, but it interjects a brief 'Approach' into the change. I can not remember why this is done, but selecting 'Transition Down' as a special effect on 'Stop' will allow you to do this. (and wow your rivet counting crew)

Lamps - Special Effects

- H2 Red Flicker. Many of you know that many searchlight signals (not just the H2) have an internal arm that swings back and forth in front of the lamp. It hangs by gravity with a red roundel in center position. Either green or yellow requires swinging the arm out of its center position with electromagnets. Not quite as obvious is the fact that you can not change between yellow and green without passing the red between them. This causes the red flash. The other part of the effect is that the arm is free swinging and during a change it will overshoot its position as it settles down. This swings the color roundels past their normal positions which causes the signal to flicker off briefly.
- Strobe lights can be found around the layout. Sometimes it is nice to be able to utilize unused signal outputs for other purposes.

Signal Logic



We have covered all the edges. Now we can talk about the central subject, Signal Logic itself.

Signal Logic

- Signal Logic is just a series of conditions (called conditionals) that are checked to see what signal rule should currently be in effect.
- Logic conditionals should be easy to cascade with the calculations for the most restrictive rules having priority over less restrictive rules. We do this by checking each conditionals in order from top down. Any rule that is found to be true first checks for any more restrictive rule still in effect. (which exits processing if found) Then it sends its appropriate events, and finally skips over any less restrictive rules for the mast.
- Built in logic conditionals may directly send events representing signal rules (or anything) when it is found to be true. (or false) A cascade option allows even more events to be sent in special situations. Note: this logic may be used for many other purposes than just calculating signal aspects.

Internal Signal Logic

- Logic Functions consist of the usual AND, OR, XOR operators. In addition there are two 'change' operators. These change the true/false sense of a conditional based on the AND and OR of the variables.
- Additionally we have added a non-standard logic operator called 'AND Then'. This makes it very easy to keep track of train direction. You can simply watch two block detectors and determine train direction by the order in which they are activated.
- A recent addition is the ability to control the action associated with both true and false evaluations of a conditional. These options are to 'Send then Exit Group', 'Send then Evaluate Next', 'Exit Group', and 'Evaluate Next'.

Internal Signal Logic

- In order to keep this presentation under two or three hours we will leave the configuration of a node using the built in logic for another day.
- Maybe we should consider presenting a 'Make and Take' clinic where a user can configure and take home a signal interlocking node.
- One of the early LCC adopters present in this room is using LCC internal node logic to create an operating, hardware based CTC panel, using LCC nodes and their internal logic. This is really good for each of you here, because he has put his finger on numerous places where improvements needed (still need?) to be made in the logic.
- The advantage of logic in nodes over DIY solutions, is that the user can configure it after the node has been installed on the layout.

JMRI Signal Control

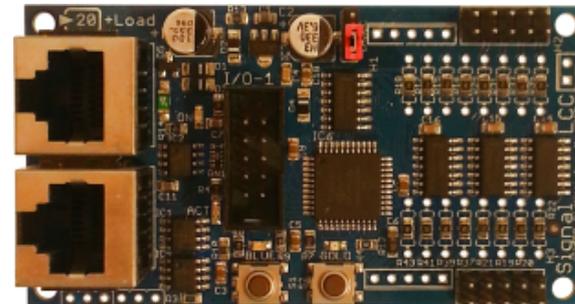
- The most recent addition to JMRI is an OpenLCB controlled signal mast. We will show how to use that tool which allows you to take advantage of the built in JMRI signal rules for many popular railroads.
- The disadvantage of JMRI for some users such as museums and clubs is that a computer running JMRI becomes a permanent and necessary part of the signaling system, and usually the signals don't run while the system is under development. This nullifies some key LCC advantages, which include updates and configuration changes to a running system.

JMRI OlcbSignalMast

- The new JMRI OlcbSignalMast was actually added to JMRI 4.11.2 in January 2018. Unfortunately the GUI for this mast is still under development at the time of this clinic. (August 7th, 2018)
- This clinic shows the use of this new GUI based on a development version which hopefully will be publicly available soon. (4.13.2?) As you will readily see, the new GUI tool itself works properly. The issue is that it required other improvements to JMRI that are still untested, and/or cause issues with some existing systems.
- This new mast is a true event driven mast. Previous masts have been based on control by ‘turnouts’ or decoders.

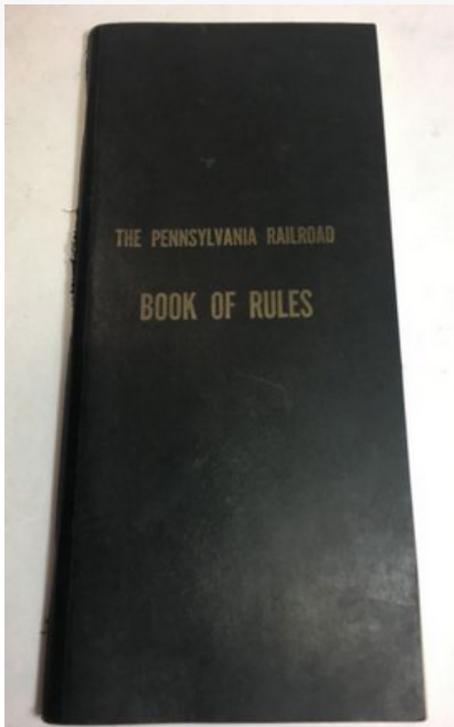
JMRI OlcbSignalMast

- The first step is to determine what signal system you are using.
- In this example it is a pair of double head PRR signals. These are found in JMRI under the PRR 1957 rules.
- Next determine the control system. To match the clinic title we will use an LCC signal driver node.



JMRI OlcbSignalMast

- Then
 - a) Locate a rule book.
or
 - b) Locate a rule set in JMRI.



The screenshot shows a Mozilla Firefox browser window with the title "JMRI: PRR-1956 Signaling Definition". The address bar contains "jmri.org/xml/signals/PR". The page features a purple header with the JMRI logo and a search bar. Below the header, there is a section titled "JMRI: PRR-1956 Signaling Definition" with the following text:

This directory contains signaling definitions corresponding to the 1956 PRR rules as described in the August 1956 version of the PRR publication "RULES FOR CONDUCTING TRANSPORTATION ", Signals.

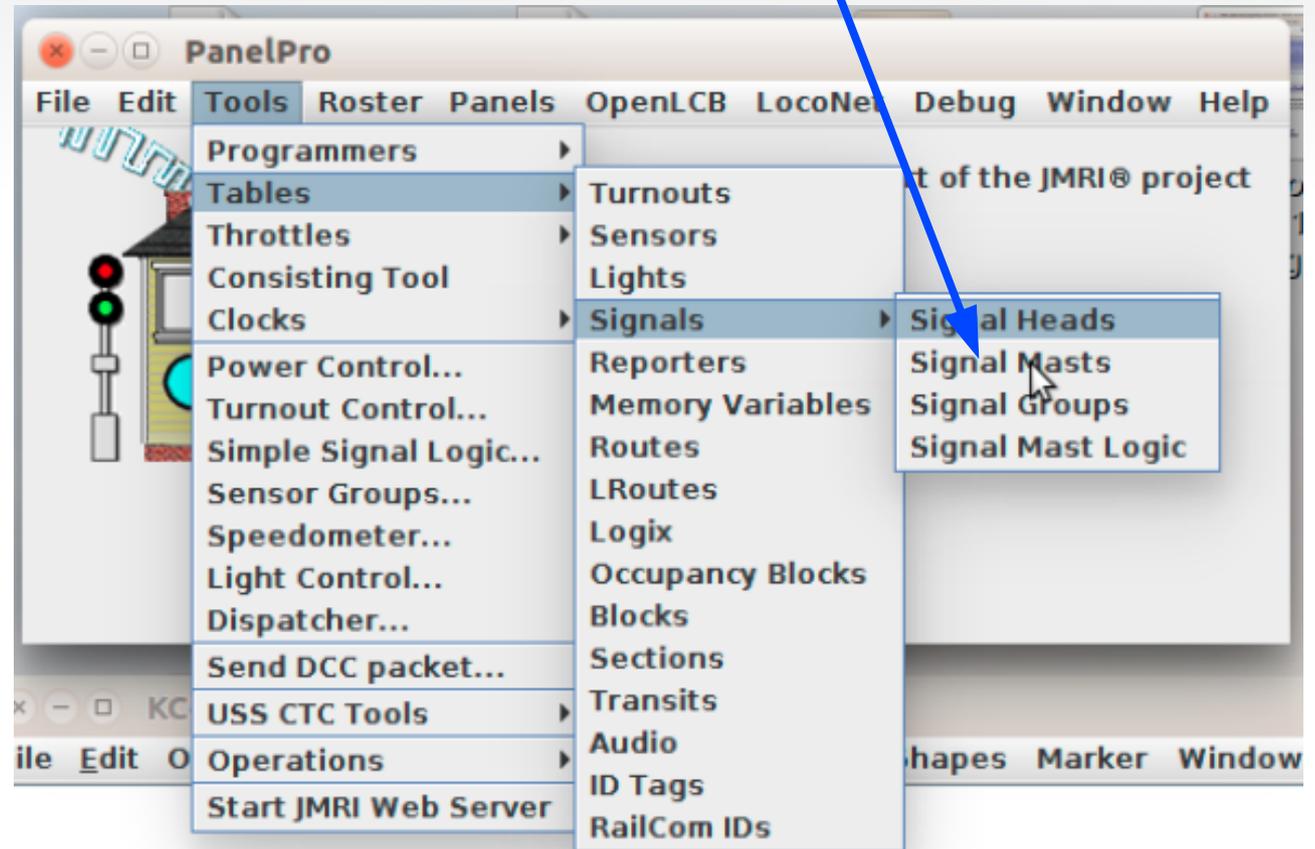
See the [aspect page](#).

Signal Mast definitions:

- [Single position light high absolute signal](#)
- [Single position light high permissive signal](#)
- [Double position light high signal](#)
- [Single position light dwarf signal](#)
- [Double position light dwarf signal](#)

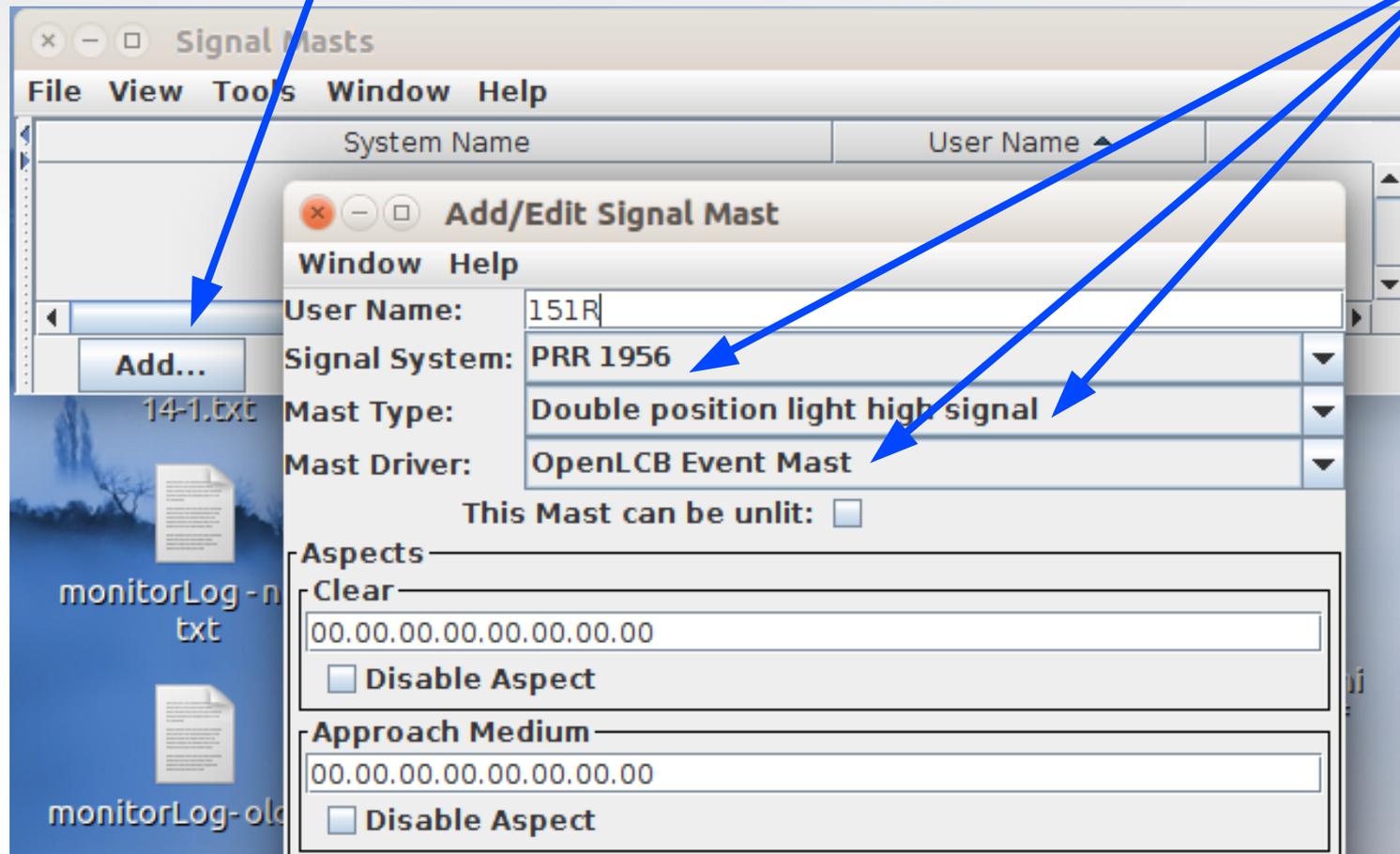
JMRI OlcbSignalMast

- Open the JMRI Signal Masts creation tool.



JMRI OlcbSignalMast

- Click on Add.
- Fill in the required information to select the OLCB mast.



JMRI OlcbSignalMast

- Open the CDI for the controller node.
- Configure each rule to match the required appearance.

The screenshot shows the configuration window for 'Configure Cressman West - 151R and 152R (02.01.57.10.00.08)'. The window is divided into several sections:

- Segment: Rule to Aspect**
- Select Mast**: A row of tabs for Mast 1 (151R) through Mast 8. Mast 1 (151R) is selected.
- Function**: Mast Processing dropdown set to 'Normal', with 'Refresh' and 'Write' buttons.
- Mast ID**: Text field containing '151R', with 'Refresh' and 'Write' buttons.
- (P) Track Circuit Link Address. Copy and Paste into linked Track Circuit. (Read Only)**: Text field containing '02.01.57.10.00.08.01.A0', with 'Refresh', 'Write', 'Copy', 'Paste', and 'Search' buttons.
- Other uses of this Event ID:** Cressman West.Rule to Aspect.Select Mast(1,151R).(P) Track Circuit Link Address. Copy and Paste into linked
- Lamp Fade**: Incandescent dropdown, with 'Refresh' and 'Write' buttons.
- Rules**: A row of tabs for Rule 1 through Rule 8. Rule 1 is selected.
- Name**: Text field containing '0-Stop', with 'Refresh' and 'Write' buttons.
- Track Speed (on approach to signal)**: Text field containing 'Stop', with 'Refresh' and 'Write' buttons.
- EventID**: (C) Event to Set Aspect. Note: Aspects are cleared automatically by the logic. Text field containing '02.01.57.10.00.08.01.A8', with 'Refresh', 'Write', 'Copy', 'Paste', and 'Search' buttons.
- Other uses of this Event ID:**

JMRI OlcbSignalMast

- Place the CDI and mast next to one another on your screen.
- In the CDI at the EventID that sets each aspect, Click [Copy].
- Then double-click to select the aspect event, and Ctrl-V to paste the EventID into the Add Mast Window.

Cressman West.Rule to Aspect.Select Mast(1,151R).(P) Track Circuit Link Address. C

Lamp Fade
Incandescent Refresh Write

Rules
Rule 1 Rule 2 Rule 3 Rule 4 Rule 5 Rule 6 Rule 7 Rule 8

Name
0-Stop Refresh Write

Track Speed (on approach to signal)
Stop Refresh Write

EventID
(C) Event to Set Aspect. Note: Aspects are cleared automatically by the logic.
02.01.57.10.00.08.01.A8 Refresh Write Copy Paste Search

Other uses of this Event ID:
Cressman West.LOGIC.Logic event processing(1,151R Stop).A trigger or change will
Cressman West.LOGIC.Logic event processing(1,151R Stop).A trigger or change will
Cressman West.Rule to Aspect.Select Mast(1,151R).(P) Track Circuit Link Address. C

Approach Slow
00.00.00.00.00.00.00.00.00
 Disable Aspect

Approach
00.00.00.00.00.00.00.00.00
 Disable Aspect

Slow Approach
00.00.00.00.00.00.00.00.00
 Disable Aspect

Restricting
00.00.00.00.00.00.00.00.00
 Disable Aspect

Stop
02.01.57.10.00.08.01.A8
 Disable Aspect

Continue to copy and paste for each aspect.

JMRI OlcbSignalMast

- Once each EventID has been added to the mast click on [Create] to enter it into the Mast Table.

The screenshot shows a dialog box with the following structure:

- Lit/UnLit**
 - Lit:** 00.00.00.00.00.00.00.00
 - Not Lit:** 00.00.00.00.00.00.00.00
- Held/UnHeld**
 - Held:** 00.00.00.00.00.00.00.00
 - Not Held:** 00.00.00.00.00.00.00.00

At the bottom of the dialog are two buttons: **Cancel** and **Create**. A red circle highlights the **Create** button, and a red arrow points from it to the **[Create]** text in the first bullet point of the text above.

- With LCC as soon as you click [Create] the mast appears in the Mast Table and is live on the layout.
- Be sure to save your panel before closing down JMRI. Your LCC mast will retain its information, but JMRI will not.

JMRI Panel

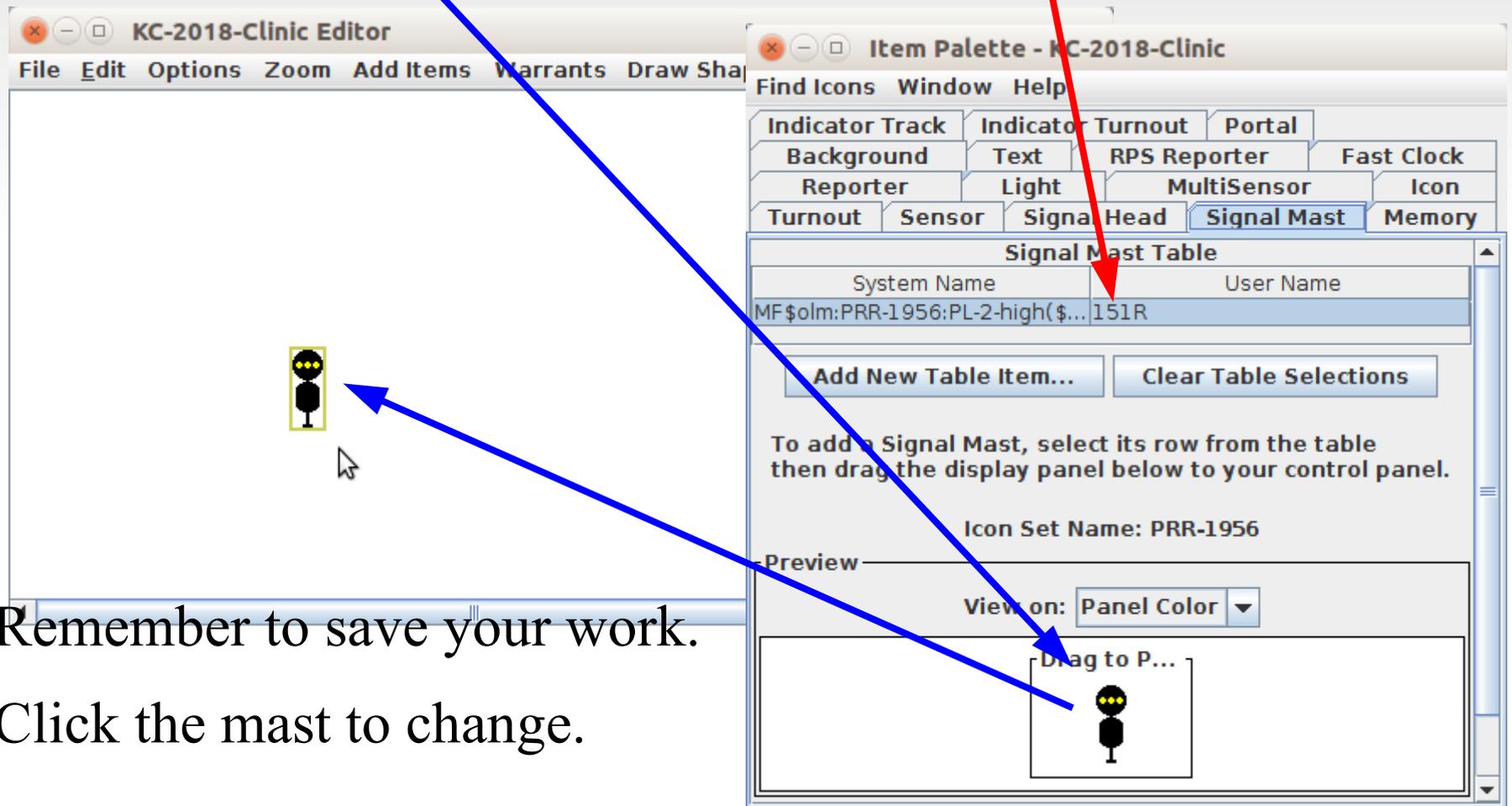
- Now open a Panel in JMRI.
- Select 'Add Items' then 'Item Palette'.

The screenshot shows the JMRI software interface. The main window is titled 'KC-2018-Clinic Editor' and has a menu bar with 'File', 'Edit', 'Options', 'Zoom', 'Add Items', 'Warrants', and 'Draw Shape'. The 'Add Items' menu is open, showing options: 'Item Palette' (with a red arrow pointing to it), 'Occupancy Blocks' (with 'Ctrl-B' as a shortcut), and 'Item Tables' (with 'Ctrl-T' as a shortcut). A second window titled 'Item Palette - KC-2018-Clinic' is open in the foreground. It has a menu bar with 'Find Icons', 'Window', and 'Help'. The main area contains several tabs: 'Indicator Track', 'Indicator Turnout', 'Portal', 'Background', 'Text', 'RPS Reporter', 'Fast Clock', 'Reporter', 'Light', 'MultiSensor', 'Icon', 'Turnout', 'Sensor', 'Signal Head', 'Signal Mast' (highlighted with a blue arrow), and 'Memory'. Below these tabs is a table titled 'Signal Mast Table' with columns 'System Name' and 'User Name'. The first row contains the text 'MF\$olm:PRR-1956:PL-2-high(\$... 151R'. Below the table are two buttons: 'Add New Table Item...' and 'Clear Table Selections'. A text block reads: 'To add a Signal Mast, select its row from the table then drag the display panel below to your control panel.' Below this is a label 'Icon Set Name: PRR-1956'. At the bottom, there is a 'Preview' section with a 'View on:' dropdown menu set to 'Panel Color'. Below the preview is a box labeled 'Drag to P...' containing a signal mast icon.

- An Item Palette window opens.
- Select Signal Mast.

JMRI Panel

- Select the new mast from the Signal Mast Table.
- Then click and drag the mast to your panel.



- Remember to save your work.
- Click the mast to change.

JMRI Panel

- The new OpenLCB signal mast object in JMRI has the important feature of being an equal node with the actual mast on your layout. If JMRI logic changes the aspect it changes on both the panel and on the layout. That much is the same as other signals.
- Unlike some other signal types, external logic, be it another copy of JMRI, or internal logic in the signals themselves, may also be used to change both the layout and panel signals.

- Signaling is way too complex a subject to cover in much detail in the short time we have here today.
- Lets open the floor to questions and comments to find out more about your expectations of LCC.

Acknowledgements

Key OpenLCB Contributors: Bob Jacobsen, Alex Shepherd, David Harris, Stuart Baker, Balazs Racz, Jim Kueneman, Don Goodman-Wilson, John Plocher

Developer Group

10 to 15 actively working on code at any time

25 to 50 regular contributors and supporters

Many of the same people as are supporting JMRI

User Group

Started November 2009

July 2018 we had over 250 members

NMRA liaison: Stephen Priest

NMRA w.g. chairman: Karl Kobel

Info

Users Groups:

<https://groups.io/g/openlcb>

<https://groups.io/g/layoutcommandcontrol>

To Join: openlcb+subscribe@groups.io

layoutcommandcontrol+subscribe@groups.io

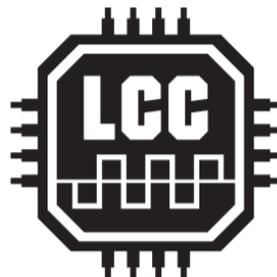
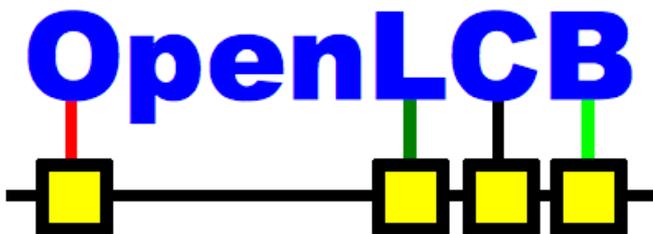
Useful Links:

<http://openlcb.org> or <http://openlcb.com>

<http://nmra.org>, choose S&RP scroll to 9.7

A not so Useful Link:

<http://www.layoutcommandcontrol.com/>



Questions

- ?