

RFID in Model Railroad Operations

Seth Neumann
Chris Drome



What are we trying to do?

There are many situations in operating a model railroad where it is useful to know the location of a particular piece of rolling stock or to know which piece of rolling stock is at a given location. In the context of operating my layout, the clerk is responsible for handwriting switch lists, which is time consuming and error prone.

We've created a system that automates the process of generating and printing lists of cars in standing order by identifying the cars in the train's consist.

How did we do this? RFID



What is RFID?

Radio-frequency identification (RFID) is the use of a wireless non-contact system that uses radio-frequency electromagnetic fields to transfer data from a tag attached to an object, for the purposes of automatic identification and tracking. Some tags require no battery and are powered by the electromagnetic fields used to read them. The tag contains electronically stored information which can be read from up to several meters (yards) away. Unlike a bar code, the tag does not need to be within line of sight of the reader and may be embedded in the tracked object.

* Wikipedia

RFID Applications

- Radio Frequency Identification
 - Inventory tracking
 - “Chip” your pets
 - Keyless entry for automobiles
 - Employee badges
 - FastTrak Lane on highways
 - Prototype Railroads use it to identify cars for classification





RFID system elements

- Systems
 - *Active reader passive tag*
 - Passive reader active tag
 - Active reader active tag
- Readers (active)
 - Emits RF pulses in reader's focus
 - Reads responses from tags
- Tags (passive)
 - Responds to RF pulses from reader
 - Rectifies pulse to provide power
 - Turns internal circuitry on
 - Transmits its ID



What can you do with RFID?

- Print “Track Lists” of cars in standing order
- Streamline “OS”-ing by reporting trains as they pass stations
- Feed car locations in staging to traffic generator programs such as JMRI Ops
- Keep track of member’s rolling stock in clubs
- Use your imagination ...

“Track List” Printer Application

I use switch lists to route cars on my railroad. Switch lists eliminate the need for crews to carry car cards with them, but the switch lists must be written out.

I don't like switch list programs because they usually don't do a good job in yards. I wanted a system to print the cars in standing order to minimize writing.



Switchlist

Station: _____

Train: _____

	Road	Number	Type	
1	MLLX	97295	LO	
2	DUPX	37234	LO	
3	CGLX	10262	LO	
4	SP	358735	GS	
5	UCLX	17009	T	
6	MOBX	14271	T	
7	MP	269468	XP	
8	WP	3773	XP	
9	SP	654809	XP	
10				
11				
12				
13				



System Requirements

- Must work on HO layout (track and rolling stock)
- Must work with DCC
- Must read tags reliably and consistently
- Must *not* read rolling stock on adjacent tracks
- Must *not* be visible from normal viewing angles
- Must minimize damage to layout and rolling stock
- Must work regardless of where metallic weights are located on rolling stock and locomotives



System Design - Hardware

- RFID reader
 - Commercial product vs. custom solution: build as little “magic stuff” as possible to minimize cost and risk
 - Arduino is inexpensive, well supported, open source
- Server: Linux (Ubuntu 12.04)
 - Inexpensive, well supported, open source
 - Runs well on (even older) commodity hardware
 - Open source databases, web servers, etc. available
 - Can dual-boot with Windows



System Design - Software

- MySQL database
 - Stores rolling stock data, switch lists
- Tomcat web server
 - Provides interactive, web-based UI
 - Manages rolling stock data
 - Formats and prints switch lists
- RFID hardware interface
 - Polls USB-based readers
 - Accepts connections from Ethernet-based readers
 - Processes incoming raw data



RFID Reader (USB)

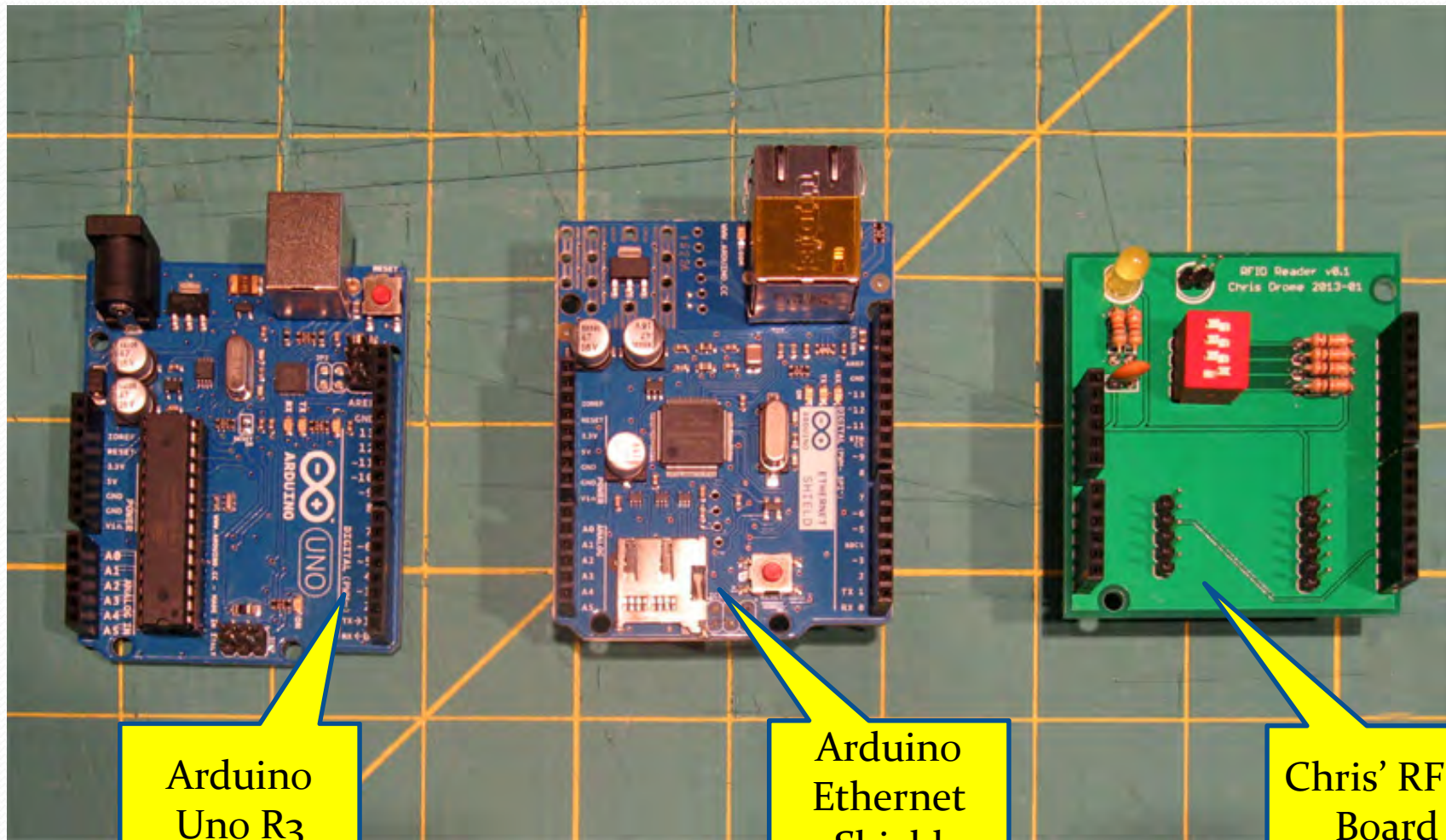
- USB-based RFID reader
 - Arduino board
 - Innovation ID-12 and ID-20
- Testing
 - Many different DCC locomotives (with/without sound)
 - Electrically isolated track above ID-12
- Limitations
 - Physical proximity of reader to server (5M = 16')
 - Reliable placement of tags on rolling stock (trucks)



RFID Reader (Ethernet)

- Ethernet-based RFID reader
 - Arduino board
 - Ethernet board
 - Innovation ID-12
 - Allows readers to be placed beyond the limits of USB (16') and as far as 100M.

Remote RFID Reader Components

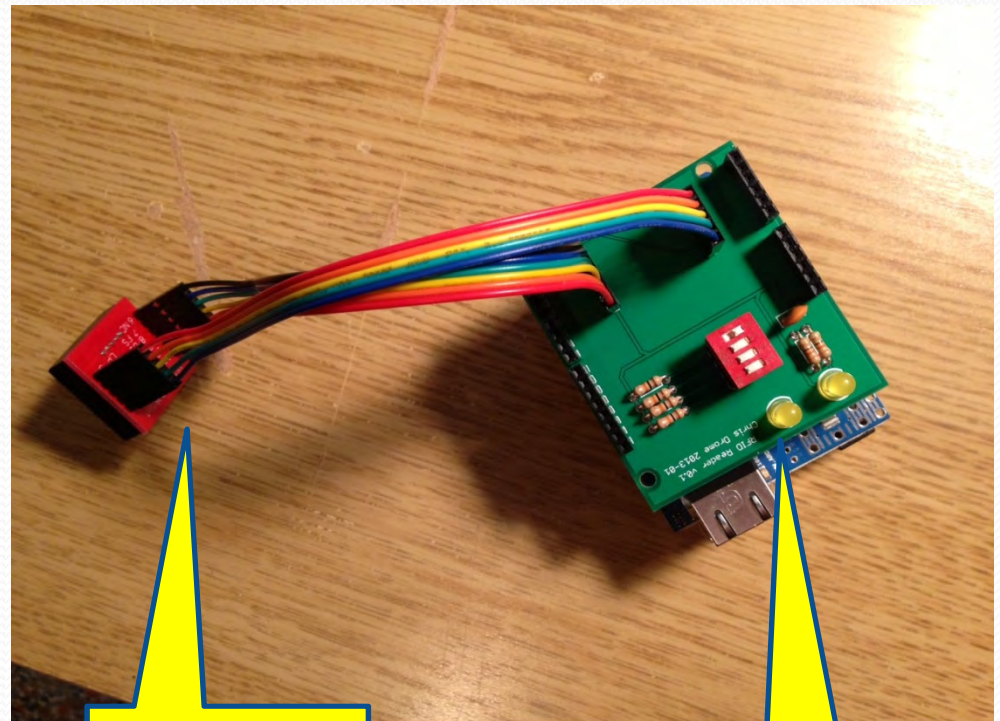
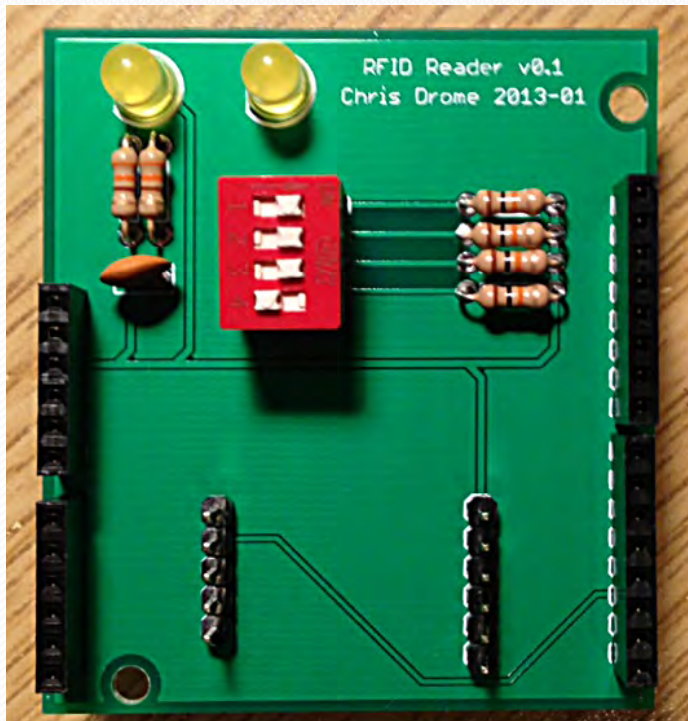


Arduino Uno R3

Arduino Ethernet Shield

Chris' RFID Board

PC Board Reader Shield



ID-12
Module +
Adapter

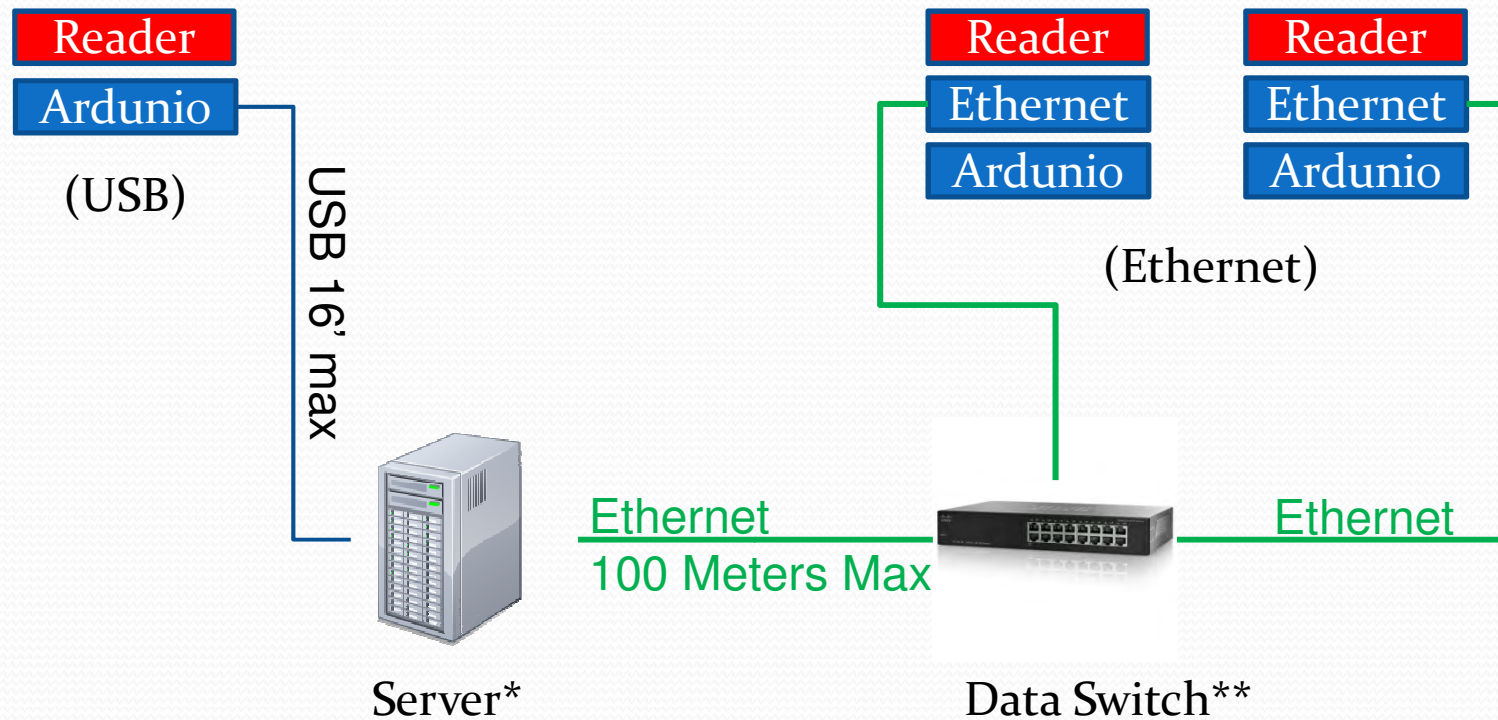
"Read"
LED, can
be remote



Custom System Software

- RFID reader (USB) polling software
 - Board writes to USB; software reads from USB
- RFID reader (Ethernet) daemon
 - Daemon accepts connections from Ethernet boards
- Train consist generator
 - Generates train lists from raw data
- Web service
 - Management of rolling stock roster
 - Format and view switch lists

System Overview



* The server can be any commodity hardware that runs Ubuntu Linux, we used an old Pentium 5, 1.8GHz.

** The switch pictured above is from Linksys (~ \$5/port), but you can just put it on your home network if you've got space. Static IP addresses were assigned to the boards such that they would not conflict with DHCP.

RFID Hardware - Tags

Tags come in various sizes and shapes and run at different frequencies. We use 125 KHz tags which conform to the EM4100 standard.



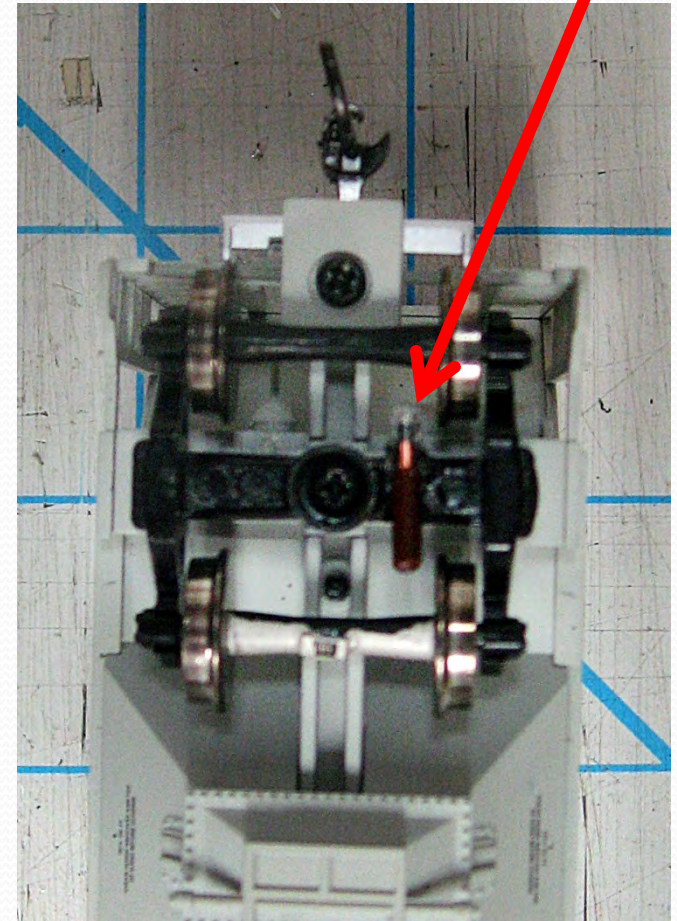


Tag Installation – Truck Mount

- Easily applied to (relatively inexpensive and easily replaceable) delrin truck
- Consistent 0.400” distance above reader:
 - Reads reliably
 - Does *not* read from adjacent track
- Hard to see from normal viewing angles
- Closer to under track reader than metal weights
- Far enough from car body that metal weights or brass body do not interfere with reading

Tag Installation

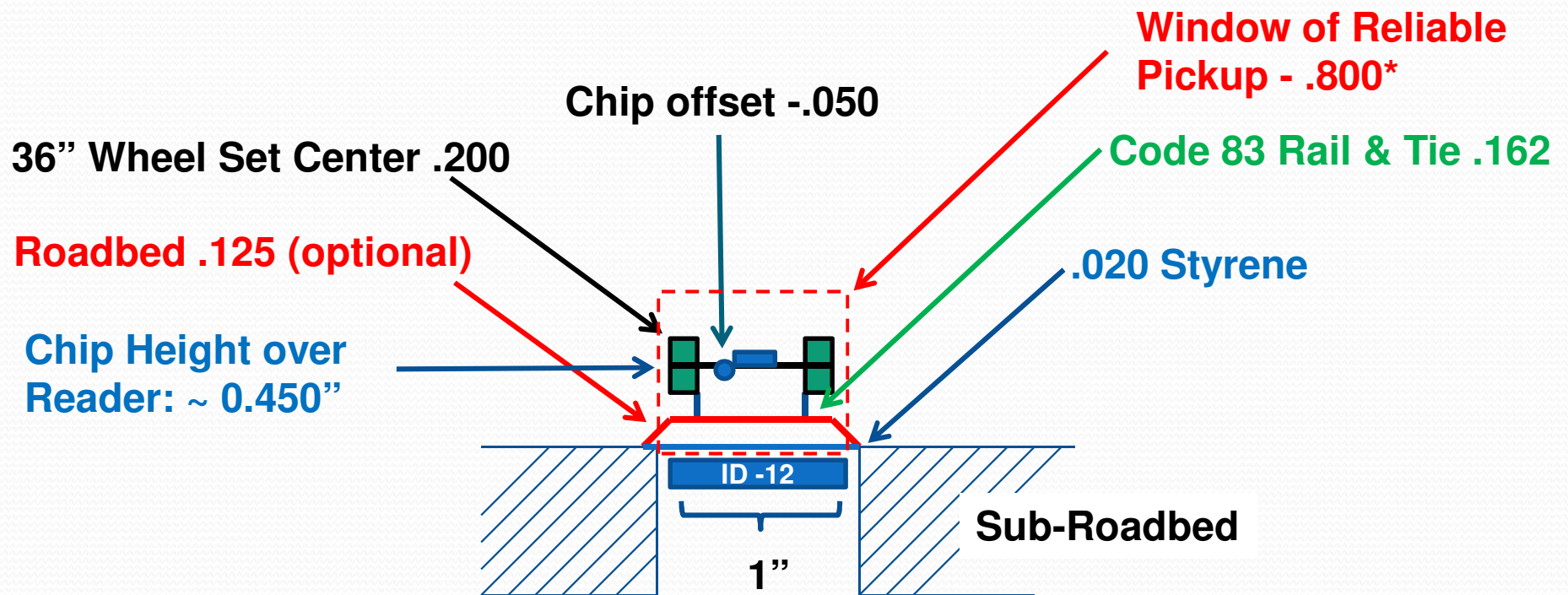
- Remove truck from car
- File a small notch next to pivot with a small round file
- Apply MicroPrep or similar product to inside of notch, allow to dry (5 minutes)
- Apply a drop of thickened CA to inside of notch
- Insert “Kitty Chip” in notch
- Allow to dry (5-10 minutes)
- Paint chip if desired
- Replace truck on car – I put them on the right side of the “B” end (aligned with brake wheel) for consistency



RFID Hardware - Readers



RFID Detection Geometry (HO)



* Chip aligned parallel to car, if chip is vertical, then 1.00 "

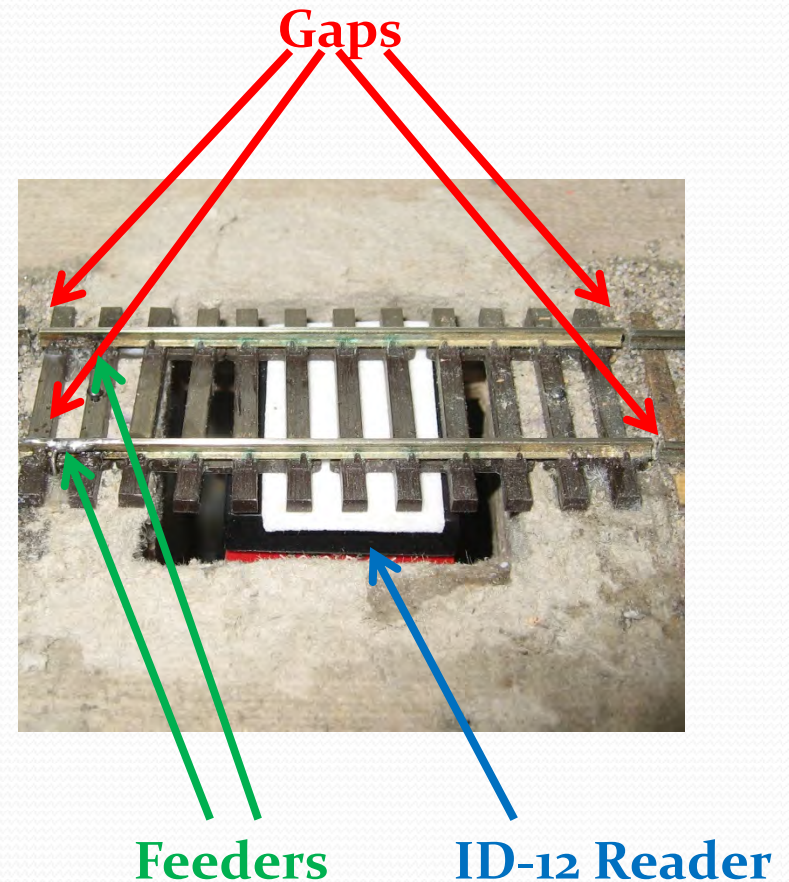
Reader Placement and Installation - 1

- Below track installation – like electromagnetic uncoupler
- Consistent distance: top aligned with roadbed:
 - Reads reliably
 - Does not read from adjacent track
- Invisible
- “Line of Sight” to tag, not through car weights

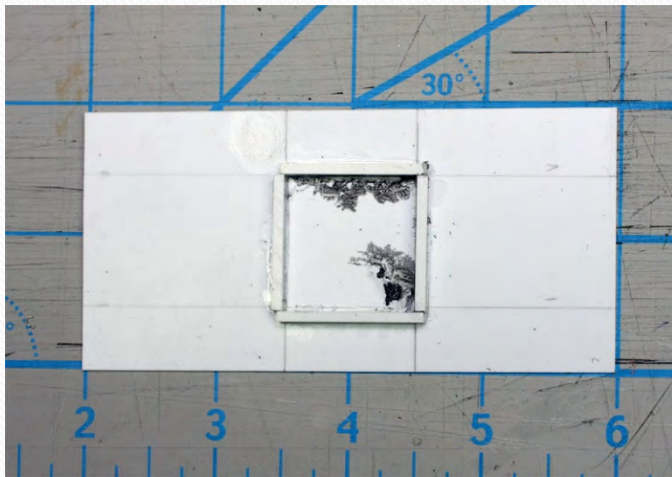


Reader Placement and Installation - 2

- Reader can be confused by large currents, so feed it separately (goes back to same power buss)
- Feed both rails from same end so currents in same direction
- In the final installation we covered the reader with a piece of .020 styrene



Reader Assembly and Installation - 3



Styrene cover – note ribs
To hold reader with “double
stick” tape

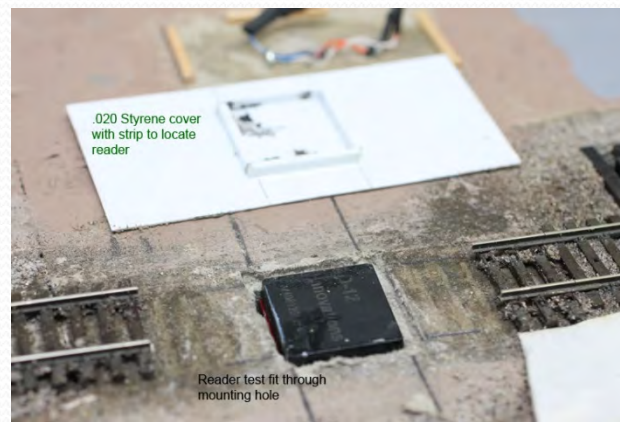


Track removed in reader location

Reader Assembly and Installation - 4



1" Hole for Reader

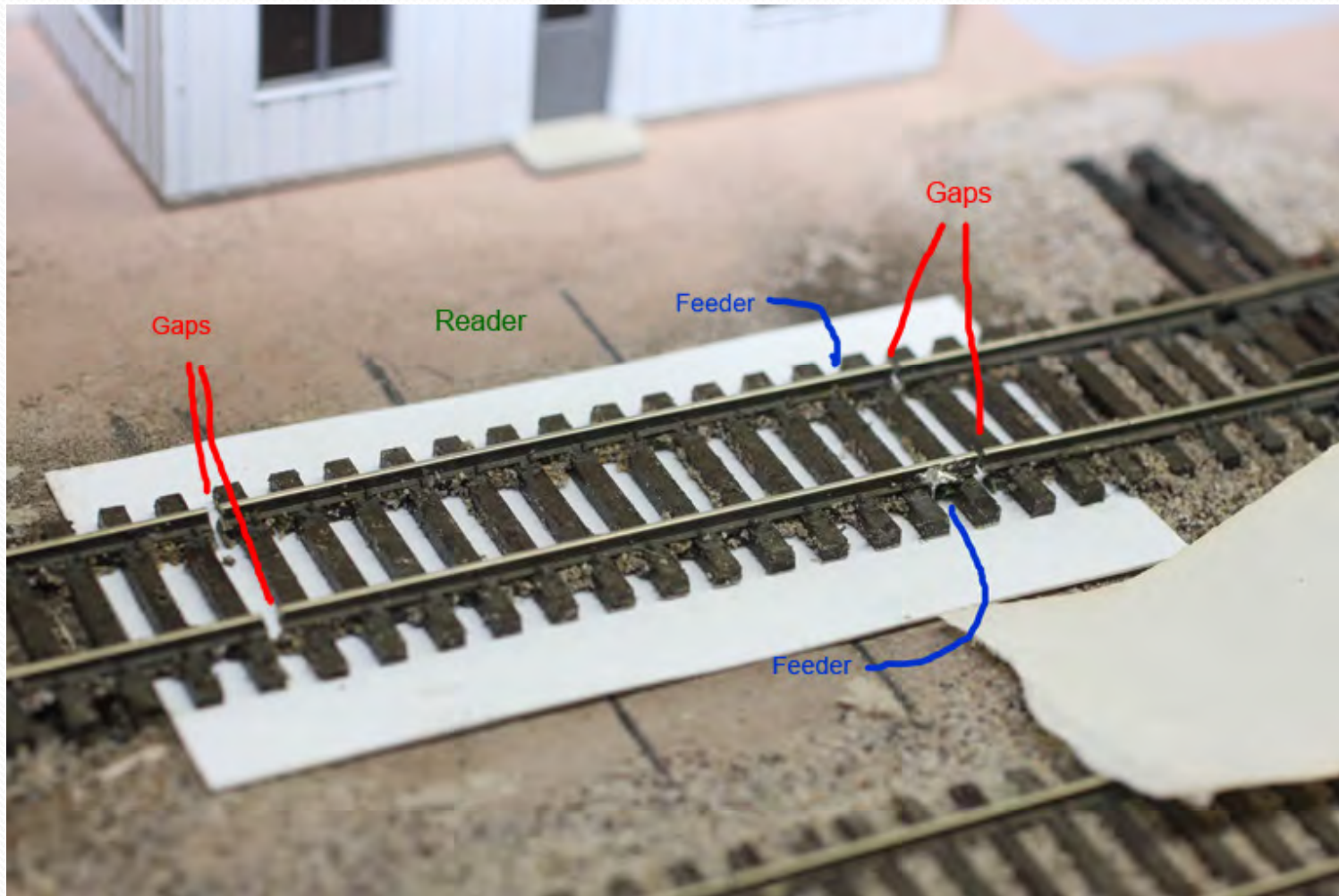


Reader Test Fit + Cover

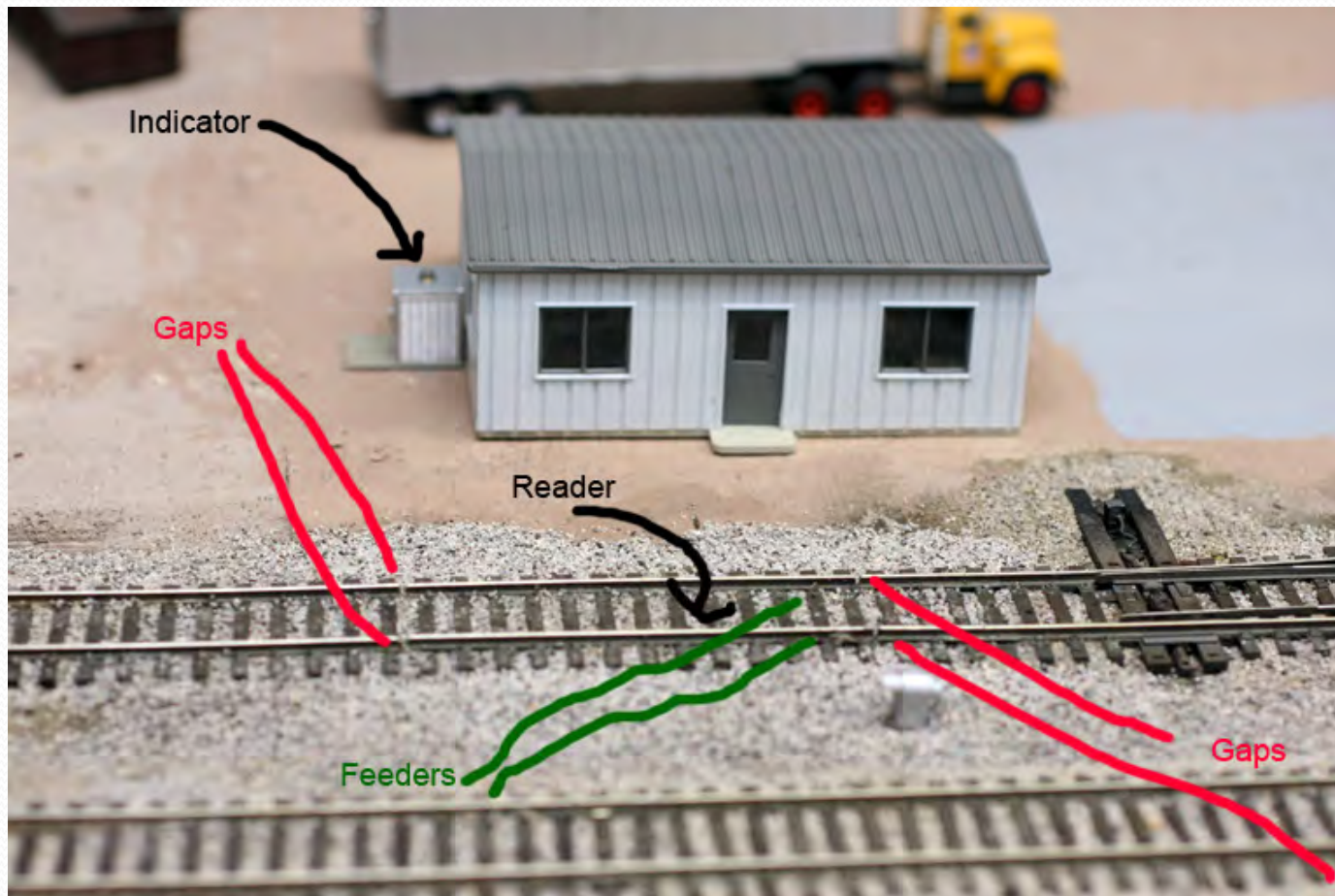


Cover in Place and Track down

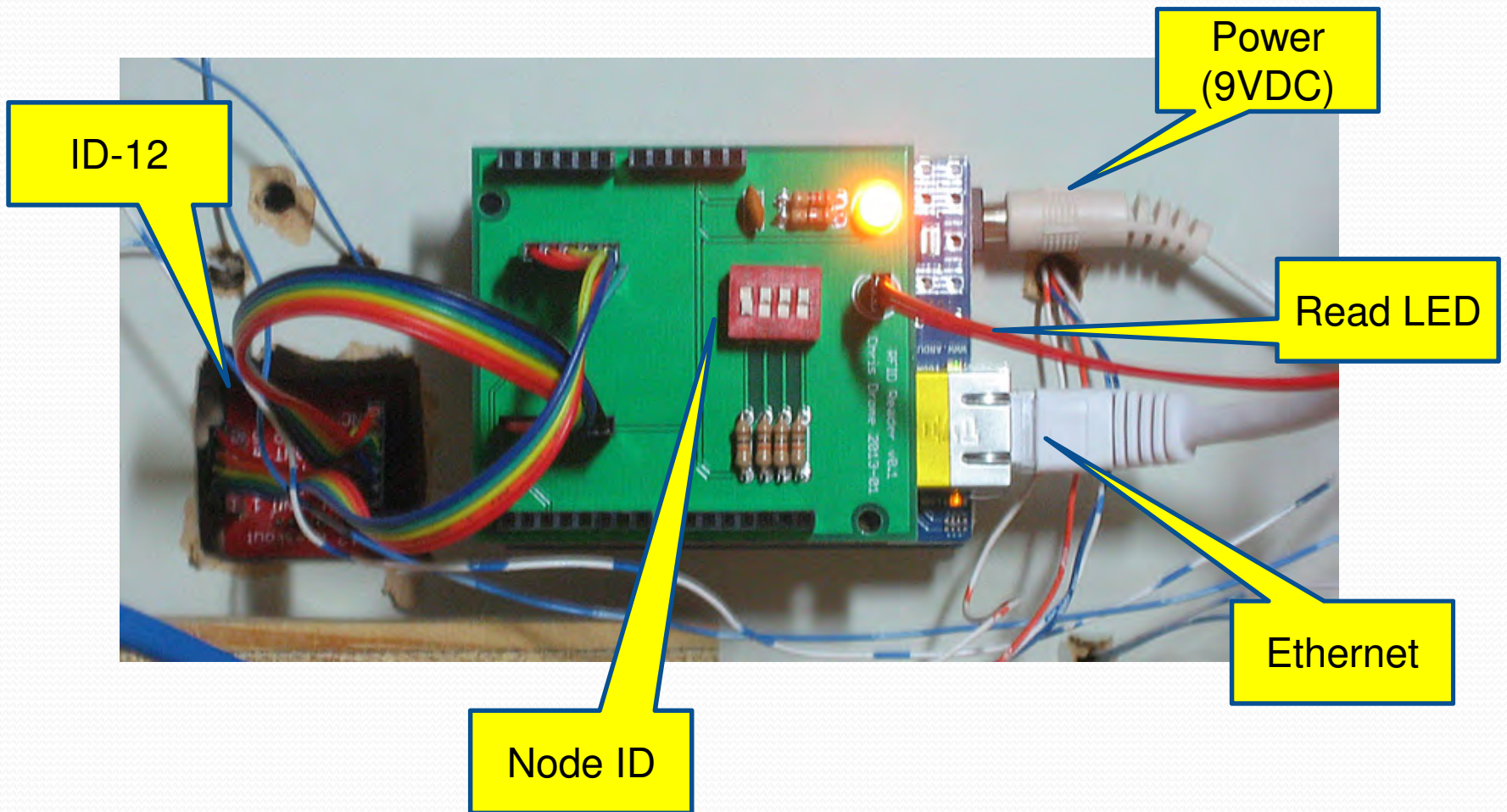
Reader Assembly and Installation - 5



Reader Assembly and Installation - 6



Reader Assembly and Installation - 7



Arduino Stack Mounted on Underside of Layout



Bill of Materials for DIY System

- Reader Node (~\$100 each)
 - Arduino Uno R3
 - Arduino Ethernet Shield
 - RFID Reader Shield
 - Reader Module
- Chip – 1 per car
 - 1.2mm x 10mm glass tag 125KHz, EM4100 (~\$2 ea in 100s)
- Server (any computer capable of Ubuntu)
- Data switches and cables as required (should be about \$10/node)

Additional Features

- Widget on server page to allow printing from any device (e.g. clerk's smart phone)
- Hand-held RFID reader “Wand” to read cars at industries (could also have permanent tags to code locations)
- Develop “OS” application
- Integrate with JMRI Reporter, JMRI Ops (already has stub)
- “Appliance” based server
- Low cost RIFD reader network

RFID Networking Approaches

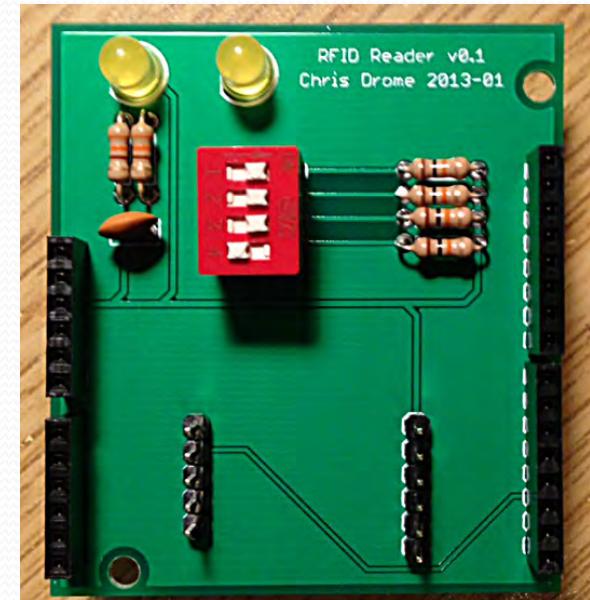
- We used wired Ethernet because it was relatively cheap and available - \$30 for Arduino shield, but price varies. If all the readers are within 15' of the computer, then direct USB connections will work (MERG approach).
- Arduino has WiFi but it's expensive - \$ 85+
- Arduino Xbee adapters (wireless) - \$35
- Arduino Bluetooth adapters - \$20+, limited range, drivers?
- NMRAnet – one vendor (RailStar) now has Arduino and NMRA-Net on board. It's pricey, ~\$100, but we expect the price to come down as adoption picks up
- We believe the answer may be to have the “slave” readers connected via dedicated reader+Xbee boards. It appears they could be implemented at a system cost (including interconnects) that's competitive with wired approaches, and would not require maintaining an extra bus under the layout.

JMRI “Reporter”

- JMRI has support for devices which “report” the identity of a train on the layout
- RFID Similar in effect to transponding (RailCom, Digitrax)
- Reporter reads the hardware output to JMRI and forwards the tag ID of the train, after a 2 second time out. Tag to Loco ID needs to be resolved
- JMRI can use this information to track trains on a PanelPro screen or other display
- Paul Bender implemented hardware for MERG and then CORE-ID, an ID-20 and X-bee radio network and Matthew Harris wrote the JMRI Reporter code for it. Our hardware is similar and we plan to implement.
- Rodney Black is looking into supporting reporters in CATS

RFID Parts and Services we are planning to offer

- Arduino RFID Shield PC board implementation of our prototype
- Pre-packaged reader nodes
- “Appliance” version of our server with predefined interface to JMRI (reporter and operations)
- 3D printed RFID “Antenna” detail
- Consulting and implementation Services





Other Model RR uses of RFID

- Train Brain has an RFID module, although their site didn't indicate what they use it for
- MERG has a kit with an interface board for their USB concentrator (originally used by Bender/Harris)
- Low level support for RFID in JMRI Ops
- Various demonstrations on YouTube



RFID Fact and Fiction - 1

- RFID won't work with DCC – **False**
 - All DCC issues can be managed with planning (see next)
- RFID won't work because of the electrical field generated by locomotives – **False**
 - Testing found no problem with most DCC locomotives
 - Any disruptions are limited to about one car length from the motor
 - Electrically insulating (attach feeders from the same side) the track around the reader fixes this problem
 - We have not seen any problem with sound enabled DCC locomotives
 - We have had success tagging the pilot or trailer of steam locomotives
 - We have had success tagging the fuel tank of diesels and the firebox of steam locomotives



RFID Fact and Fiction - 2

- RFID won't work at speed - **False**
 - RFID works reliably at any reasonable yard or switching speed.
- RFID won't work with rolling stock that has metal bodies or weights - **False**
 - Reads may fail if tags are mounted directly against a metal surface
 - Separating the tag from any metal surfaces fixes this problem
 - Mount tags on plastic trucks
 - Use a 0.100 piece of styrene as a spacer
 - Mounting tags on the center sill of brass cars did not prevent the tag from being read properly

Resources

- Sparkfun - <http://www.sparkfun.com/>
- Arduino - <http://arduino.cc/>
- RFID4MRR yahoo group - <mailto:RFID4MRR-subscribe@yahoogroups.com>
- MERG group - <http://www.merg.org.uk/index.php>
- JMRI - <http://jmri.sourceforge.net/>
- Paul Vandermyde - <http://www.tntrains.org/rfidinstall.html>
- Paul Bender - <http://nscaleintermodal.com/XBee-MRR/TransparentMode.html#RFID>