Arduino DCC Control

How do we use the Arduino Ecosystem for our DCC layouts
What Will be Covered

- Quick Arduino review
- DCC and DCC++ concepts
- Arduino Hardware for DCC++
- Arduino Software
- DCC++ demo
- Connecting Devices
- Speedometer Demo
- Automation Demo
Arduino Overview

- Arduino is an open source computer hardware and software company out of Italy
- Original product was developed for education by offering a low cost development product
- Rapidly grew into a DIY ecosystem with a large product range and following.
Arduino Overview

- “Arduino” commonly refers to the hardware and software platform
- The company also developed software to ease the development (programming) process
- There are many components or shields available to address many sensing and automation needs
DCC Concept

- Digital Command Control
- Originally Developed by Lenz in Germany
- NMRA Standard
- Standard consist of 2 parts
  - electrical
  - communication
- Many vendors have followed
- Most have stayed with the standards
DCC++ Concept

- Open Source software and hardware system
- Follows the NMRA standards and guidelines for DCC
- Hardware specific
- Runs on Arduino Uno or Mega
- Requires an Arduino Motor Shield
- Supports all gauges and handles about 3 amp
DCC++ Hardware

- DCC++ can use either an UNO or MEGA
- We will use a Mega for our implementation
DCC++ Hardware

- DCC++ requires a motor shield to drive the rail power
- There are two types of shields available
  - Arduino Motor Shield R3
  - Pololu Dual MC33926 Motor Shield
- We will use the Arduino Motor Shield for our implementation
Arduino Motor Shield

Pin Mappings for Arduino MEGA with Arduino Motor Shield

- Jumper Wire
- Pin Available for Custom Use
- Pin Reserved for DCC++ System
- Pin Reserved for DCC++ System, Unless Drive Traces Out on Back Board
- Pin Reserved if Using Arduino Ethernet Shield or WiFi Shield, Otherwise Pin is Available for Custom Use

Programming Track
Main Ops Track
DC Power Supply

03/2019 Thomas Ose - OMS(re)Models
Motor Shield Modification
DCC++ Software

- The software is Open Source
- The design concept consists of 2 modules
  - DCC++ Base Station
  - DCC++ Controller
- The DCC++ Base Station is a combination of the hardware and the software loaded on the Arduino
- The DCC++ design uses software rather than a throttle to control the Base Station
DCC++ Software

- The developers provide a controller package that is very fundamental but requires lots of modifications since it is intended for the developers layout
- JMRI is also supported and that is what we will be using in our demo
- The demo keeps it simple by using the USB port to access the Base Station. Other connection options are available.
DCC++ Demo
Connecting Devices

Pin Assignment

1  =  Ground  (black)
2  =  VCC (5 volt positive)  (red)
3  =  Signal  (white or Yellow)

Signal Type

Push Button  =  Digital (on or off)
Variable Resistor  =  Analogue
Sensor  =  Digital or Analogue depending on sensor type
TCRT5000 Photoelectric Switch

- Infrared Sensor
- Readily Available
- Allows for Analogue and Digital feedback
- Adjustable Reflective trigger depth
- Reliable but has issues with pure black
- Sensor bulky and hard to conceal
- Separating sensor and module helps
Introduction to Speed Calculation

- In our case it is really scale speed calculation
- For prototypical representation speed is critical
- Speed matching is important for consisting
- Speed calculation is based on
  - Start point
  - Distance
  - Time
  - End point
Speed Calculation Continued

- So we can base the speed calculation on the following

  - **Speed = (Distance * Time) / Scale**

- In order to calculate Speed to MPH we need other variables
Speed Calculation Continued

- \( L = \) length of travel in inches
- \( F = \) scale factor
- \( T = \) time in seconds for locomotive to travel length.

\[
\text{Scale MPH} = \left( \frac{L \times F}{12} \right) \div 5280 \times 3600 \div T
\]
Speedometer Demo

Diagram:
- Arduino Mega
- LCD Display
- DCC ++
- DCC Signal
- Power

Nodes A and B connected by DCC Signal
Comparative Photo Sensor

- Uses 2 photo sensors
- Compares T1 with T2
- if T1 is less than T2 than something is covering T1
- Very compact and reliable
- Originally designed by Geoff Bunza – self modified
- Assembly Required
Comparative Photo Sensor

- Designed to replace ties
- Can be painted as long as the lenses are clear
- Can be buried or hidden in other ways
- Solid state so long life span
Arduino Demo

DCC++ for Power Only

CmdrArduino

Slow Fast

Fast
Where to Buy

- Arduino  http://www.arduino.cc
- SparkFun  http://www.sparkfun.com
- Adafruit  http://www.adafruit.com
- Amazon  http://www.amazon.com
- ebay  http://www.ebay.com

or you can order them from me
Thank You for your time

If you have any further questions you can reach me at

OMS(re)Models
Thomas Ose
tmo@osemicro.com
(217) 483-4368

http://www.omsremodels.com