2015 FRC Control System

Beta Test by Team 1209

Overview

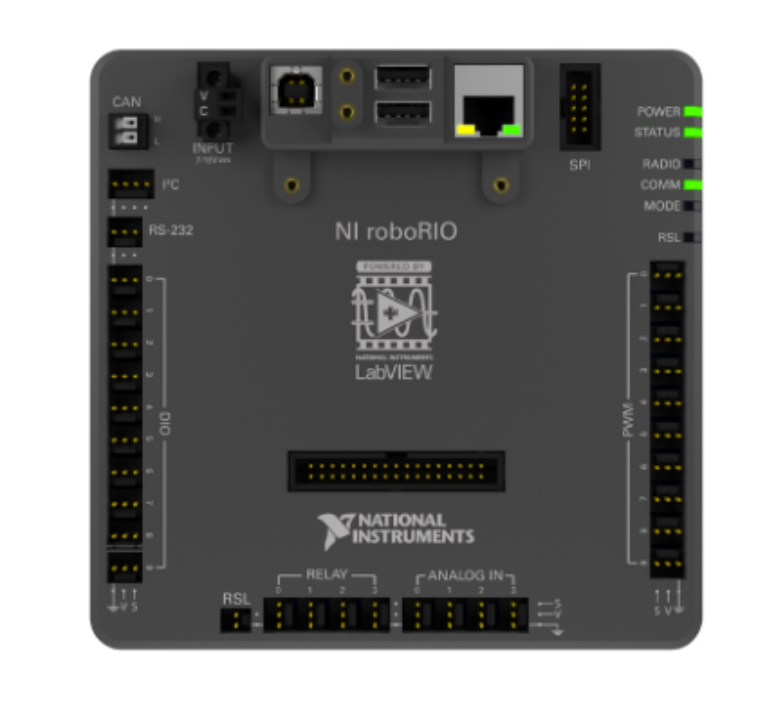
New Components

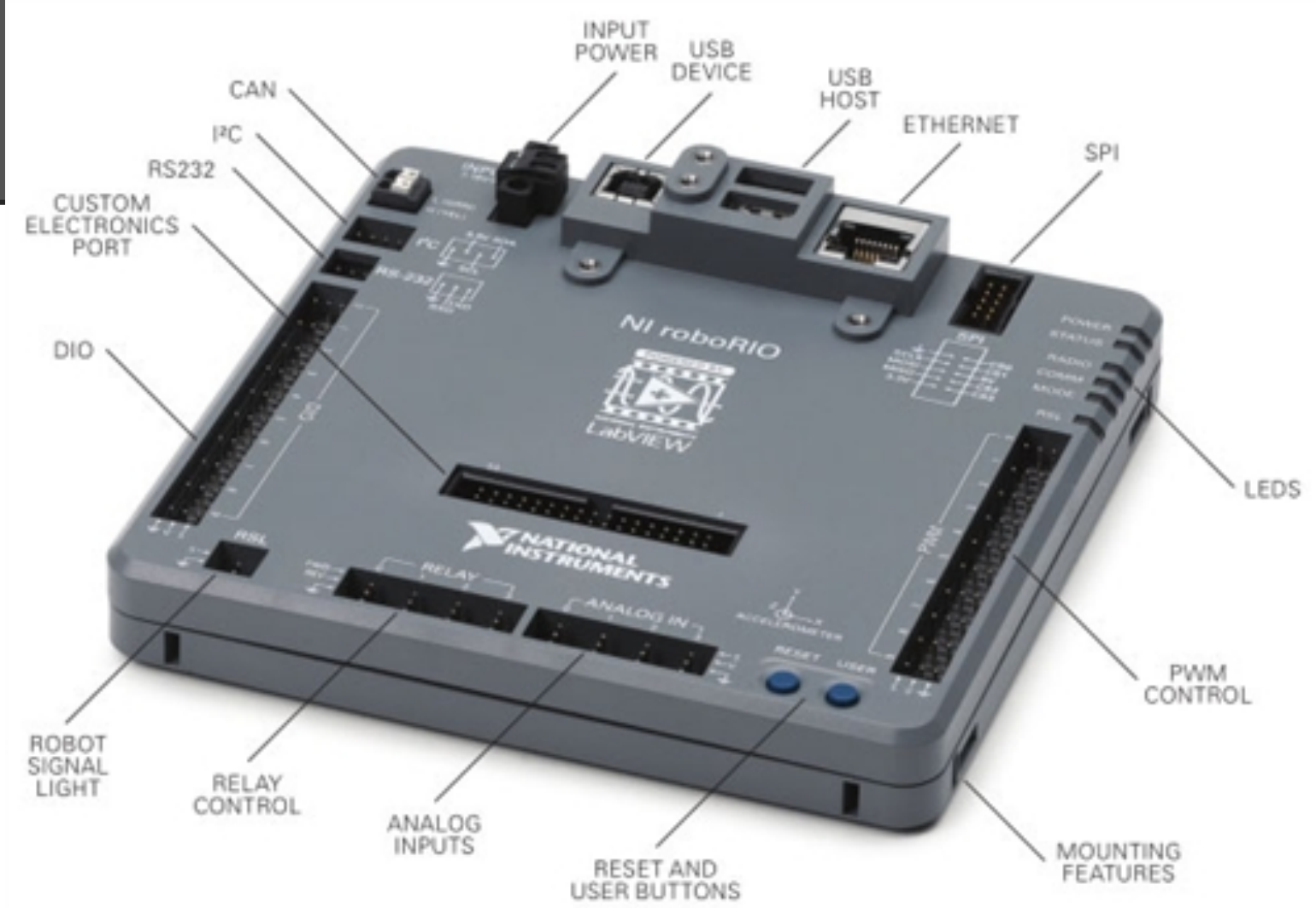
- RoboRIO
- Power Distribution Panel (PDP)
- Pneumatics Control Module (PCM)
- Voltage Regulator Module (VRM)
- Victor SP

RoboRIO

Capabilities

- Combines functions of cRIO and digital sidecar
- Built-in accelerometer
- CAN
- Two USB ports
- User-customizable I/O
- 5.75” x 5.625” x 1.375”




INPUT POWER
USB DEVICE
USB HOST
ETHERNET
SPI
CAN
I²C
RS232
CUSTOM ELECTRONICS PORT
DIO
LEDS
PWM CONTROL
ROBOT SIGNAL LIGHT
RELAY CONTROL
ANALOG INPUTS
RESET AND USER BUTTONS
MOUNTING FEATURES
NI roboRIO
LabVIEW
NATIONAL INSTRUMENTS
RELAY
ANALOG IN
RSL
PWM
RESET
USER

Power Distribution Panel

- Same snap-action breakers and WAGO connections as old PDB
- Similar size
- CAN connection with built-in terminator
  - Can give current of any power channel
  - Do not change default device ID

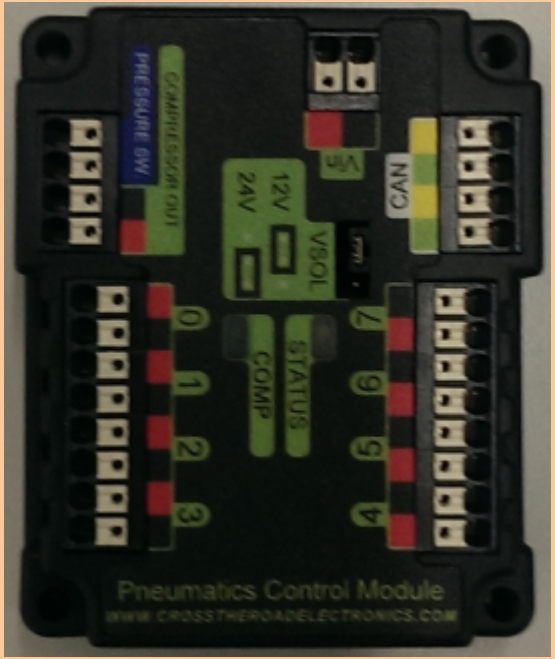

J1
J2
VEHICLE CONTROLLER PWR
VEHICLE PCM PWR
8
7
9
6
10
5
11
4
12
3
13
2
14
1
15
0
STAT
COMM
TERM/ ON OFF
RES.
CAN
WWW.CROSSTHEROADELECTRONICS.COM

PDP Connectors

- Old screw terminals are too large
- Must be able to crimp 6 AWG wire
  (We recommend using a swedge tool)
- Weidmuller connectors 16-20 AWG
  (Tinning recommended; Don’t twist wires)

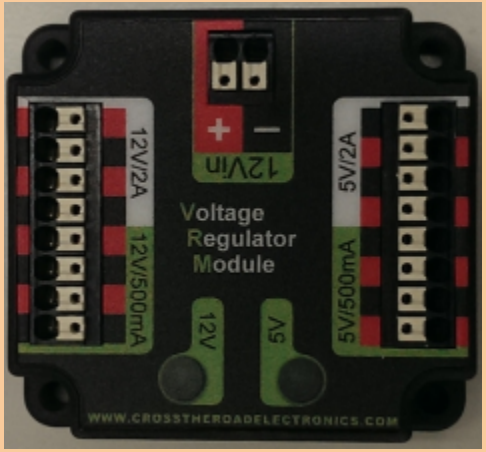
Pneumatics Control Module

- Compressor power
- Pressure switch
- Eight solenoids, 12 or 24 V
- CAN connection required
- Do not change the default device ID



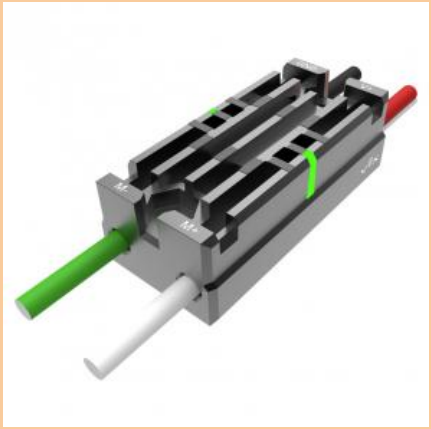
Voltage Regulator Module

- Eight power channels
  - 12V/2A x2
  - 12V/500mA x2
  - 5V/2A x2
  - 5V/500mA x2

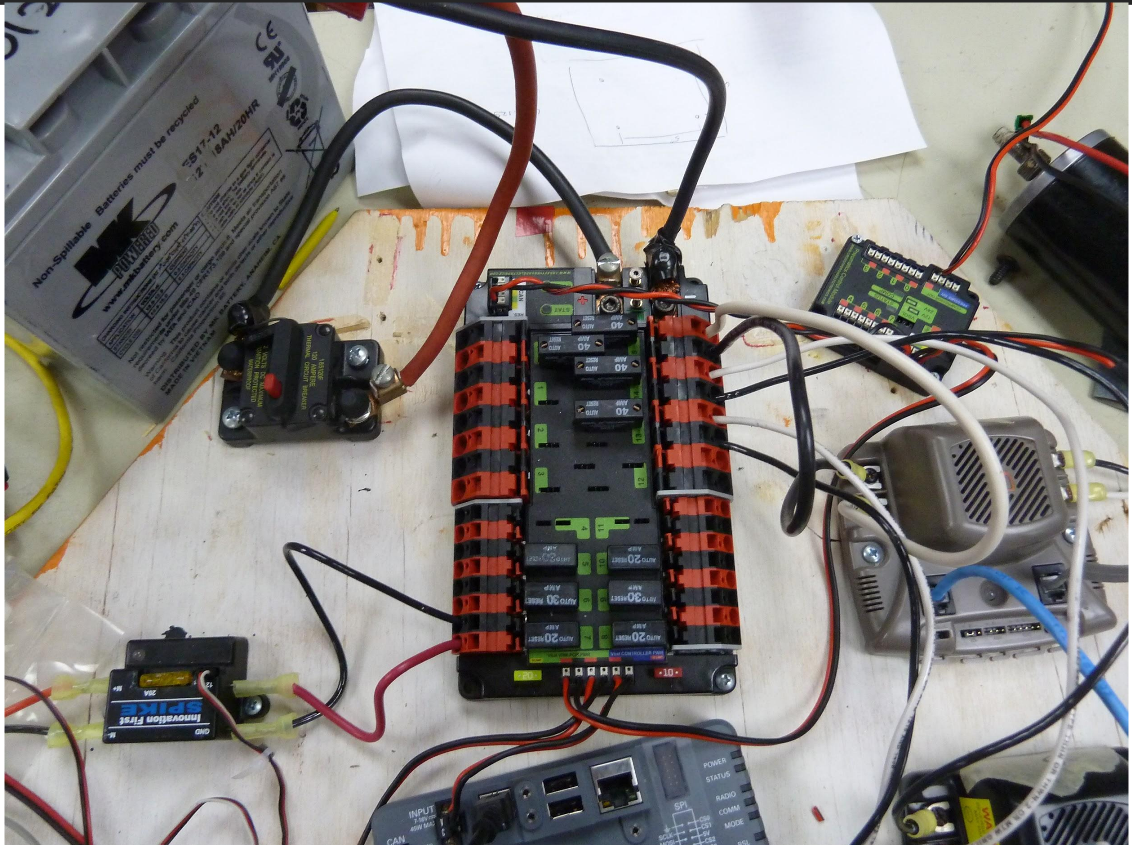


Victor SP

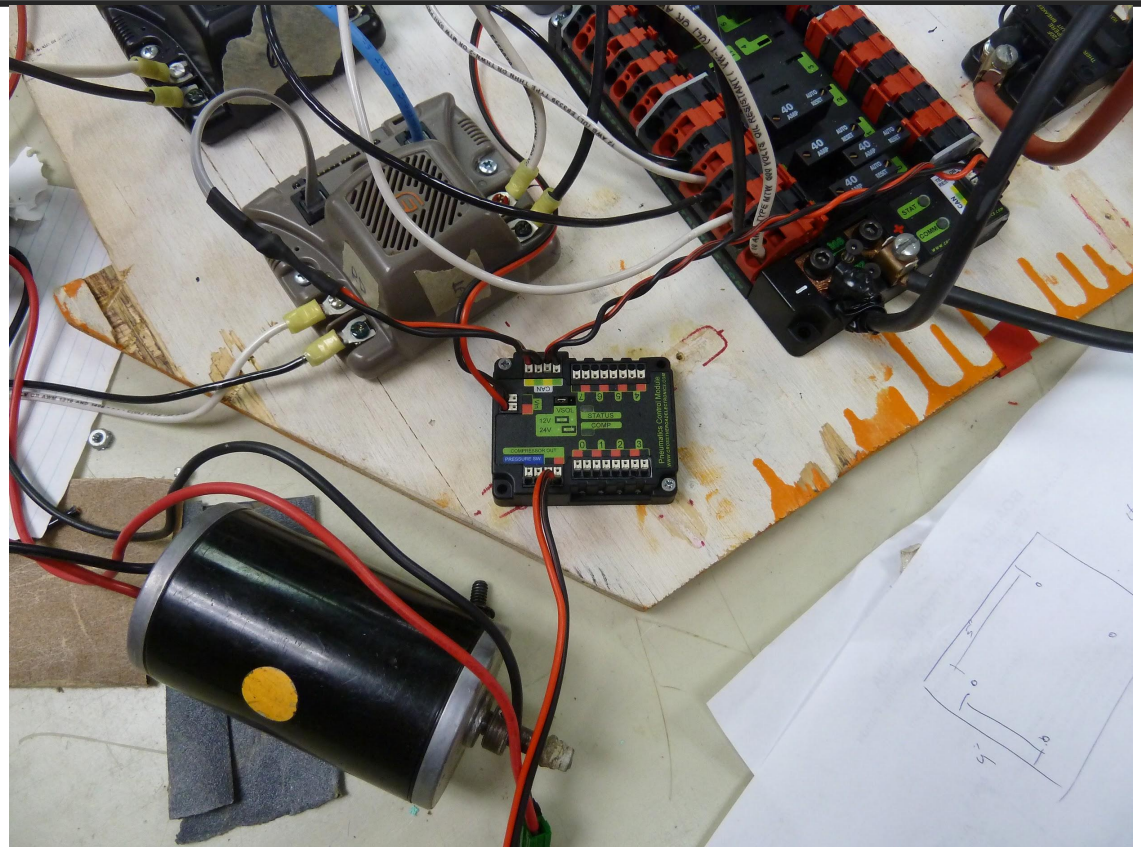
- PWM Motor Controller
- Exactly like a Victor 888 but smaller package
- No fan, so no worry of metal shavings



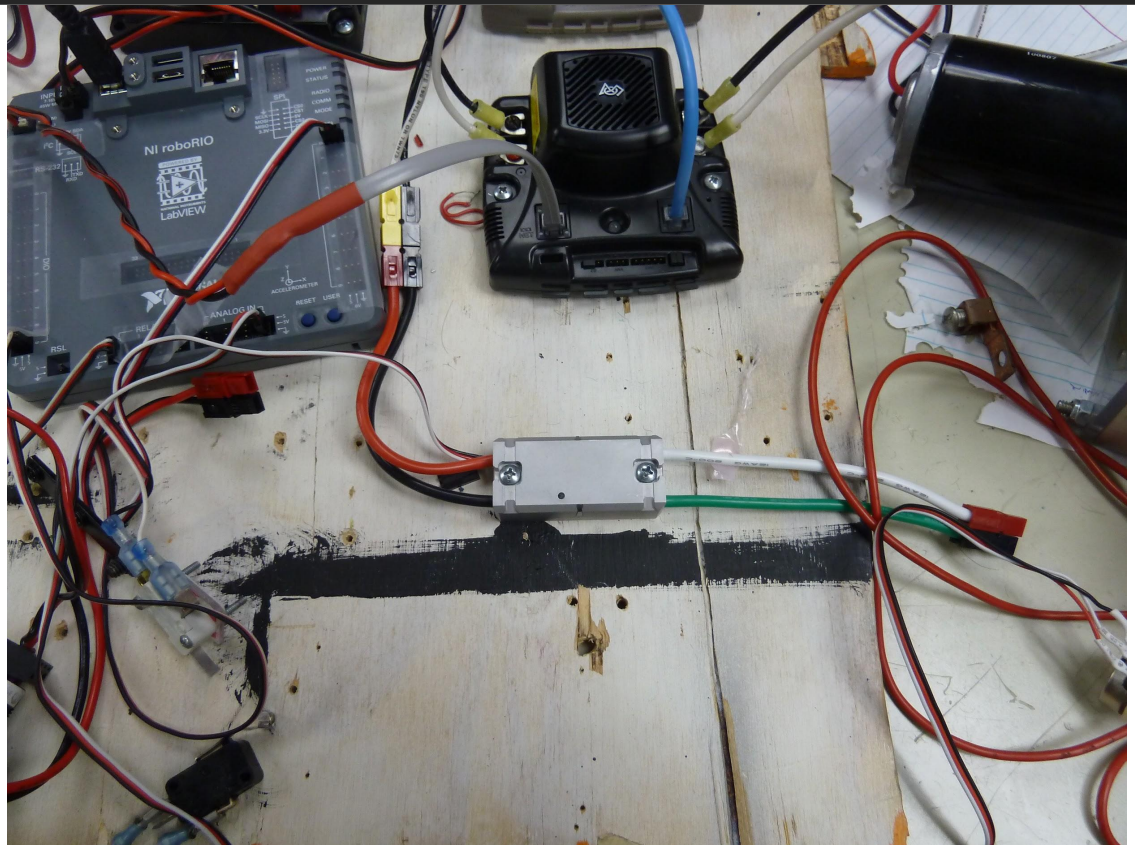
PDP


Non-Spillable Batteries must be recycled
POWER
STATUS
RADIO
COMM
MODE

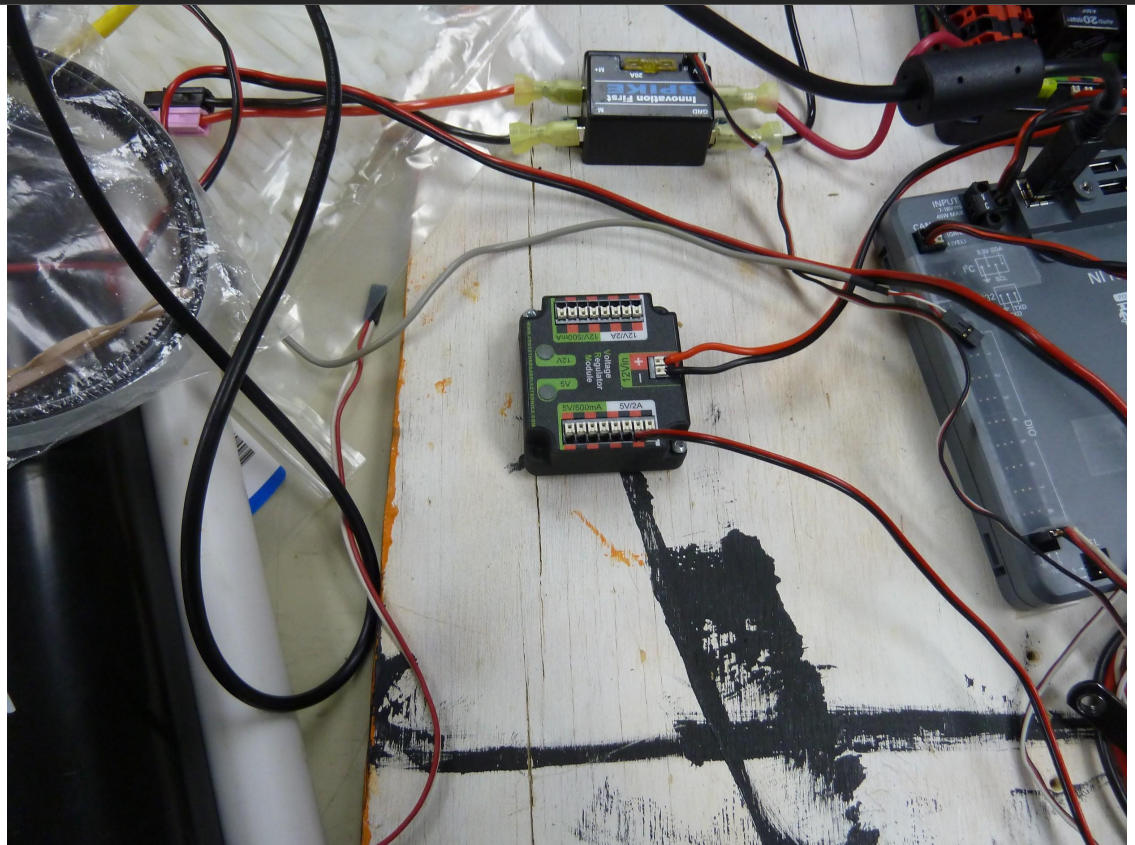
PCM



Victor SP



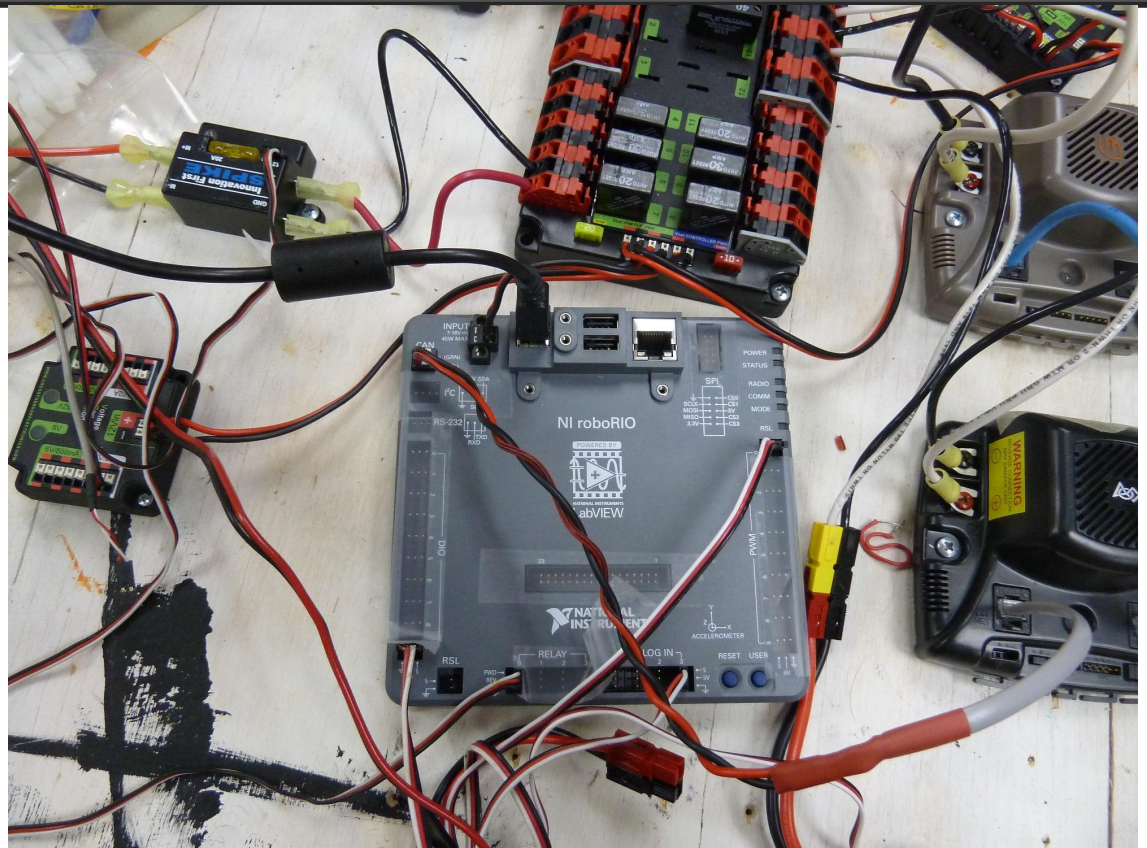
VRM



CAN with Jaguars

- RoboRIO needs to be at one end of CAN network
- PDP could be at other end (recommended)
- RoboRIO, PDP, and PCM connect to CAN through two wires
- Jaguars (only 2015) use four-pin modular plug
  - Yellow (H) goes to pin 2, Green (L) goes to pin 3

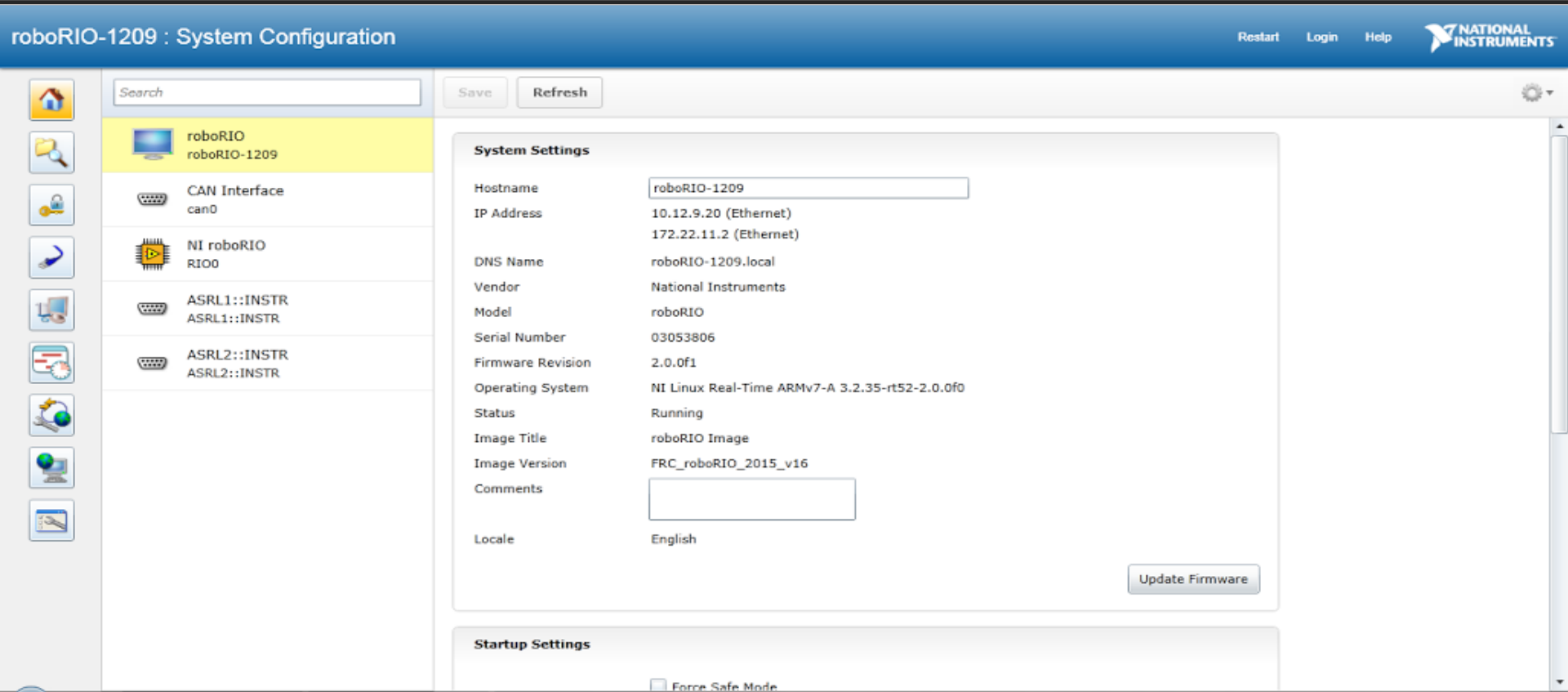
roboRIO


NI roboRIO
POWERED BY
LabVIEW
INPUT
CAN
I²C
SPI
RS-232
POWER
STATUS
RADIO
COMM
MODE
RSL
DIO
PWM
RELAY
LOG IN
ACCELEROMETER
RESET
USER
WARNING

Networking

- Connect to the roboRIO with ethernet or USB
- RoboRIO configuration uses web browser (USB only)
- Wireless uses same router as before

Networking


roboRIO-1209 : System Configuration
Restart
Login
Help
NATIONAL INSTRUMENTS
Search
Save
Refresh
roboRIO
roboRIO-1209
CAN Interface
can0
NI roboRIO
RIO0
ASRL1::INSTR
ASRL1::INSTR
ASRL2::INSTR
ASRL2::INSTR
System Settings
Hostname
roboRIO-1209
IP Address
10.12.9.20 (Ethernet)
172.22.11.2 (Ethernet)
DNS Name
roboRIO-1209.local
Vendor
National Instruments
Model
roboRIO
Serial Number
03053806
Firmware Revision
2.0.0f1
Operating System
NI Linux Real-Time ARMv7-A 3.2.35-rt52-2.0.0f0
Status
Running
Image Title
roboRIO Image
Image Version
FRC_roboRIO_2015_v16
Comments
Locale
English
Update Firmware
Startup Settings
Force Safe Mode

Eclipse IDE

- IDE for C++ and Java development
- Used to push code to roboRIO
- Updates to tools done within Eclipse
- Very similar to past IDEs as far as interface and menus

C/C++ - BenchtopTest/src/Robot.cpp - Eclipse

File Edit Source Refactor Navigate Search Project WPILib Run Window Help

Quick Access C/C++

Project Explorer

- BenchtopTest
  - Binaries
  - Includes
  - src
    - Robot.cpp
  - Debug
  - build.properties
  - build.xml

Robot.cpp | WPILib.h | AnalogInput.h | CANJaguar.h | Encoder.h

```
269             }
270
271             if (z < deadzone * 2 && z > -deadzone * 2) {
272                 z = 0;
273             } else if (z < 0) {
274                 z = 4 * pow(z + .5, 3);
275             } else {
276                 z = 4 * pow(z - .5, 3);
277             }
278
279             potPos = potentiometer.GetVoltage();
280             angle = gyro.GetAngle();
281             if (revControl) {
282                 myRobot.MecanumDrive_Cartesian(-x, -z, y);
283             } else {
284                 myRobot.MecanumDrive_Cartesian(x, -z, -y); // Mecanum drive based off of joystick
285             }
286             //Shoot Code -------------------------------------------------
287             if (shootStick.GetRawButton(2) && (shooterFlag == false)) {
288                 shooterTimer.Start();
289                 shooterMotorFlag = true;
290                 shooterFlag = true;
291             }
292             if (shooterTimer.Get() >= shootTime) {
293                 shooterMotorFlag = false;
294                 shooterTimer.Stop();
295                 shooterTimer.Reset();
296                 revShooterTimer.Start();
```

Ou... | Ma... | Ta...

- WPILib.h
- CANJaguar.h
- DigitalInput.h
- math.h
- stdio.h
- RobotDemo
  - myRobot : RobotDrive
  - shooterDrive : RobotDrive
  - blocker : Relay
  - tuskMotor : Victor
  - leftTusk : Relay
  - rightTusk : Relay
  - moveStick : Joystick
  - shootStick : Joystick
  - frontLeft : CANJaguar
  - backLeft : CANJaguar
  - frontRight : CANJaguar
  - backRight : CANJaguar
  - shooterMotor1 : Victor
  - shooterMotor2 : Victor
  - gyro : Gyro
  - angle : float
  - shooterFlag : bool

Problems | Tasks | Console | Properties

No consoles to display at this time.

Writable | Smart Insert | 286 : 51

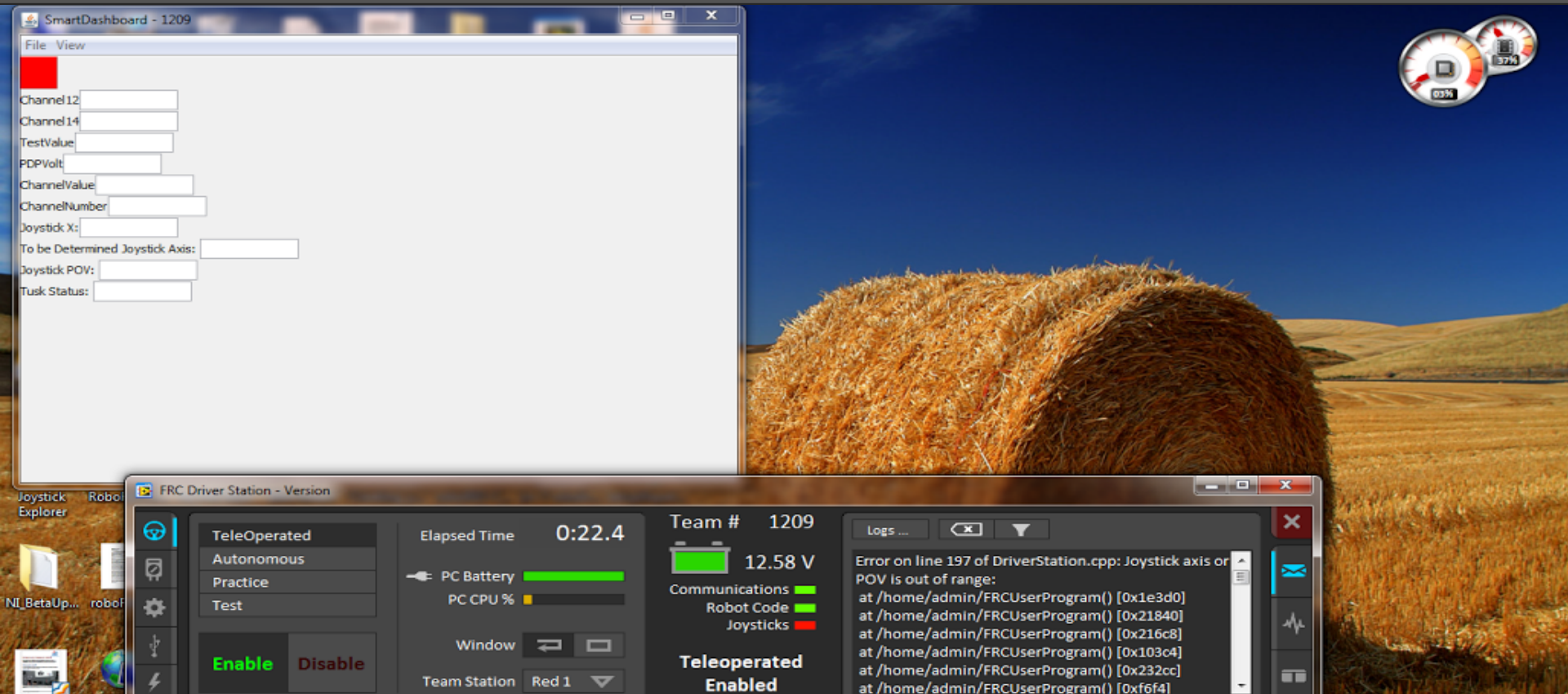
Changes to Code

- All channels count start now with 0
  - Changes to using joysticks and their axes
- Some classes and objects renamed
  - AnalogChannel to AnalogInput
- Driver Station LCD removed
  - Implementation of SmartDashboard as output driver station.
- Some functions renamed, can be checked in the WPILib Header file and child directories

Eclipse Issues

- Downloading, unzipping and installing the FRC Update Suite took a lengthy period time, close to two hours.
  - Be prepared to spend a day dedicated to setup alone.
- Took about 20 min. to install all necessary plugins for programming.

Drive Station Interface



Questions?