# cRIO-9033 SoftMotion CNC Demo

# Purpose:

This demo demonstrates the ability for cRIO to act as a controller for general motion control and/or CNC control applications. The FPGA code provided is a template for 5-axes open-loop motion control, complete with limit switch support. This code also leverages my [DXF Parser](https://github.com/johnwu130/LabVIEW-DXF-Parser) and [GCODE Parser](https://github.com/johnwu130/LabVIEW-GCode-Parser) to interpret these files into the corresponding SoftMotion commands. Sample DXF and GCODE files are also provided in the example.

# Disclaimer:

1. The DXF and GCODE Parser are only proof-of-concept examples. They have not been extensively tested with all kinds of files. This parsers are not meant to be complete SW products.
2. This example code is also not meant to be a complete CNC controller solution. There are many solid CNC controllers in the market today, such as Siemens and Fanuc. Please do not give the customer the wrong expectations.

# HW/SW Requirements:

* cRIO-9033
* NI 9401 x1 (For CW/CCW pulse output)
* NI 9403 x1 (For home switches and enable signals)
* Mini-displayport to VGA/DVI/HDMI converter and Monitor (res: 1920x1080)
* 3-axis gantry machine (such as CNC routers or 3D printers.)
* LabVIEW 2014, Real-Time, FPGA, and SoftMotion Modules

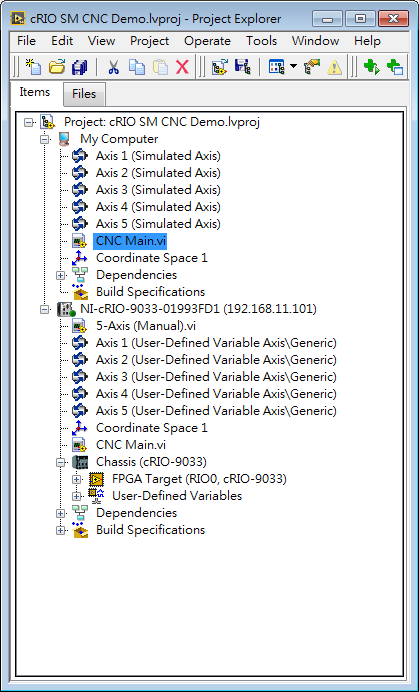
You can also run this code in pure PC simulation mode, if you do not have the HW. This demo supports up to 5 axes of motion, but only 3 are used for this demo. Code can be easily compiled for other targets.

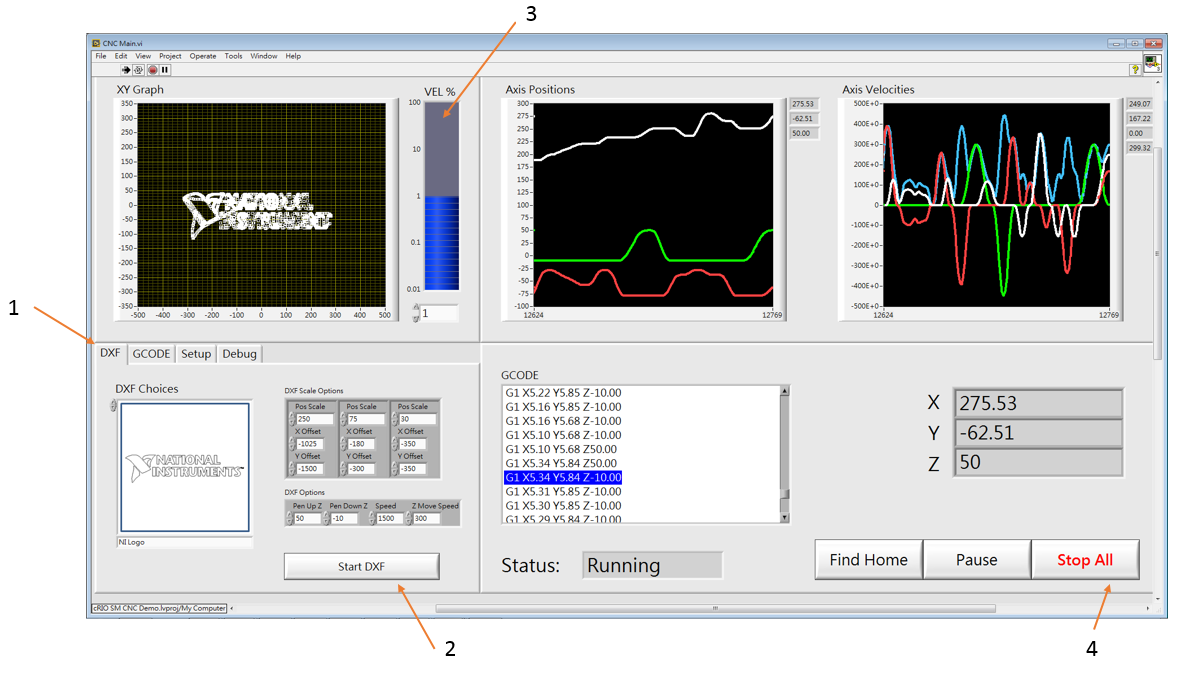
# Signal Connections:

|  |  |
| --- | --- |
| **cRIO-9033** | **Servo Motor** |
| Mod1/DIO0 | X Axis CW |
| Mod1/DIO1 | X Axis CCW |
| Mod1/DIO2 | Y Axis CW |
| Mod1/DIO3 | Y Axis CCW |
| Mod1/DIO4 | Z Axis CW |
| Mod1/DIO5 | Z Axis CCW |
| Mod1/DIO6 | Axis 4 CW (not used) |
| Mod1/DIO7 | Axis 4 CCW (not used) |
| Mod2/DIO0 | Axis 5 CW (not used) |
| Mod2/DIO1 | Axis 5 CCW (not used) |
| Mod3/DIO0 | X Rev Limit (not used) |
| Mod3/DIO1 | X Fwd Limit (not used) |
| Mod3/DIO2 | X Home Limit |
| Mod3/DIO3 | Y Rev Limit (not used) |
| Mod3/DIO4 | Y Fwd Limit (not used) |
| Mod3/DIO5 | Y Home Limit |
| Mod3/DIO6 | Z Rev Limit (not used) |
| Mod3/DIO7 | Z Fwd Limit (not used) |
| Mod3/DIO8 | Z Home Limit |
| Mod3/DIO28 | X Enable (optional) |
| Mod3/DIO29 | Y Enable (optional) |
| Mod3/DIO30 | Z Enable (optional) |
| Mod3/DIO31 | Axis 4 Enable (optional) |

# Running the example:

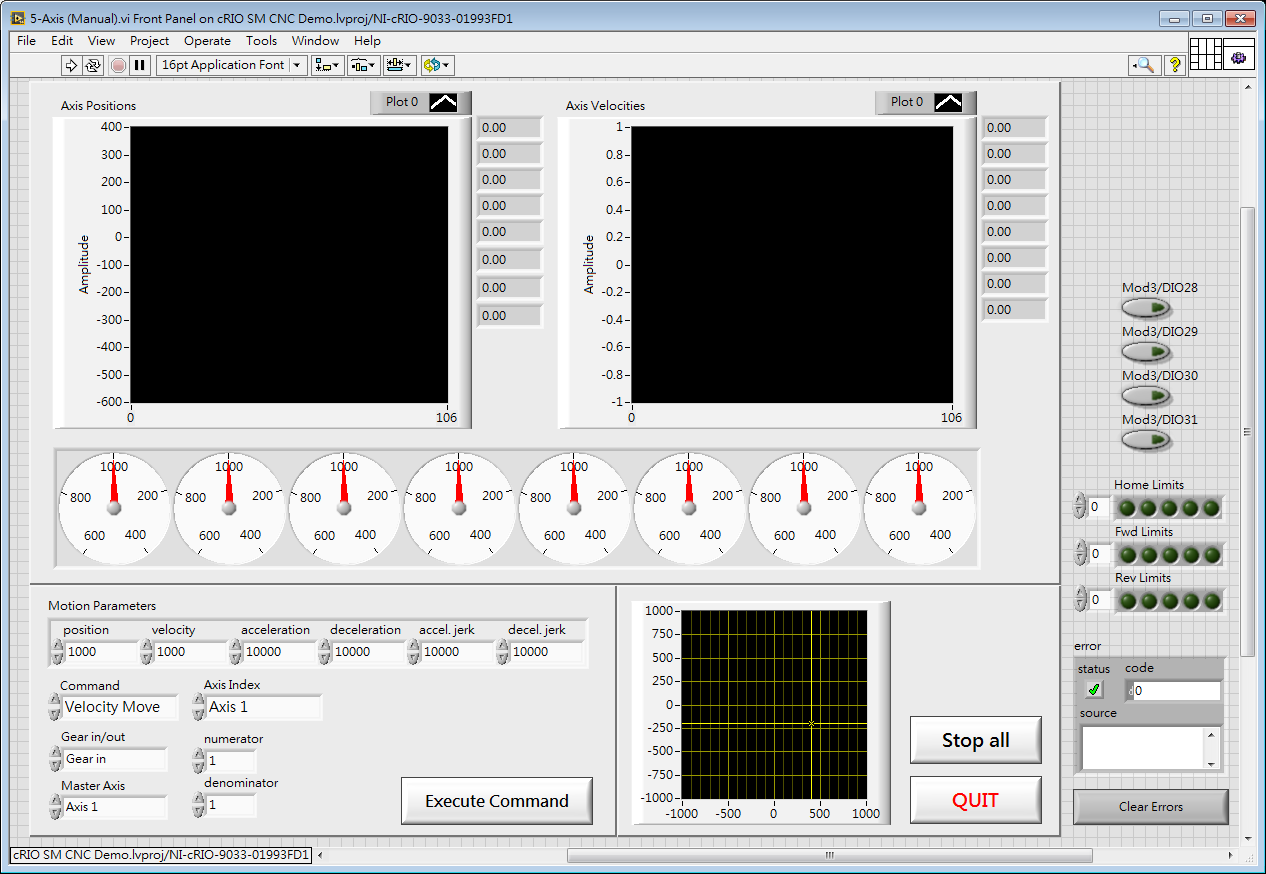
For PC simulation, open the “cRIO SM CNC Demo.lvproj” project file, and open CNC Main.vi under “My Computer”.





1. Choose from the DXF or GCODE tab, and choose from the Ring Control which file to execute.
2. Press “Start DXF” or “Start GCODE”
3. Adjust the speed if necessary.
4. Press “Stop All” to stop all motors. You can repeat 1-4 for another DXF/GCODE file.

For running on cRIO, open the “CNC Main.vi” under the cRIO Target. Operating instructions are the same as the PC simulation. Another file, “5-Axis (Manual).vi” is also provided for general testing and debugging.



# Notes:

1. My motors did not need “enable” signals, so I did not add this functions into CNC Main.vi. But the function has already been compiled in the FPGA bitfile if you need to access it.
2. For Linux-based CompactRIO targets, please copy the 2 folders “DXF” and “GCODE” into the \C\ directory before running CNC Main.vi. This location is where the program looks for the DXF and GCODE files.

John Wu

APAC Embedded Regional Marketing Engineer

[wei-han.wu@ni.com](mailto:wei-han.wu@ni.com)

12/14/2015