

6/10/2013 Ether

Context for this post is given in the following CD thread:

<http://www.chiefdelphi.com/forums/showthread.php?t=117267>

LabVIEW, C++, and Java FRC development installations all come with a Mecanum Cartesian module[1] that accepts a gyro input for field-centric control of fwd/rev and strafing.

The FWD/REV and Stafe L/R inputs to the MC module are supplied directly by the Y and X axes of the Joystick. Make sure the "Y" is positive for forward movement.

The "gyro" input to the MC module is the gyro angle reading. This makes the X and Y joystick commands field-centric for fwd/rev and strafing.

The Joystick Angle[2] in degrees is computed from the Joystick X and Y axes using the following formula:

$$\text{theta} = \text{atan2}(X, Y) * (180/\pi)$$

The above formula gives the clockwise angle from the "12 o'clock" position of the joystick. Again, make sure the "Y" value you are using is positive in the forward direction.

Make sure your gyro's angle reading is zero in the "straight downfield" direction, and increases in the clockwise direction (the joystick angle and the gyro angle must share the same zero and polarity).

For the driver interface you want, the "rotation" input to the MC module is not provided directly by the driver. Instead it is provided by the output of a closed-loop control[3] of your choosing operating on the joystick X and Y axes and the gyro angle reading.

In your closed-loop control[3], the process variable is the gyro angle and the setpoint should be computed based on the smallest magnitude error between the gyro and joystick angles. Formulas for this in C and LabVIEW are shown in this post:

<http://www.chiefdelphi.com/forums/showpost.php?p=1021821&postcount=8>