cRIO-9076_RT.vi

This VI shows an example of how to implement a closed loop position move using the NI 9505. The position loop is implemented using PID. The output of the loop is used as the command for the current loop, which then sets the PWM duty cycle for the motor. This example assumes that an encoder is connected for position and velocity feedback. This example does not implement advanced motion features such as trajectory generation and splining.

Controls | Plot_data

GWiz v3.2

cRIO-9076_RT.vi

Period(ms) 29.01 | Missed 0

Data plot

STOP

Ball & Beam

Manual

Motor Voltage 0.000

-6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6

Auto

Stiction V Offset cw 0.1500

Stiction V Offset ccw -0.1500

Motor Voltage

-6.0 -4.0 -2.0 0.0 2.0 4.0 6.0

-0.27

Ball Position (m) 0.251

0.0 0.1 0.2 0.3 0.4

0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1

0 15500

Ref

Gear

Ball

Gear Angle Theta (rad)

0.00 2.00 4.00 6.00 8.00 10.00 12.00

-0.215

Reference Generator

signal type: square

amplitude: 0.075

dc offset: 0.175

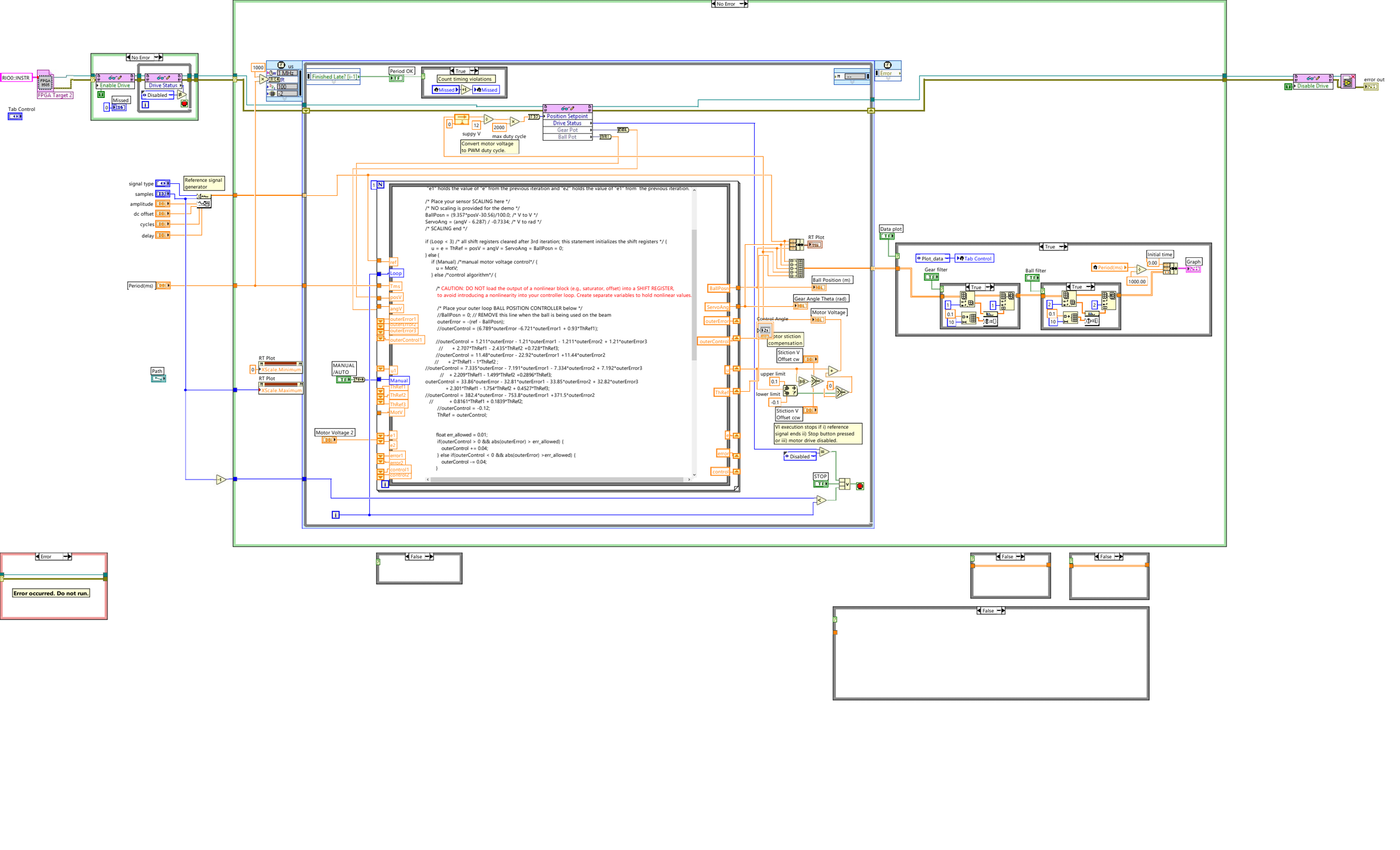
cycles: 10.00

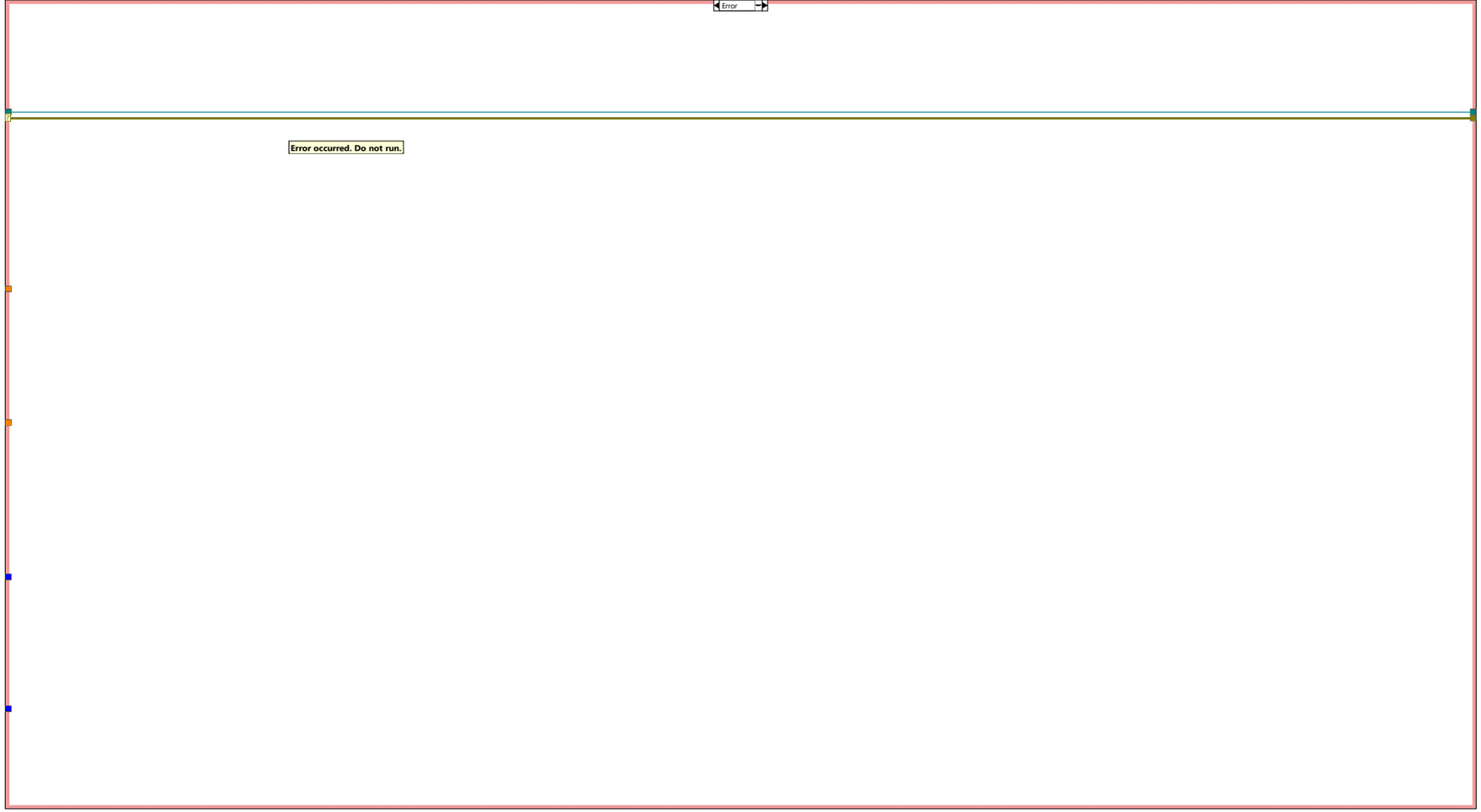
sample: 15500

delay: 0.00

Control Angle

-0.160044


RIO0::INSTR
FPGA 3606
FPGA Target 2
Tab Control
No Error
Enable Drive
Drive Status
Disabled
Missed
1000
us
MHz
100
C2
Finished Late? [i-1]
Period OK
True
Count timing violations
Missed
Error
Disable Drive
error out
0
12
supply V
2000
max duty cycle
Convert motor voltage to PWM duty cycle.
Position Setpoint
Drive Status
Gear Pot
Ball Pot
signal type
samples
amplitude
dc offset
cycles
delay
Reference signal generator
Period(ms)
Path
RT Plot
XScale.Minimum
XScale.Maximum
MANUAL /AUTO
Motor Voltage 2
"e1" holds the value of "e" from the previous iteration and "e2" holds the value of "e1" from the previous iteration.
/* Place your sensor SCALING here */
/* NO scaling is provided for the demo */
BallPosn = (9.357*posV-30.56)/100.0; /* V to V */
ServoAng = (angV - 6.287) / -0.7334; /* V to rad */
/* SCALING end */
if (Loop < 3) /* all shift registers cleared after 3rd iteration; this statement initializes the shift registers */ {
u = e = ThRef = posV = angV = ServoAng = BallPosn = 0;
} else {
if (Manual) /*manual motor voltage control*/ {
u = MotV;
} else /*control algorithm*/ {
/* CAUTION: DO NOT load the output of a nonlinear block (e.g., saturator, offset) into a SHIFT REGISTER,
to avoid introducing a nonlinearity into your controller loop. Create separate variables to hold nonlinear values.
/* Place your outer loop BALL POSITION CONTROLLER below */
//BallPosn = 0; // REMOVE this line when the ball is being used on the beam
outerError = -(ref - BallPosn);
//outerControl = (6.789*outerError -6.721*outerError1 + 0.93*ThRef1);
//outerControl = 1.211*outerError - 1.21*outerError1 - 1.211*outerError2 + 1.21*outerError3
// + 2.707*ThRef1 - 2.435*ThRef2 +0.728*ThRef3;
//outerControl = 11.48*outerError - 22.92*outerError1 +11.44*outerError2
// + 2*ThRef1 - 1*ThRef2 ;
//outerControl = 7.335*outerError - 7.191*outerError1 - 7.334*outerError2 + 7.192*outerError3
// + 2.209*ThRef1 - 1.499*ThRef2 +0.2896*ThRef3;
outerControl = 33.86*outerError - 32.81*outerError1 - 33.85*outerError2 + 32.82*outerError3
+ 2.301*ThRef1 - 1.754*ThRef2 + 0.4527*ThRef3;
//outerControl = 382.4*outerError - 753.8*outerError1 +371.5*outerError2
// + 0.8161*ThRef1 + 0.1839*ThRef2;
//outerControl = -0.12;
ThRef = outerControl;
float err_allowed = 0.01;
if(outerControl > 0 && abs(outerError) > err_allowed) {
outerControl += 0.04;
} else if(outerControl < 0 && abs(outerError) >err_allowed) {
outerControl -= 0.04;
}
ref
Loop
t ms
posV
angV
outerError1
outerError2
outerError3
outerControl1
u1
Manual
ThRef1
ThRef2
ThRef3
MotV
e1
e2
error1
error2
control1
control2
BallPosn
ServoAng
outerError
outerControl
u
ThRef
e
error
control
RT Plot
Ball Position (m)
Gear Angle Theta (rad)
Motor Voltage
Control Angle
Motor stiction compensation
Stiction V Offset cw
upper limit
0.1
lower limit
-0.1
Stiction V Offset ccw
VI execution stops if i) reference signal ends ii) Stop button pressed or iii) motor drive disabled.
Disabled
STOP
Data plot
Plot_data
Tab Control
Gear filter
Ball filter
Initial time
0.00
Period(ms)
1000.00
Graph
Error
Error occurred. Do not run.
False
False
False
False


Error
Error occurred. Do not run.