Lab 1

Clayton Haight – 20832547

Ethan Childerhose - 20820649

a) Familiarized.
b) Familiarized:

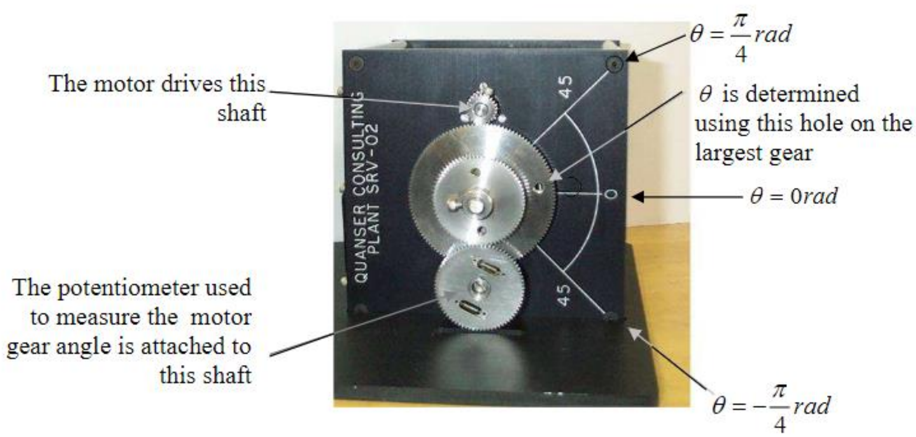


FIGURE 8: The motor system plant for Lab 1

c) Determine the gain K and the theta offset

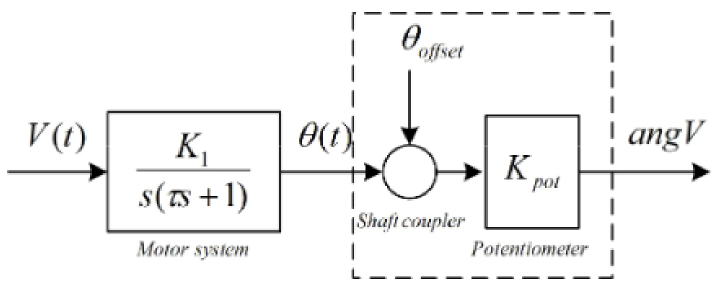


FIGURE 9: Nonlinearity introduced by servo-angle potentiometer

The offset was determined to be 4.212 with a gain error of 1.3576 creating this as the ServoAng
ServoAng = 1.3576*(angV – 4.212);

d) Determine stiction:

CW: 0.476V

CCW: -0.476V
We incrementally increased the voltage until the gears started moving. This station is symmetrical, the offset is 0.476V in either direction.

e)

The motor model is as follows:

$$\frac{K_1}{s(\tau s+1)}$$

We increased the C gain to 100 to get an overshoot while keeping motor voltage under 6V, which is its max. The response peaks at 0.137s after the step and overshoots by 7.5%.
The transfer function of the system would then be:

$$\frac{\frac{100K_1}{\tau s^2+s}}{1+\frac{100K_1}{\tau s^2+s}}=\frac{\frac{100K_1}{\tau s^2+s}}{\frac{100K_1+\tau s^2+s}{\tau s^2+s}}=\frac{100K_1}{\tau s^2+s}\frac{\tau s^2+s}{100K_1+\tau s^2+s}=\frac{100K_1}{\tau s^2+s+100K_1}=\frac{\frac{100K_1}{\tau}}{s^2+\frac{1}{\tau}s+\frac{100K_1}{\tau}}$$

$1.\ \omega_n=\sqrt{\frac{100K_1}{\tau}}$ $2.\ \omega_n=\frac{1}{2\tau\zeta}$ $3.\ T_p=\frac{\pi}{w_n\sqrt{1-\zeta^2}}$ $4.\ \%OS=e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}\cdot 100$

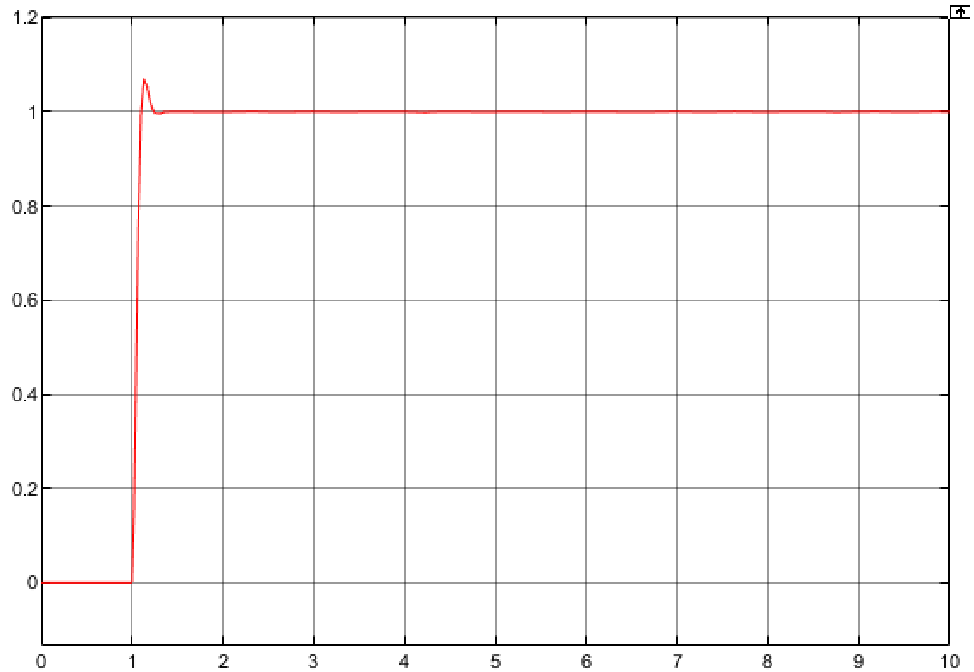
$\zeta=0.63615\quad \omega_n=29.72\quad \tau=0.0264\quad K_1=0.2335$

1. Plugging in 7.5% for %OS allows us to solve damping ratio as 0.6362
2. Plug damping into 3 with Tp which is 1.37s gives a natural frequency of 29.72
3. This then allows us to solve tau which is 0.0264 rad/(Vs)
4. Finally we solve for K1 which comes out to 0.2332

f) A simple if statement shall do the trick:

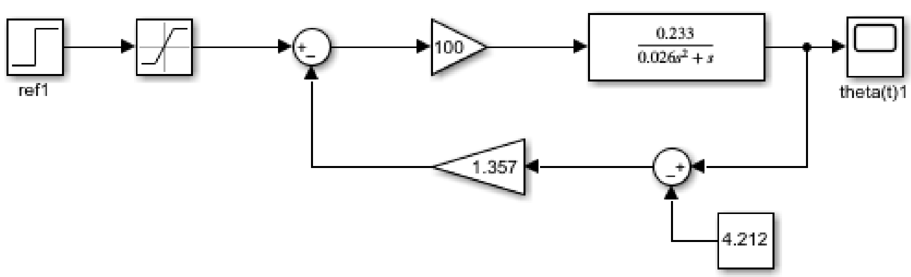
```
if (ref > 0.7)
{ref = 0.7;}
else if (ref < -0.7)
{ref = -0.7;}
```

Simulated Step Response



Time (s)

g) This is the equivalent block diagram of the system with the saturation and sensor calibration.



h) See the code below.

/* ======== USER INTERFACE TEMPLATE ============= */
/* Insert below the code for your scaling, saturation block, and controllers.*/

/* Variables may be declared on the box border, as shown for the input

```
"Tms" and the output "BallPosn". Variables can also be declared inline as was done for "Temp1". */

float Temp1;
float eGearAng;

/*  Shift registers permit previous values of variables to be saved.
The output variable "e" is wired to a shift register input on the For Loop border.
The inputs "e1" and "e2"are wired  to the corresponding shift register outputs.
"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 = posV ;  /* V to V */
ServoAng = 1.3576*(angV - 4.212); /* V to V */
/* 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

/* Place your gear angle SATURATOR below */
/*
if (ref > 0.7)
{ref = 0.7;}
else if (ref < -0.7)
{ref = -0.7;}
*/
/* Place your inner loop GEAR ANGLE CONTROLLER below */
 u = 100*(ref - ServoAng);

  }
}
```

/* ThRef, ThRef1, e, e1 are present, but not used in this demo.
However, they will be necessary (at a minimum) when the controllers will be implemented. */

Reyes Livera, Sergio. “ECE 484 Digital Control Systems Ball and Beam Lab Project Student Handout.”

Fisher, Michael W. “Course Notes”