Submission of Transform-Based Controller Parameters

On Wednesday, April 8, we will meet in the MTTC lab and evaluate the performance of your transform-based controllers. To assist in the timely evaluation I would like you all to email me a MATLAB "M-file" for each controller you design. Each M-file will contain the following:

- Your predicted RMS tracking error (in counts)
- The coefficients of your controller $D(z)$: (a_i, b_i, K)

The name of the M-file shall be `[your lower-case initials][controller type].m`, for example, the M-files of mine for the `PD` controller and the `lead` controller would be: `gpsPD.m` and `gpslead.m` Execution of each M-file will display your initials, your predicted RMS tracking error and set the controller coefficients (*e.g.* substitute your initials where I have shown `gps` or `GPS`).

NOTE: Each controller will be represented in the Simulink Real-Time Workshop block diagram at runtime as a "`Discrete Transfer Fcn`" block. The format of the M-files below reflect that. In designing your controllers, you will be placing z-plane poles and zeros. But after finalizing each design, multiply out the numerator and denominator so that you have a discrete transfer function with numerator and denominator polynomials and a separate gain K.

1. Lead compensator. The lead compensator contains a single pole and zero, thus:

$$D(z) = \frac{U(z)}{E(z)} = K\frac{z+b}{z+a}$$

and the M-file for this controller shall be:

```
% Lead compensator design
disp('GPS Lead compensator RMS error = XX.XX counts');
% Lead compensator coefficients
b = XX.XX;
a = XX.XX;
K = XX.XX;
```

Example: If your initials were "sta", and your lead compensator was

$$D(z) = 0.297\frac{z-0.7}{z+0.2}$$

and your RMS error were 37.3 counts, the corresponding M-file would be named **stalead.m** and would contain:

```
% Lead compensator design
disp('STA Lead compensator RMS error = 37.3 counts');
% Lead compensator coefficients
b = -0.7;
a = 0.2;
K = 0.297;
```

The other three controllers follow this same format. All the a and b coefficients in my Simulink models are *positive*, so if yours is negative you must have a negative number.

2. Proportional + Derivative Controller (PD). The PD Controller contains a pole at the origin and a single zero, thus:

$$D(z) = \frac{U(z)}{E(z)} = K\frac{z+b}{z}$$

and the M-file for this controller shall be:

```
% PD controller design
disp('GPS PD Controller RMS error = XX.XX counts');
% PD controller coefficients
b = XX.XX;
K = XX.XX;
```

3. Proportional + Integral + Derivative Controller (PID). The Proportional + Integral + Derivative Controller (PID) contains a pole at +1 and a pole at the origin, plus two zeros; thus in "`Discrete Transfer Fcn`" it will be:

$$D(z) = \frac{U(z)}{E(z)} = K\frac{z^2 + b_1 z + b_2}{z(z-1)} = K\frac{z^2 + b_1 z + b_2}{z^2 - z}$$

and the M-file for this controller shall be:

```
% PID controller design
disp('GPS PID Controller RMS error = XX.XX counts');
% PID controller coefficients
b1 = XX.XX;
b2 = XX.XX;
K = XX.XX;
```

4. Proportional + Integral + Derivative "Lead" Controller (PIDLead). The Proportional + Integral + Derivative "Lead" Controller (PIDLead) contains a pole at +1 and another single pole, plus two zeros; thus:

$$D(z) = \frac{U(z)}{E(z)} = K\frac{z^2 + b_1 z + b_2}{(z-1)(z+a)} = K\frac{z^2 + b_1 z + b_2}{z^2 + (a-1)z - a}$$

and the M-file for this controller shall be:

```
% PIDLead controller design
disp('GPS PIDLead Controller RMS error = XX.XX counts');
% PIDLead controller coefficients
b1 = XX.XX;
b2 = XX.XX;
a = XX.XX;
K = XX.XX;
```