5.b.)

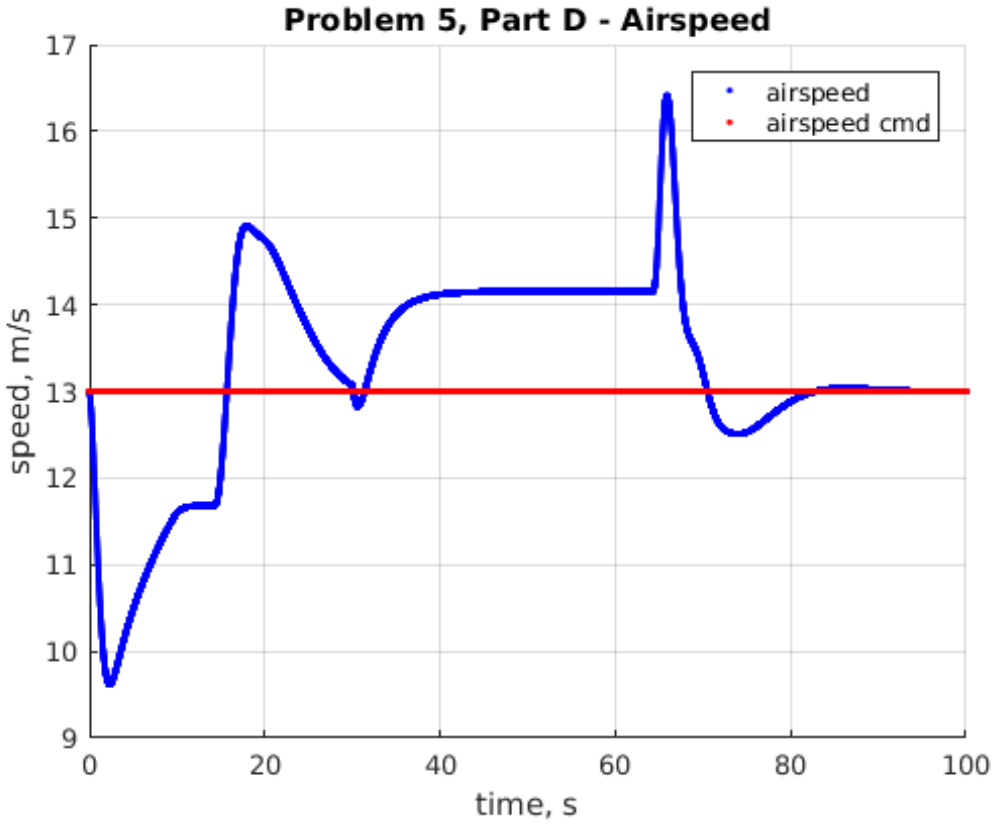
```
W_V2 = 40;
omega_n_V2 = omega_n_theta/W_V2;
zeta_V2 = 1;
P.airspeed_pitch_kp = (models.a_V1 -
2*zeta_V2*omega_n_V2)/P.K_theta_DC/P.gravity;
P.airspeed_pitch_ki = -omega_n_V2*omega_n_V2/P.K_theta_DC/P.gravity;
P.airspeed_pitch_kd = 0.0;
```

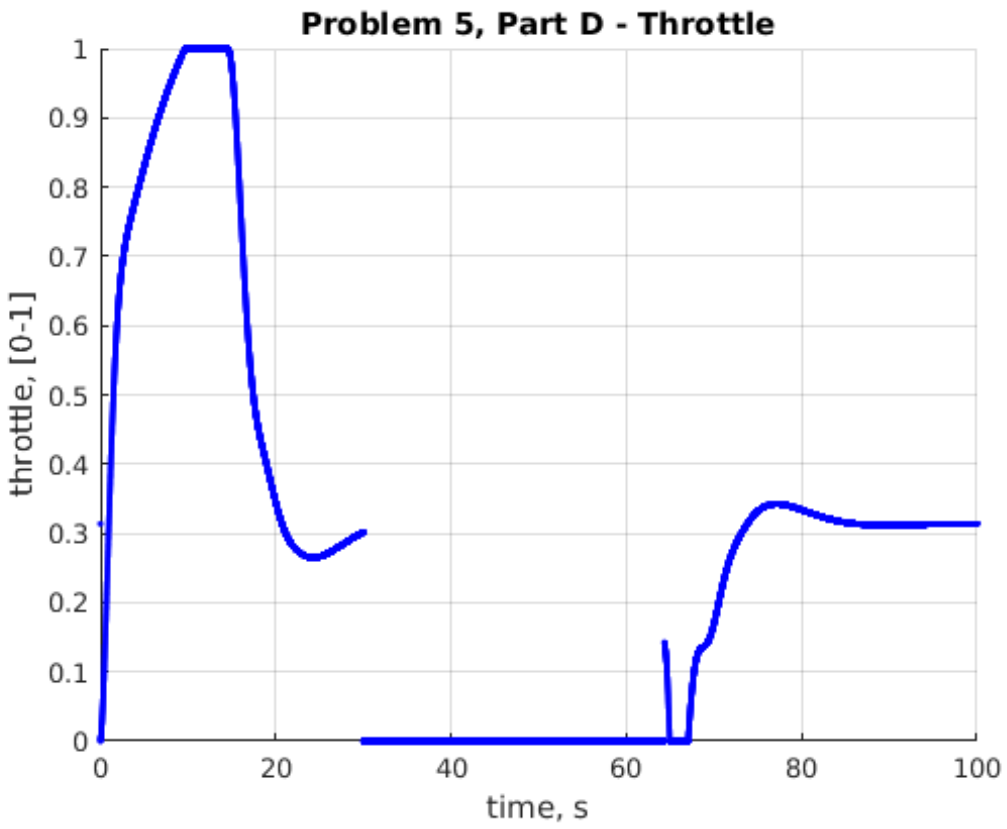
```
kp =
    0.1469
ki =
    0.0361
kd =
     0
```

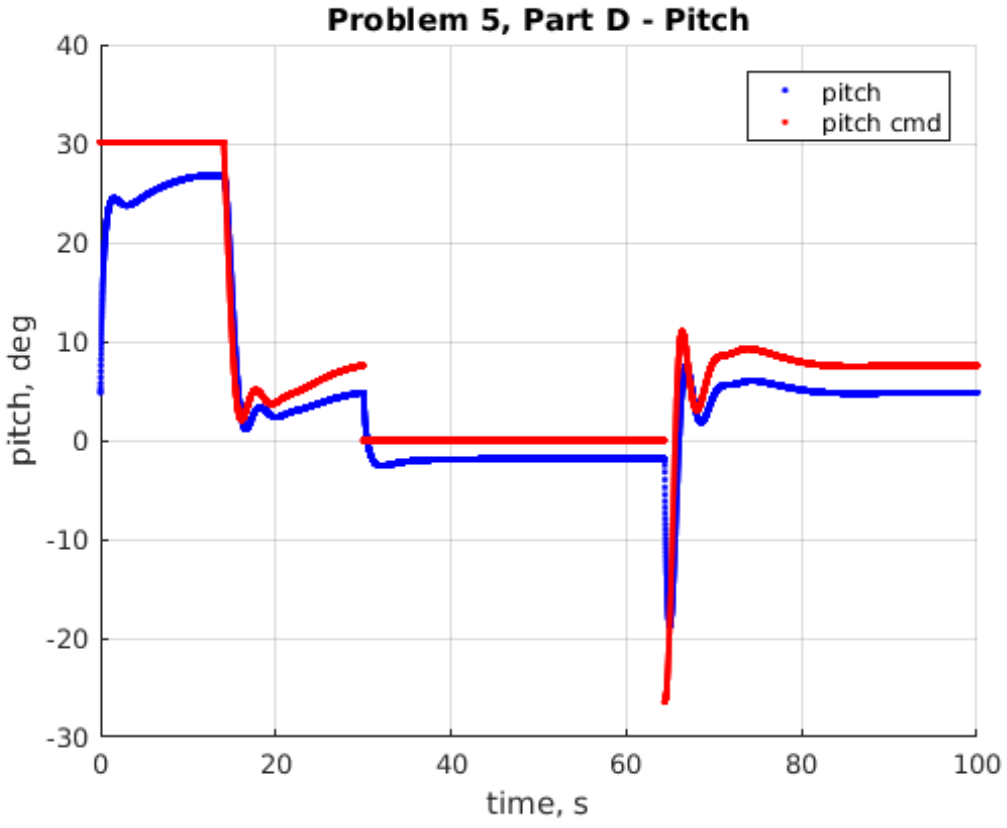
5.c.)

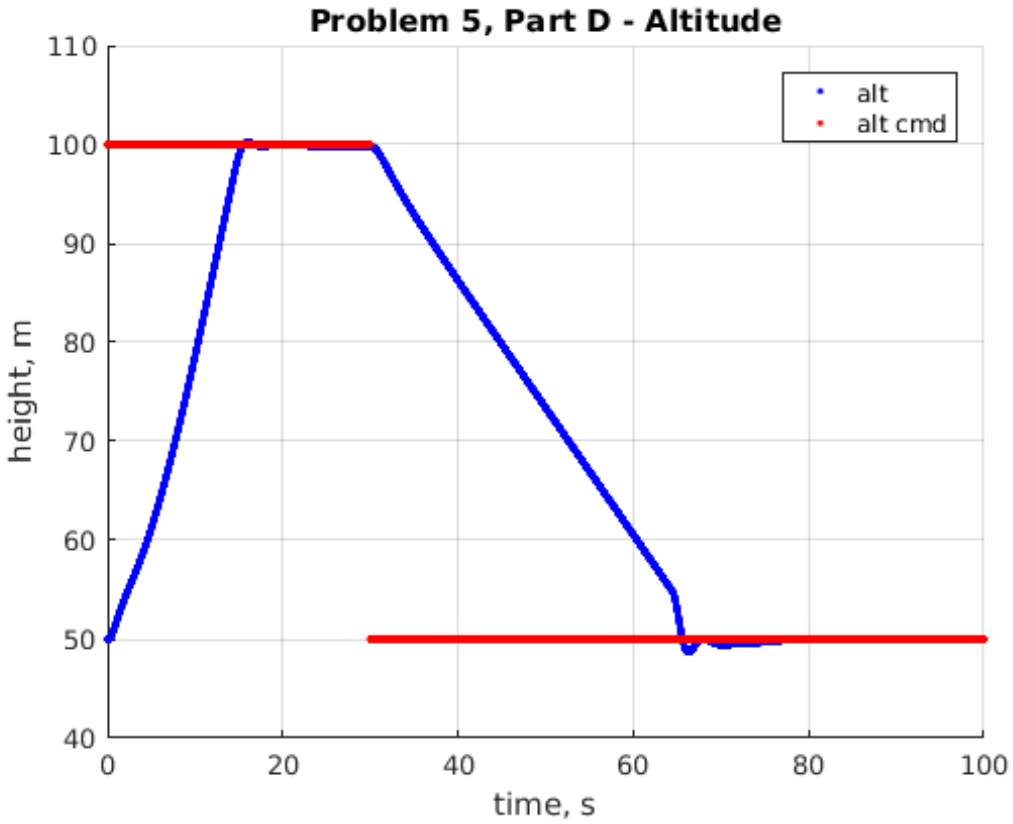
```
if(firstTime)
    PIR_pitch_hold(0,0,0,firstTime, P);
    PIR_alt_hold_using_pitch(0,0,0,firstTime, P);
    PIR_airspeed_hold_using_throttle(0,0,0,firstTime, P);
    PIR_airspeed_hold_using_pitch(0,0,0,firstTime, P);
end
h_hold = 5;
if h_hat < h_hold
    % Climb Logic
    delta_t = 1;
    theta_c = PIR_airspeed_hold_using_pitch(Va_c, Va_hat, 0.0, firstTime, P);
elseif h_hat > h_c + h_hold
    % Descend Logic
    delta_t = 0;
    theta_c = PIR_airspeed_hold_using_pitch(Va_c, Va_hat, 0.0, firstTime, P);
else
    % Altitude Hold Logic
    delta_t = PIR_airspeed_hold_using_throttle(Va_c, Va_hat, 0.0, firstTime,
P);
    theta_c = PIR_alt_hold_using_pitch(h_c, h_hat, 0, firstTime, P);
end
```

5.d.)


Problem 5, Part D - Airspeed
airspeed
airspeed cmd
speed, m/s
time, s
17
16
15
14
13
12
11
10
9
0
20
40
60
80
100


Problem 5, Part D - Throttle
throttle, [0-1]
time, s
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
20
40
60
80
100


Problem 5, Part D - Pitch
pitch, deg
time, s
pitch
pitch cmd
40
30
20
10
0
-10
-20
-30
0
20
40
60
80
100



5.e.)
I would make sure there is no undershoot when going vehicle is commanded to go down in altitude.