Design of a Quadcopter for Payload Delivery.

By: Adam Haj.
Major: Mechanical Engineer.
University: UMES
Date: July 28th, 2018
Advisor: Dr Payam Matin.
Project Goals

- To design a Quadcopter capable of flying to a specific location, dropping off a package and returning to the original location autonomously.
Design and Prototype

- Must be able to transport a package of approximately 3 lb.
- Must be able to move autonomously from one place to another specified location.
- Lithium polymer battery powered.
- Flight time of at least 30 min.
- Package maximum size of 8 in. X 11 in. X 12 in.
Design Constraints

- Weight cannot exceed: 55 lbs (25Kg) with a drone registration.
- FAA Part 107 (Unmanned Aerial Vehicle Operation).
- Attitude cannot exceed: 400 ft AGL.
- Cost must be less than: $300.
- Must stay in the line of sight.
The Prototype
Cont.....
SolidWorks Modeling
Governing Equations

• The CLMEs.

\[
\begin{align*}
m(\dot{U} + QW - RV) &= -mg\sin\theta + (F_{Ax} + F_{Tz}) \\
m(\dot{V} + UR - PW) &= mg\cos\theta\sin\phi + (F_{Ay} + F_{Ty}) \\
m(\dot{W} + PV - QU) &= mg\cos\theta\cos\phi + (F_{Az} + F_{Tz}) \\
\dot{I}_{xx} - \dot{R}_{xx} - PQI_{xz} + RQ(I_{zz} - I_{yy}) &= I_A + I_T \\
\dot{I}_{yy} + PR(I_{xx} - I_{zz}) + (P^2 - R^2)I_{xz} &= M_A + M_T \\
\dot{I}_{zz} - \dot{R}_{xz} + PQ(I_{yy} - I_{xx}) + QRI_{xz} &= N_A + N_T
\end{align*}
\]

• The CAMEs.
Cont....

• Kinematic equations.

• Flight Path equations.
Matlab Code & Simulation

clear
This is the code meant to perform flight simulations for a Quadrocopter
for the University of Maryland Engineering 2016 Senior Design

Class. The purpose of this code is to return the variables
u1, v1, w1, a1, ps1, theta, p1, q1, r1, and 1. This will show the behavior of the
system

Defined by the Equations of Motion described in the Design Calculations
Version. (Equations of motion were created in relation to body frame P)

Developed by Addy We for the senior design 2016.

% constant
At (dt=0.01):
Ft = 1;
else
Ft = 0
end

q = 0.01; % gravity
qfsq = q.; tis the increase in thrust of us eigenvalues

T = 4.5; ;
let Jss = 1.0
Nt = .4/5; % Fz = z

n = 1.5 ; % theta


w2 = [v w b phi theta psi]

w1(1) = v
w1(2) = w
w1(3) = b
w1(4) = phi
w1(5) = theta
w1(6) = psi
w1(7) = phi
w1(8) = theta
w1(9) = phi

w2(1) = w2(11) - 0.1 * w1(9); w2(11) = w2(11) - 0.1 * w1(9)
MATLAB Code & Simulation

```matlab
subplot(4,2,1), plot(t, u(:,4), '-r', t,u(:,5), '-g', t,u(:,6), '-b');
grid on
title('Angular Velocity');
xlabel('t (sec)');
ylabel('P, Q, R (rad/s)');
legend('P', 'Q', 'R');

subplot(4,2,2), plot3( u(:,4), u(:,5), u(:,6));
grid on
title('Angular Velocity');

subplot(4,2,3), plot(t, u(:,7), '-r', t,u(:,8), '-g', t,u(:,9), '-b');
grid on
title('Euler Angles');
xlabel('t (sec)');
ylabel('phi, theta, psi (rad)');
legend('roll', 'pitch', 'yaw');

subplot(4,2,4), plot3( u(:,7), u(:,8), u(:,9));
grid on
title('Euler Angles');

subplot(4,2,5), plot(t, u(:,10), '-r', t,u(:,11), '-g', t,u(:,12), '-b');
grid on
title('Position');
xlabel('t (sec)');
ylabel('X, Y, Z (m)');
legend('X', 'Y', 'Z');

subplot(4,2,6), plot3( u(:,10), u(:,11), u(:,12));
grid on
title('Position');
```
The accomplishment

- MATLAB code and simulation on progress.
- Camera, servos and actuators being added to the system.
- Designing and building the prototype.
- Analytical design slide.
Acknowledgment

- Maryland Space Grant Consortium.
- University of Maryland Eastern Shore- Engineering Department.
Questions????