

---

```

function[dydt] = odequad(t,y)
global g m rad I_x I_y I_z k alpha eta f f1 f2 f3 f4 X Y Z
if y(13) == 0
    u_E = y(1);
    v_E = y(2);
    w_E = y(3);
    p = y(4);
    q = y(5);
    r = y(6);
    psi = y(7);
    theta = y(8);
    phi = y(9);
    N = y(10);
    E = y(11);
    D = y(12);
    cond = y(13);

    dudt = X/m - g*sin(theta) - q*w_E + r*v_E;
    dvdt = Y/m + g*cos(theta)*sin(phi) - r*u_E + p*w_E;
    dwdt = Z/m + g*cos(phi)*cos(theta) - p*v_E + q*u_E;
    dpdt = ((rad/sqrt(2))*(f1 + f2 - f3 - f4) - q*r*(I_z - I_y))*(1/
I_x);
    dqdt = ((rad/sqrt(2))*(f1 + f4 - f2 - f3) - r*p*(I_x - I_z))*(1/
I_y);
    drdt = (k*(f2 + f4 - f1 - f3) - p*q*(I_y - I_x))*(1/I_z);
    dphidt = p + (q*sin(phi) + r*cos(phi))*tan(theta);
    dthetadt = q*cos(phi) - r*sin(phi);
    dpsidt = (q*sin(phi) + r*cos(phi))*sec(theta);

    dydt(1) = dudt;
    dydt(2) = dvdt;
    dydt(3) = dwdt;
    dydt(4) = dpdt;
    dydt(5) = dqdt;
    dydt(6) = drdt;
    dydt(7) = dphidt;
    dydt(8) = dthetadt;
    dydt(9) = dpsidt;
    dydt(10) = u_E;
    dydt(11) = v_E;
    dydt(12) = w_E;
    dydt(13) = y(13);

    dydt = dydt';

end
%condition that introduces aero forces and moments on quadcopter
if y(13) == 1
    u_E = y(1);
    v_E = y(2);

```

---

---

```

w_E = y(3);
p = y(4);
q = y(5);
r = y(6);
psi = y(7);
theta = y(8);
phi = y(9);
N = y(10);
E = y(11);
D = y(12);

dudt = X/m - g*sin(theta) - q*w_E + r*v_E;
dvdt = Y/m + g*cos(theta)*sin(phi) - r*u_E + p*w_E;
dwdt = Z/m + g*cos(phi)*cos(theta) - p*v_E + q*u_E;
dpdt = ((rad/sqrt(2))*(f1 + f2 - f3 - f4) + -alpha*p^2 - q*r*(I_z
- I_y))*(1/I_x);
dqdt = ((rad/sqrt(2))*(f1 + f4 - f2 - f3) + -alpha*q^2 - r*p*(I_x
- I_z))*(1/I_y);
drdt = (k*(f2 + f4 - f1 - f3) + -alpha*r^2 - p*q*(I_y - I_x))*(1/
I_z);
dphidt = p + (q*sin(phi) + r*cos(phi))*tan(theta);
dthetadt = q*cos(phi) - r*sin(phi);
dpsidt = (q*sin(phi) + r*cos(phi))*sec(theta);

dydt(1) = dudt;
dydt(2) = dvdt;
dydt(3) = dwdt;
dydt(4) = dpdt;
dydt(5) = dqdt;
dydt(6) = drdt;
dydt(7) = dphidt;
dydt(8) = dthetadt;
dydt(9) = dpsidt;
dydt(10) = u_E*cos(theta)*cos(psi) +
v_E*(sin(phi)*sin(theta)*cos(psi) - cos(phi)*sin(psi)) +
w_E*(cos(phi)*sin(theta)*cos(psi) + sin(phi)*sin(psi));
dydt(11) = u_E*cos(theta)*sin(psi) +
v_E*(sin(phi)*sin(theta)*sin(psi) + cos(phi)*cos(psi)) +
w_E*(cos(phi)*sin(theta)*sin(psi) - sin(phi)*cos(psi));
dydt(12) = -u_E*sin(theta) + v_E*sin(phi)*cos(theta) +
w_E*cos(phi)*cos(theta);
dydt(13) = 0;

dydt = dydt';

```

end

*Not enough input arguments.*

*Error in odequad (line 3)*

*if y(13) == 0*

*Published with MATLAB® R2018b*