

Function that's used to plot results from linearized model for an airplane

This function takes in the time and results vectors from the ode45 function, as well as a title for plots and a letter that indicates what part of the problem is being plotted.

```
function [] = plutter(time, results, titlestring)
```

Extract the results from the results vector

```
global E1
Deltau = results(:,1);
Deltaw = results(:,2);
Deltaq = results(:,3);
Deltatheta = results(:,4);
DeltaX = results(:,5);
DeltaZ = results(:,6);
```

Create the figure for inertial position (only plotting Z and X components, as Y remains unchanged).

```
figure();
hold on
plot(DeltaX, DeltaZ);
plot(linspace(0,max(DeltaX),length(DeltaX)), linspace(E1.Altitude,E1.Altitude,length(DeltaX)), 'k');
xlabel('X Position [m]');
ylabel('Y Position [m]');
zlabel('Z Position [m]');
ti1 = title(['Inertial Position for ', titlestring]);
set(ti1, 'Interpreter','latex');
set(ti1, 'FontSize', 12);
legend('Inertial Position','Initial Altitude','Location','best');
grid on
hold off
```

Create a new figure using a tiled layout to plot the responses of the four states, and title it.

```
figure();
tiles = tiledlayout(4,1);
ti2 = title(tiles, ['Responses for ', titlestring]);
set(ti2, 'Interpreter', 'latex');
set(ti2, 'FontSize', 20);

% Tile 1, Delta u Response
nexttile;
plot(time, Deltau);
ylabel('m/s');
plot1title = title('$\Delta u, \frac{m}{s}$');
set(plot1title, 'Interpreter', 'latex');
set(plot1title, 'FontSize', 20);
grid on

nexttile;
plot(time(1:500), Deltaw(1:500));
ylabel('m/s');
```

```

plot1title = title('$\Delta w, \frac{m}{s}$');
set(plot1title, 'Interpreter', 'latex');
set(plot1title, 'FontSize', 20);
grid on

nexttile;
plot(time(1:500), Deltaq(1:500));
ylabel('rad/s');
plot1title = title('$\Delta q, \frac{rad}{s}$');
set(plot1title, 'Interpreter', 'latex');
set(plot1title, 'FontSize', 20);
grid on

nexttile;
plot(time, Deltatheta);
ylabel('rad/s');
plot1title = title('$\Delta \theta, \frac{rad}{s}$');
set(plot1title, 'Interpreter', 'latex');
set(plot1title, 'FontSize', 20);
grid on

xlabel(tiles, 'Time, [s]');

```

end