Table of Contents


```
function E = Etot(x,r,Cba,ba,params)

% Usage: E = Etot(x,r,Cba,ba,params)
%
% Description: Function takes in the output of the sim and outputs the
% rotational energy of the system as a time series
%
% Inputs:
%   x       -  13 x 1 or 18 x 1 matrix of the spacecraft state
%              where n is the number of time steps
%   r       -  time series of position vector magnitude
%   Cba     -  3 x 3 DCM
%   params  -  struct of simulation parameters
%
% Outputs:
%   E  -  total energy time series
```

Extract Necessary Parameters

```
gg_model = params.gg_model;
mag_model = params.mag_model;
atm_model = params.atm_model; %#ok<NASGU>
SRP_model = params.SRP_model; %#ok<NASGU>
J2on = params.J2on;

mu = params.Earth.mu_e;
R = params.Earth.Rmean;
J2 = params.Earth.J2const;
ms = params.sc.mWet;
I = params.sc.IB_b;
mb = params.sc.mom_b;
```

Useful Values

```
I3 = [0; 0; 1];
```

Check for J2 Inclusion

```
if J2on~=1
    J2=0;
else
```

```
    % do nothing
end
```

Calculate Total Energy

kinetic energy

```
T = 0.5*ms*x(4:6)'*x(4:6)+0.5*x(end-2:end)'*I*x(end-2:end);

% potential energy from fields and torques
% gravity well
V = -mu*ms/r+mu*ms*J2*R^2*(3*(x(1:3)'*I3)^2/2/r^2-0.5)/r^3;
% gravity gradient torque
if gg_model
    V = V+0.5*mu*(3*(Cba*x(1:3))'*I*Cba*x(1:3)/r^2-trace(I))/r^3;
end
% magnetic moment
if mag_model
    V = V-mb'*Cba*ba;
end

% total energy
E = T + V;
```

Published with MATLAB® R2019b