The Standard Atmosphere

1. Temperature is a specified function of altitude. Temperature is either constant with altitude, as in the stratosphere, or varies at a constant rate with altitude. This rate is called the lapse rate. In the troposphere the lapse rate is –3.56616degR/1000ft.
2. The troposphere is the lowest atmospheric region and extends from sea level to 36089.239 geopotential feet.
3. The stratosphere is the next highest atmospheric region and extends from 36,089.239 geopotential feet to 65,616.798 geopotential feet.
4. The tropopause is the dividing line between the troposphere and the stratosphere at an altitude of 36,089.239 geopotential feet.
5. The standard atmosphere ignores the fact that any given day might be hotter or colder than standard or have a higher or lower pressure than standard. In fact, in the standard atmosphere there are no weather variations at all. If we lived in the standard atmosphere at 600 ft elevation (for Lafayette, Indiana) the temperature would always be 56.86 degF, the pressure would always be 29.2769 inHg, and the density would be 0.00233546 slug/ft^3.
6. We will use the 1976 NASA Standard Atmosphere which has 8 atmospheric regions (temperature regions) up to 295,000 geopotential feet.
7. At sea level in the standard atmosphere
   - Pressure=p_o=2116.22 lbf/ft^2=29.92 inches of Hg (mercury)
   - Temperature=T_o=518.67 degR=59 degF [note: T(degR)=T(degF)+459.67]
   - Density=ρ_o=0.00237691 slug/ft^3 [note: when $g=g_o$ a mass of 1 slug will weigh 32.17405 lbf]
   - Gravitational constant g_o=32.17405 ft/sec^2
   - Relative humidity is zero (dry air).
   - R=1716.55 $ft^2/(sec^2 degR)$
8. Geopotential altitude, h, is related to geometric altitude (tapeline altitude), h_g, by
   - $h=(R_E/(R_E+h_g))h_g$ where R_E =Radius of the Earth=20,926,476 ft.
9. The equations for pressure and density in the Standard Atmosphere are based on the Equation of State for Air (i.e., the Perfect Gas Law, $p=\rho RT$ (equation 1.5)) and the hydrostatic equation (equation 1.4).
10. In the Troposphere (h<36089.239 geopotential feet)
    - T=518.67-.00356616*h degR
    - p=2116.22*(1-6.87558563248308e-06*h)^(5.25591641274834) lbf/ft^2
    - ρ=0.00237691267925741*(1-6.87558563248308e-06*h)^(4.25591641274834) slug/ft^3
11. In the Stratosphere(36089.239 geopotential feet<h<65616.798239 geopotential feet)
    - T=389.97degR
    - p=472.675801650081*exp(-4.80637968933164e-05*(h-36089.239)) lbf/ft^2
    - ρ=0.000706115448911997*exp(-4.80637968933164e-05*(h-36089.239)) slug/ft^3
12. Commonly we write Standard Atmosphere properties in terms of the ratio of a quantity to the sea level value of that quantity.
    - Pressure ratio $\delta=p/p_o$
    - Density ratio $\sigma=\rho/\rho_o$
    - Temperature ratio $\theta=T/T_o$
    - Equation of State for Air $\delta=\sigma\theta$
13. The speed of sound, a, is given by
    - $a=sqrt(\gamma RT)=sqrt(\gamma p/\rho)$ where γ =1.4 for air.
    - Note Altimeter at right with Kohlsman window showing 29.92 in Hg)

```
function
[temp,press,rho,Hgeopvector]=atmosphere4(Hvector,GeometricFlag)
%function
[temp,press,rho,Hgeopvector]=atmosphere4(Hvector,GeometricFlag)
% Standard Atmospheric data based on the 1976 NASA Standard Atmoshere.
% Hvector is a vector of altitudes.
% If Hvector is Geometric altitude set GeometricFlag=1.
% If Hvector is Geopotential altitude set GeometricFlag=0.
% Temp, press, and rho are temperature, pressure and density
% output vectors the same size as Hgeomvector.
% Output vector Hgeopvector is a vector of corresponding geopotential
altitudes (ft).
% This atmospheric model is good for altitudes up to 295,000
geopotential ft.
% Ref: Intoduction to Flight Test Engineering by Donald T. Ward and
Thomas W. Strganac

% index   Lapse rate   Base Temp     Base Geopo Alt        Base
Pressure            Base Density
%   i     Ki(degR/ft)  Ti(degR)        Hi(ft)              P, lbf/ft^2
RHO, slug/ft^3
format long g
D= [1     -.00356616    518.67               0
2116.22       0.00237691267925741
    2        0          389.97        36089.239
472.675801650081      0.000706115448911997
    3      .00054864    389.97        65616.798
114.343050672041      0.000170813471460564
    4      .00153619    411.57       104986.878
18.1283133205764      2.56600341257735e-05
    5        0          487.17       154199.475
2.31620845720195      2.76975106424479e-06
    6     -.00109728    487.17       170603.675
1.23219156244977      1.47347009326248e-06
    7     -.00219456    454.17       200131.234
0.38030066501701      4.87168173794687e-07
    8        0          325.17       259186.352
0.0215739175227548      3.86714900013768e-08];

R=1716.55;  %ft^2/(sec^2degR)
gamma=1.4;
g0=32.17405;      %ft/sec^2
RE=20926476;      % Radius of the Earth, ft
K=D(:,2);   %degR/ft
T=D(:,3);   %degR
H=D(:,4);   %ft
P=D(:,5);   %lbf/ft^2
RHO=D(:,6); %slug/ft^3
temp=zeros(size(Hvector));
press=zeros(size(Hvector));
rho=zeros(size(Hvector));
Hgeopvector=zeros(size(Hvector));

% Convert from geometric altitude to geopotental altitude, if
necessary.
if GeometricFlag
      Hgeopvector=(RE*Hvector)./(RE+Hvector);
```

```
      disp('Convert from geometric altitude to geopotential altitude in
feet')
else
   Hgeopvector=Hvector;
   %disp('Input data is geopotential altitude in feet')
end
ih=length(Hgeopvector);
n1=find(Hgeopvector<=H(2));
n2=find(Hgeopvector<=H(3) & Hgeopvector>H(2));
n3=find(Hgeopvector<=H(4) & Hgeopvector>H(3));
n4=find(Hgeopvector<=H(5) & Hgeopvector>H(4));
n5=find(Hgeopvector<=H(6) & Hgeopvector>H(5));
n6=find(Hgeopvector<=H(7) & Hgeopvector>H(6));
n7=find(Hgeopvector<=H(8) & Hgeopvector>H(7));
n8=find(Hgeopvector<=295000 & Hgeopvector>H(8));
icorrect=length(n1)+length(n2)+length(n3)+length(n4)+length(n5)+length(
n6)+length(n7)+length(n8);
if icorrect<ih
      disp('One or more altitutes is above the maximum for this
atmospheric model')
      icorrect
      ih
end
% Index 1, Troposphere, K1= -.00356616
if length(n1)>0
      i=1;
      h=Hgeopvector(n1);
      TonTi=1+K(i)*(h-H(i))/T(i);
      temp(n1)=TonTi*T(i);
      PonPi=TonTi.^(-g0/(K(i)*R));
      press(n1)=P(i)*PonPi;
      RonRi=TonTi.^(-g0/(K(i)*R)-1);
      rho(n1)=RHO(i)*RonRi;
end
% Index 2,  K2= 0
if length(n2)>0
      i=2;
      h=Hgeopvector(n2);
      temp(n2)=T(i);
      PonPi=exp(-g0*(h-H(i))/(T(i)*R));
      press(n2)=P(i)*PonPi;
      RonRi=PonPi;
      rho(n2)=RHO(i)*RonRi;
end
% Index 3,  K3= .00054864
if length(n3)>0
      i=3;
      h=Hgeopvector(n3);
      TonTi=1+K(i)*(h-H(i))/T(i);
      temp(n3)=TonTi*T(i);
      PonPi=TonTi.^(-g0/(K(i)*R));
      press(n3)=P(i)*PonPi;
      RonRi=TonTi.^(-g0/(K(i)*R)-1);
      rho(n3)=RHO(i)*RonRi;
end
% Index 4,  K4= .00153619
if length(n4)>0
      i=4;
```

```
      h=Hgeopvector(n4);
      TonTi=1+K(i)*(h-H(i))/T(i);
      temp(n4)=TonTi*T(i);
      PonPi=TonTi.^(-g0/(K(i)*R));
      press(n4)=P(i)*PonPi;
      RonRi=TonTi.^(-g0/(K(i)*R)-1);
      rho(n4)=RHO(i)*RonRi;
end
% Index 5,  K5= 0
if length(n5)>0
      i=5;
      h=Hgeopvector(n5);
      temp(n5)=T(i);
      PonPi=exp(-g0*(h-H(i))/(T(i)*R));
      press(n5)=P(i)*PonPi;
      RonRi=PonPi;
      rho(n5)=RHO(i)*RonRi;
end
% Index 6,  K6= -.00109728
if length(n6)>0
      i=6;
      h=Hgeopvector(n6);
      TonTi=1+K(i)*(h-H(i))/T(i);
      temp(n6)=TonTi*T(i);
      PonPi=TonTi.^(-g0/(K(i)*R));
      press(n6)=P(i)*PonPi;
      RonRi=TonTi.^(-g0/(K(i)*R)-1);
      rho(n6)=RHO(i)*RonRi;
end
% Index 7,  K7= -.00219456
if length(n7)>0
      i=7;
      h=Hgeopvector(n7);
      TonTi=1+K(i)*(h-H(i))/T(i);
      temp(n7)=TonTi*T(i);
      PonPi=TonTi.^(-g0/(K(i)*R));
      press(n7)=P(i)*PonPi;
      RonRi=TonTi.^(-g0/(K(i)*R)-1);
      rho(n7)=RHO(i)*RonRi;
end
% Index 8,  K8= 0
if length(n8)>0
      i=8;
      h=Hgeopvector(n8);
      temp(n8)=T(i);
      PonPi=exp(-g0*(h-H(i))/(T(i)*R));
      press(n8)=P(i)*PonPi;
      RonRi=PonPi;
      rho(n8)=RHO(i)*RonRi;
end
```