A single compartment model is a low-pass filter

Prove that the single compartment model with fixed flow and volume and a sinusoidal input function forms a low pass filter reducing high frequency content with a fall-off of amplitude as a function of frequency at 6 dB per octave.

The ordinary differential equation is

$V \cdot dC/dt = F \cdot (C_{in} - C)$ where V is the volume, F is the flow and $C_{in} = A \cdot \cos(k \cdot t)$.
An appropriate initial condition is $C(0)=0$.

The solution (solved by using Maple) is

```
> dsolve({V*diff(C(t),t)=F*(A*cos(k*t)-C(t)),C(0)=0},C(t));

                         F t   2
                  exp(- ---) F  A
                         V           F A (F cos(k t) + k V sin(k t))
          C(t) = - --------------- + -------------------------------
                        2    2  2                2    2  2
                       F  + k  V                F  + k  V
```

The first term exponentially decays to zero, leaving an oscillating solution which is dominated by $\frac{(F/V) \cdot A \cdot k \cdot \sin(k \cdot t)}{(F/V)^2 + k^2}$. As *k* gets large, the amplitude falls off as *1/k*.

An octave means a doubling of the frequency. Decibels, abbreviated *dB*, is defined as

$$dB = 10 \cdot \log_{10}\left(\frac{Amplitude^2}{Reference\ Amplitude^2}\right) \quad .$$

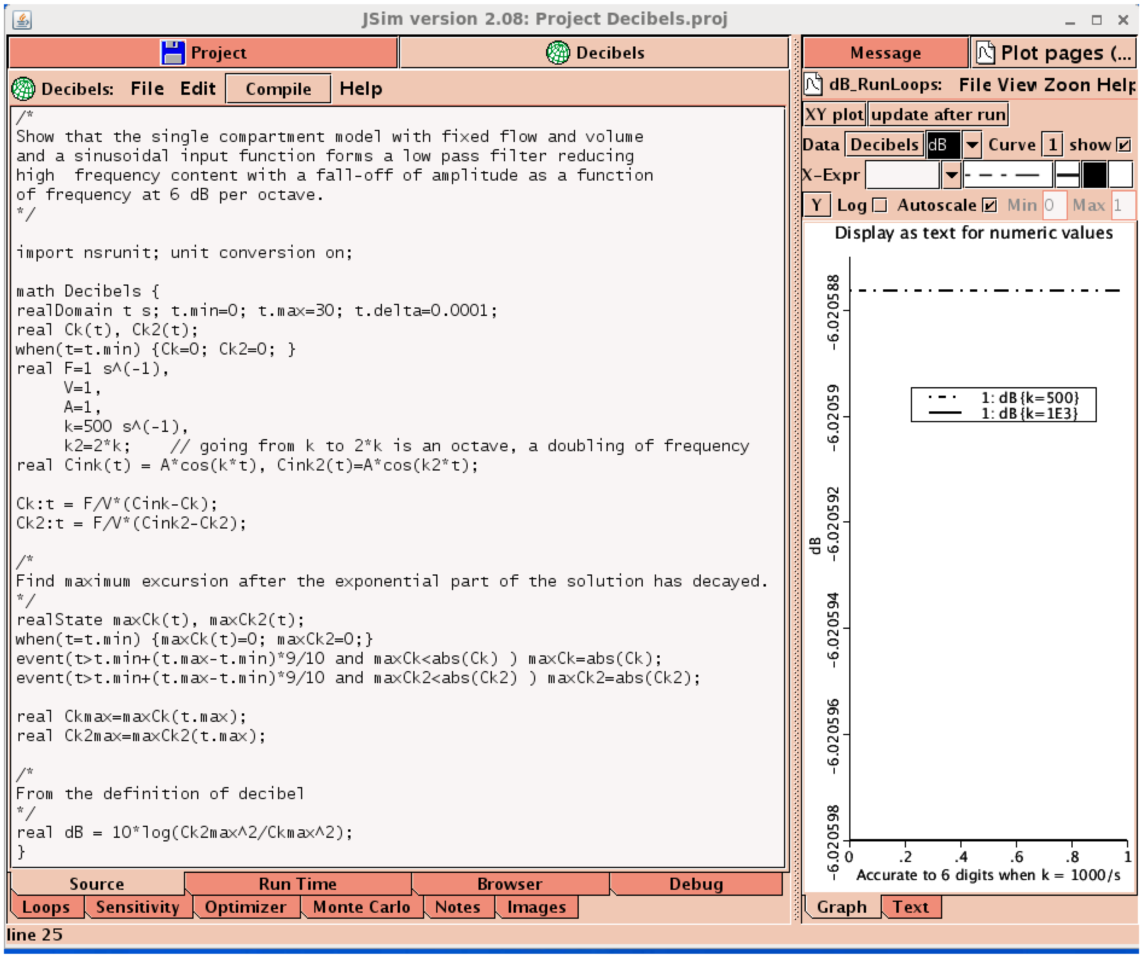
Therefore,

$$dB = 10 \cdot \log_{10}\left(\frac{1/(2 \cdot k)^2}{(1/k^2)}\right) \quad .$$

Therefore, for high frequencies, the signal is attenuated

$$dB = 10 \cdot \log_{10}(0.25) = -6.020599913$$

i.e. down *6 dB* going from *k* to *2k*, an octave.


JSim version 2.08: Project Decibels.proj
Project
Decibels
Message
Plot pages (...
Decibels: File Edit Compile Help
dB_RunLoops: File View Zoom Help
/*
Show that the single compartment model with fixed flow and volume
and a sinusoidal input function forms a low pass filter reducing
high frequency content with a fall-off of amplitude as a function
of frequency at 6 dB per octave.
*/
import nsrunit; unit conversion on;
math Decibels {
realDomain t s; t.min=0; t.max=30; t.delta=0.0001;
real Ck(t), Ck2(t);
when(t=t.min) {Ck=0; Ck2=0; }
real F=1 s^(-1),
V=1,
A=1,
k=500 s^(-1),
k2=2*k; // going from k to 2*k is an octave, a doubling of frequency
real Cink(t) = A*cos(k*t), Cink2(t)=A*cos(k2*t);
Ck:t = F/V*(Cink-Ck);
Ck2:t = F/V*(Cink2-Ck2);
/*
Find maximum excursion after the exponential part of the solution has decayed.
*/
realState maxCk(t), maxCk2(t);
when(t=t.min) {maxCk(t)=0; maxCk2=0;}
event(t>t.min+(t.max-t.min)*9/10 and maxCk<abs(Ck)) maxCk=abs(Ck);
event(t>t.min+(t.max-t.min)*9/10 and maxCk2<abs(Ck2)) maxCk2=abs(Ck2);
real Ckmax=maxCk(t.max);
real Ck2max=maxCk2(t.max);
/*
From the definition of decibel
*/
real dB = 10*log(Ck2max^2/Ckmax^2);
}
XY plot update after run
Data Decibels dB Curve 1 show
X-Expr
Y Log Autoscale Min 0 Max 1
Display as text for numeric values
1: dB {k=500}
1: dB {k=1E3}
dB
-6.020588
-6.02059
-6.020592
-6.020594
-6.020596
-6.020598
0 .2 .4 .6 .8 1
Accurate to 6 digits when k = 1000/s
Source Run Time Browser Debug
Loops Sensitivity Optimizer Monte Carlo Notes Images
Graph Text
line 25