Approximation of a Logarithmic Pot
Using an Op-amp and a Linear Pot

Selected APL symbols for non-users of APL

APL is case sensitive. i.e. x and X are not the same object.
+-×÷ mean the same as you were taught in elementary school.
Since APL can deal with a list of numbers (vector), the minus sign
  cannot be used to indicate a negative constant. Instead ¯ is used.
  Thus 5-2 ¯3 is 3 8. ¯ is never used with a variable.
+/ means add up what follows, ×/ means multiply them together
⌈/ gives the largest, ⌊/ gives the smallest e.g. ⌈/ 2 3 ¯4 is 3
← is assignment e.g. x←2×3 gives x the value 6 whenever refered to
⍕ is format, used here mostly for simultaneous assignment and display
⋆ is the power function NOT MULTIPLICATION! 3⋆2 is 9 not 6
⋆0.5 is used for square root. Pythagarus is written (+/x y⋆2)⋆0.5
APL can do multiple calculations so +/x y⋆2 sums the squares of x and y
More complicated operations can be kept in named "functions", e.g. Solve
  Int (numerical integration) and Plot. If you care how they work, ask.
⍝ what follows is a comment i.e. plain text at the end of an APL line
⋄ is used to separate short related statements on one line

I found that a reasonable approximation of a logarithmic transfer function suitable for audio tone and volume controls could be obtained using the circuit shown in Fig. 1.

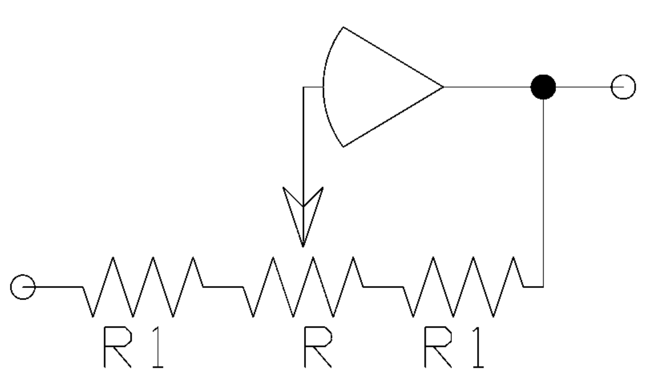
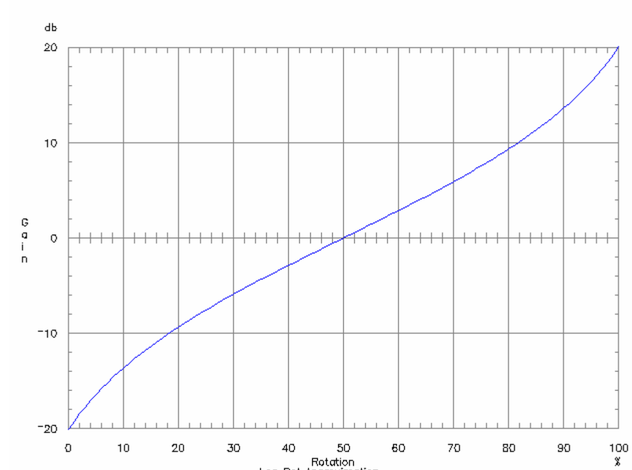


Fig. 1.

Choosing R=10K and R1=1.1K gives a slightly better than +/-20db range.

```
      R←10E3 ⋄ R1←1.1E3 ⋄ a←100 step 0 1
      PP←'' 'Log Pot Approximation' 'Rotation' 'Gain' '%' 'db'
      PP Plot (20×10⍟(R1+a×R)÷R1+(1-a)×R)Versus 100×a
```



Reducing R1 to 470 gives increased range at the expense of linearity of the db calibration.

```
R1←470
PP Plot (20×10⍟(R1+a×R)÷R1+(1-a)×R)Versus 100×a
```