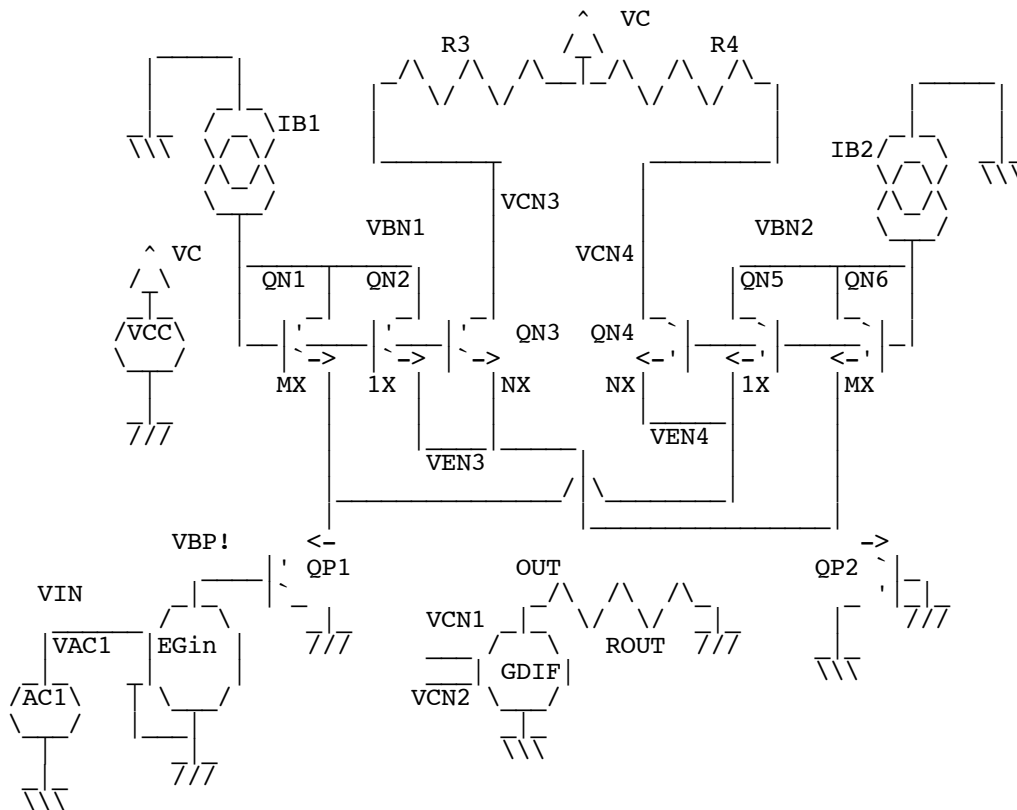


AB_BIAS_DIFF_Thd_NX

* www.idea2ic.com
 * dsauersanjose@aol.com 4/23/08



```
.OPTIONS GMIN=1e-18 METHOD=trap srcsteps = 1 gminsteps = 1
*=====
VCC      VC      0      DC      10
VAC1     VIN     0      DC      0      SIN( 0      1      1000 )
IB1      0      VBN1     1u
IB2      0      VBN2     1u
QN1      VBN1   VBN1   VE4    NPN1   4.50
QN2      VBN1   VBN1   VE3    NPN1   1.00
QN3      VCN3   VBN1   VE3    NPN1   15.00
QN4      VCN4   VBN2   VE4    NPN1   15.00
QN5      VBN2   VBN2   VE4    NPN1   1.00
QN6      VBN2   VBN2   VE3    NPN1   4.50

QP1      0      VBP1   VE3    PNP1   1.00
QP2      0      0      VE4    PNP1   1.00
R3       VCN3   VC      1K
R4       VCN4   VC      1K
E_DIF    OUT    0      VCN3   VCN4   1
ROUT     OUT    0      1K
E_GAININ VBP1   0      VIN    0      1m
```

```
.control
tran      25u      5m      0      1u
plot      out
echo      "THD% versus VIN_vpk "
setplot   new
set       NameList = (      Nx8 Nx15 Nx25 Nx30 )
compose   NxVals   values  8  15  25  30
compose   VinVals  values  10m 30m 35m 40m 60m 70m 80m 100m 110m 120m 140m 150m
170m 185m 200m 250m
settype   voltage  VinVals
```

```

let          NoOfNx  =  length(NxVals)
let          NoOfVin =  length(VinVals)

begin
unset                interrupt
* =====Loop_Nx=====
let          j      =  1
while        (j      <=  NoOfNx )
let          Nx     =  NxVals[j-1]
alter       QN3    area =  $&Nx
alter       QN4    area =  $&Nx
set         thisName =  $NameList[$&j]
let         $thisName =  0*vector(NoOfVin)

* =====Loop_Vin=====
let          k      =  1
while        (k      <=  NoOfVin )
let          Vin    =  VinVals[k-1]
alter       e_gain gain =  $&Vin
tran       25u          5m          0          1u
linearize
set         specwindow=  "blackman"
spec       200      8k      200      v(out)
let        thdsq =mag(out[9])^2 +mag(out[14])^2 +mag(out[19])^2 +mag(out[24])^2
let        thd_percent= 100*sqrt(thdsq)/mag(out[4])
echo       "$&unknown.Vin      $&thd_percent"
let        unknown.{ $thisName}[unknown.k-1] = thd_percent
repeat    3
destroy
end
if        ($?interrupt)
goto     bail
endif
let        k =          k + 1
endwhile
setscale  VinVals
plot      $NameList loglog title "AB_BIAS THD_% vs Vin_pK and Nx"
let        j =          j + 1
endwhile
label    bail
echo     "Done."
end
.endc

*=====
.model    NPN1    NPN(    BF=2100 VAF=216 )
.model    PNP1    PNP(    BF=2100 VAF=21 )
.end

```

=====END_OF_SPICE=====

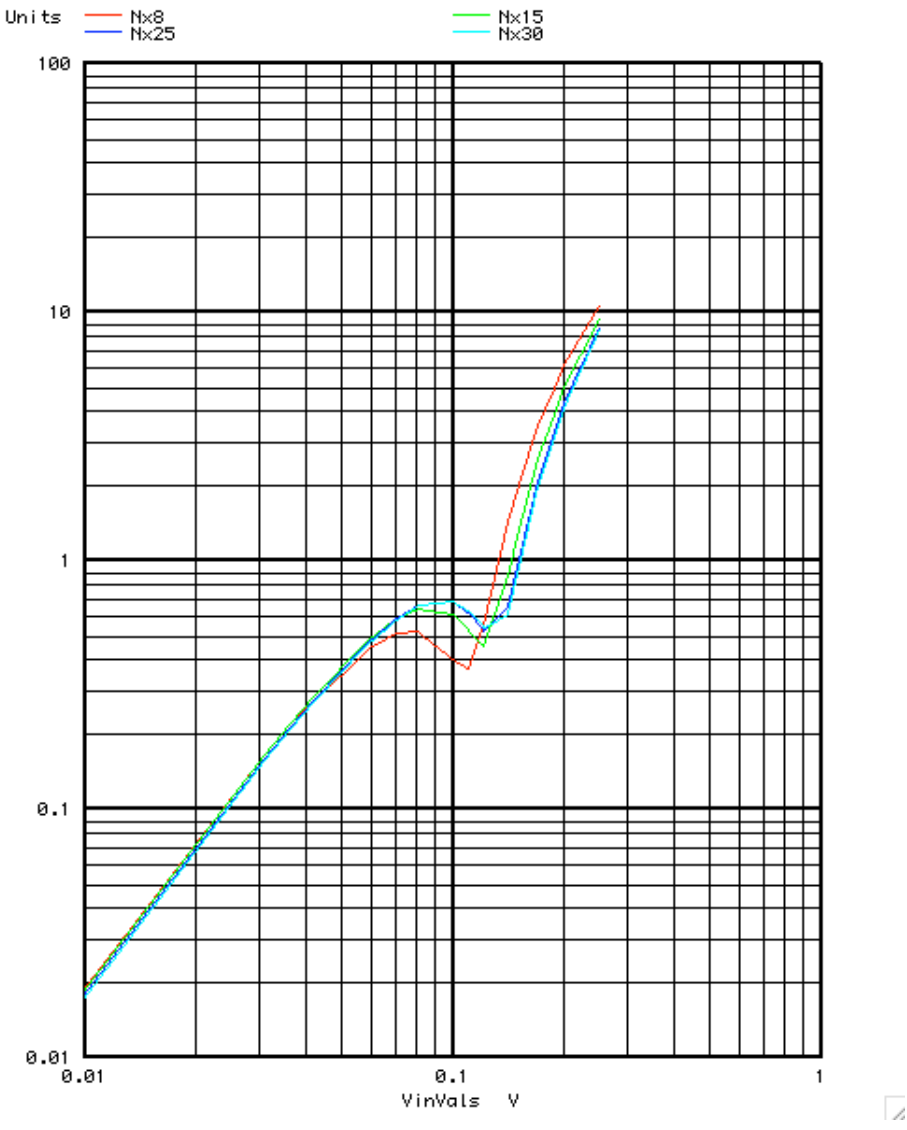
To Covert PDF to plain text click below
<http://www.fileformat.info/convert/doc/pdf2txt.htm>

This simulation only works on MacSpice for now.
Data in spice apparently gets stored in vectors which
are ready to be plotted.

The invention below is an AB_biased differential input
stage which can put out more differential current than
it draws DC wise. The current gain is defined by the
N ratio. The A bias of the AB_Biased aspect is defined by
the M area ratio term.

*

Graph 2 - unknown2: AB_BIAS THD_% vs Vin_pK and Nx



It is the ability for the output current to be able to be larger than the DC current which gives the benefits of effective lower noise and offset. But just being able to handle larger input signal voltages has its benefits was well.

The AB_Biased input stage needs to be put into a LM3080 OTA circuit to do an actual comparison in performances.