

The background of the slide features a light gray circuit board pattern with various traces and circular components. A solid black horizontal band runs across the middle of the image, serving as a backdrop for the text.

# iMonkey Spice

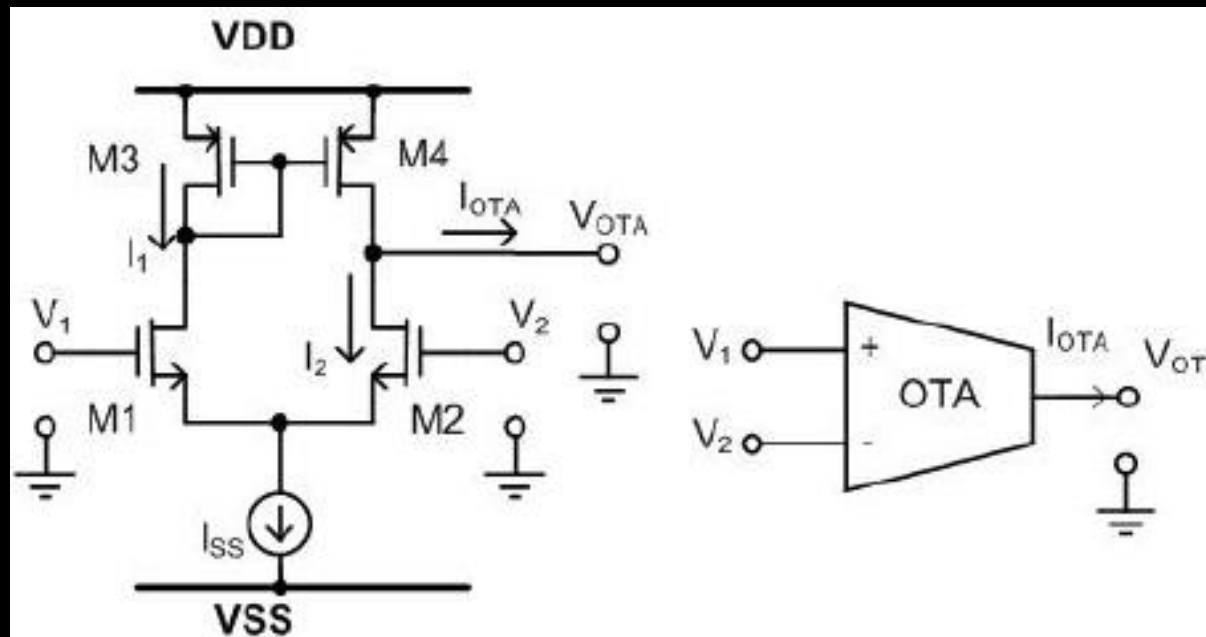
Proyecto Final – Análisis y Diseño de Algoritmos

# Contenido:

- Descripción
- Justificación
- Desarrollo
- Resultados
- Trabajo a Futuro

# Descripción

- iMonkey Spice hace uso de los algoritmos genéticos para apoyar en el diseño de un OTA Básico basado en las especificaciones de Ganancia y Ancho de Banda
- Un OTA ([Operational Transconductance Amplifier](#)) es uno de los bloques principales en el diseño de Circuitos Integrados Analógicos (Similar a un Opamp)



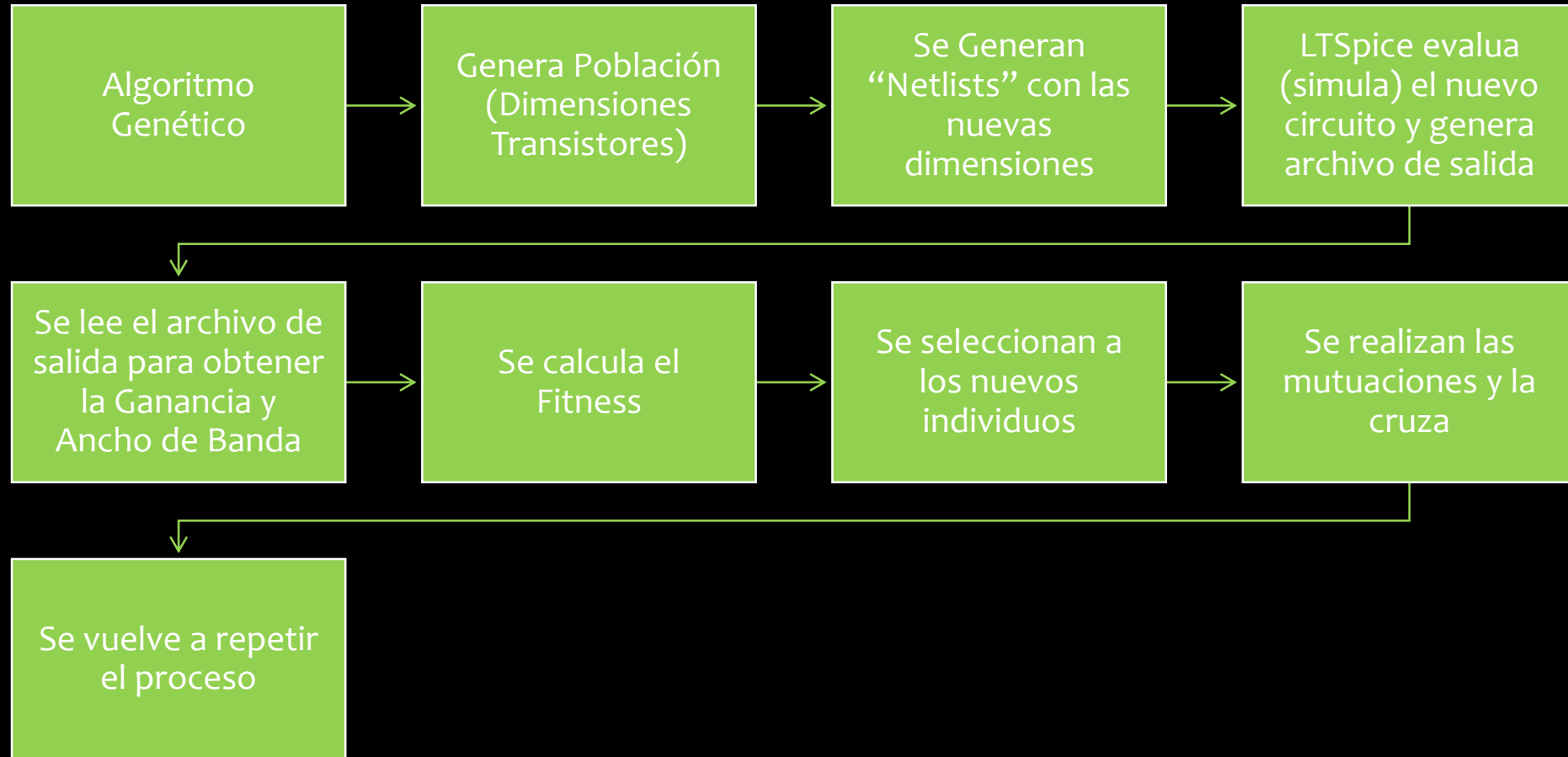
# Justificación

- El diseño de un OTA podría parecer algo trivial ya que ya se cuentan fórmulas para apoyarnos en el diseño.
- El diseño consiste en seleccionar el Ancho (W) y Largo (L) del transistor NMOS y PMOS.
- Las fórmulas con las que se cuenta están basadas en modelos de primer nivel (se asumen muchas cosas). En la vida real, estas fórmulas sirven sólo como primer paso y hay que “sintonizar” las dimensiones de los transistores para cumplir con las especificaciones requeridas.

Estado	Condición	Intensidad de drenador
CORTE	$V_{GS} < V_{tn}$	$I_D = 0$
LINEAL	$V_{GS} > V_{tn} \quad V_{DS} < V_{GS} - V_{tn}$	$I_D = \mu_n C'_{ox} \frac{W}{L} \left( V_{GS} - V_{tn} - \frac{V_{DS}}{2} \right) V_{DS}$
SATURACIÓN	$V_{GS} > V_{tn} \quad V_{DS} \geq V_{GS} - V_{tn}$	$I_D = \frac{\mu_n C'_{ox}}{2} \frac{W}{L} (V_{GS} - V_{tn})^2 (1 + \lambda V_{DS})$

Generales	
Carga del Electrón	$q = 1.602 \times 10^{-19} \text{ C}$
Constante de Boltzman	$k = 1.38 \times 10^{-23} \text{ J/K}$
Permisividad espacio libre	$\epsilon_0 = 8.854 \times 10^{-12} \text{ F/m}$
Tensión equival. a la temperatura	$V_T = kT/q$
Silicio	
Permisividad relativa del Silicio	$\epsilon_{Si} = 11.8$
Permisividad relativa del óxido de Silicio	$\epsilon_{SiO_2} = 3.9$
Concentración intrínseca a 300°K	$n_i = 1.1 \times 10^{16} \text{ m}^{-3}$
Movilidad de los electrones	$\mu_n = 0.05 \text{ m}^2 / (V \cdot S)$
Movilidad de los huecos	$\mu_p = 0.02 \text{ m}^2 / (V \cdot S)$
MOS	
Tensión efectiva de puerta	$V_{eff} = V_{GS} - V_{tn}$
Tensión Umbral	$V_{tn} = V_{tn0} + \gamma \left( \sqrt{V_{SB} +  2\phi_F } - \sqrt{ 2\phi_F } \right)$
Potencial en la superficie	$\phi_F = \frac{kT}{q} \ln \left( \frac{N_A}{n_i} \right)$
Constante de efecto sustrato	$\gamma = \frac{\sqrt{2q\epsilon_{ox}\epsilon_{Si}N_A}}{C'_{ox}}$
Capacidad de puerta por unidad de área	$C'_{ox} = \frac{\epsilon_{ox}\epsilon_0}{t_{ox}}$
Constante de efecto de modulación de canal	$\lambda \propto \frac{1}{L \sqrt{V_{DS} - V_{eff} + \phi_0}}$
Ganancia del transistor	$\beta_n = K_n \frac{W}{L} = \mu_n C'_{ox} \frac{W}{L}$

# Desarrollo



# Desarrollo

## Python

Algoritmo  
Genético

Función a  
Evaluar

## Función a Evaluar

Generación  
Netlists

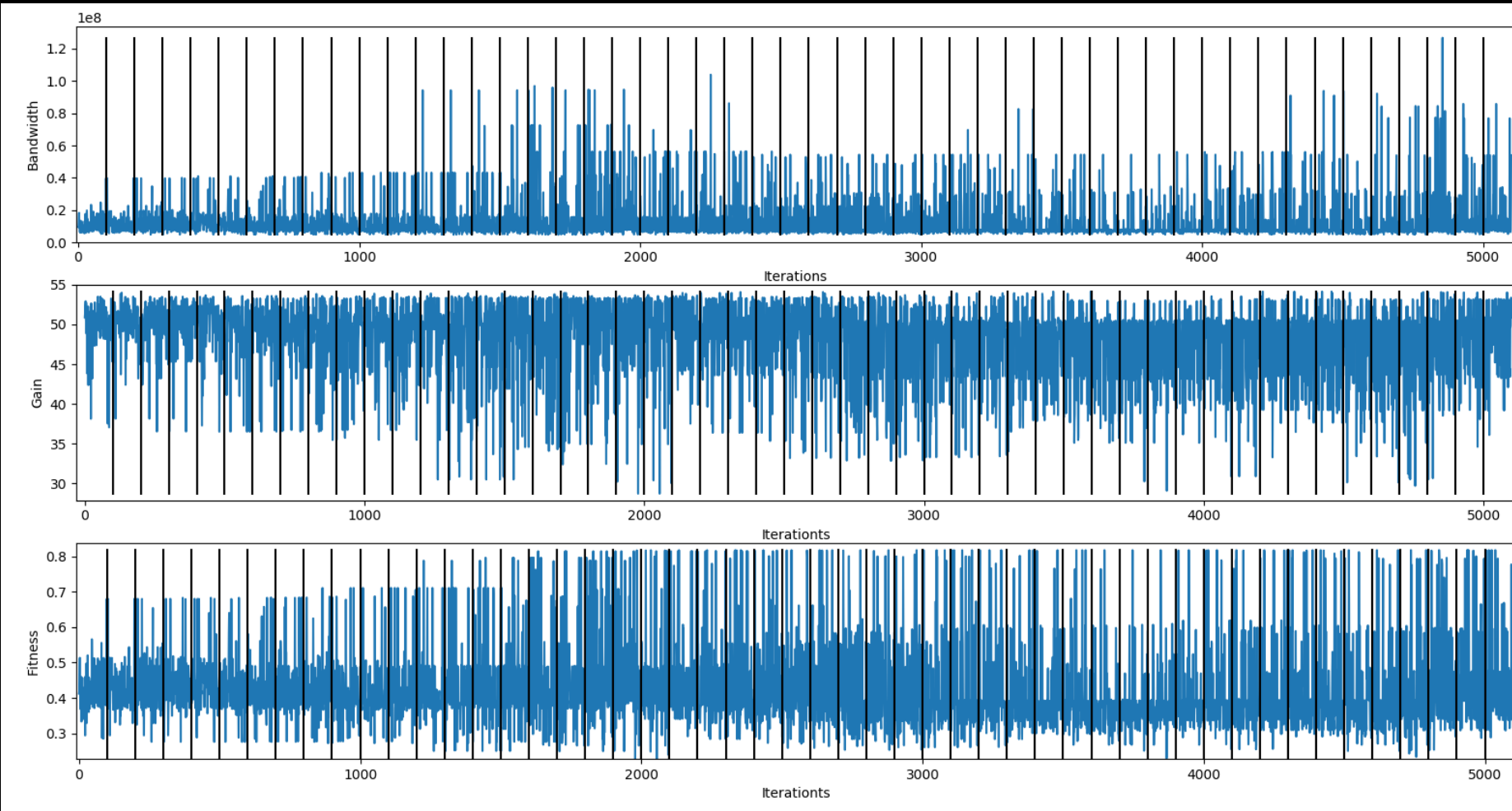
Lectura de  
Resultados

## LTSpice

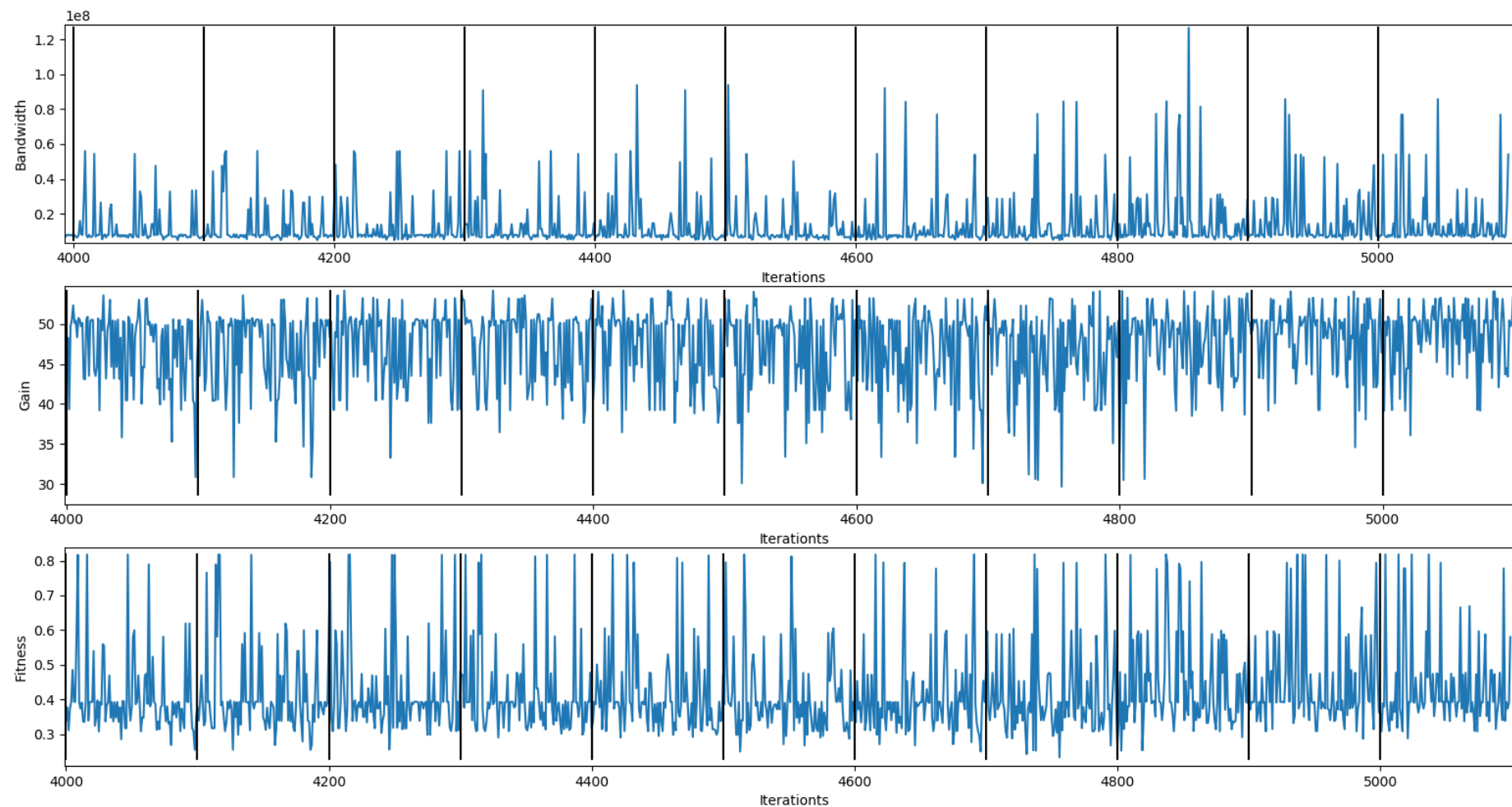
Simula el Netlist

Genera archivo de  
Resultados

# Resultados – 50 Generaciones, 100 Individuos

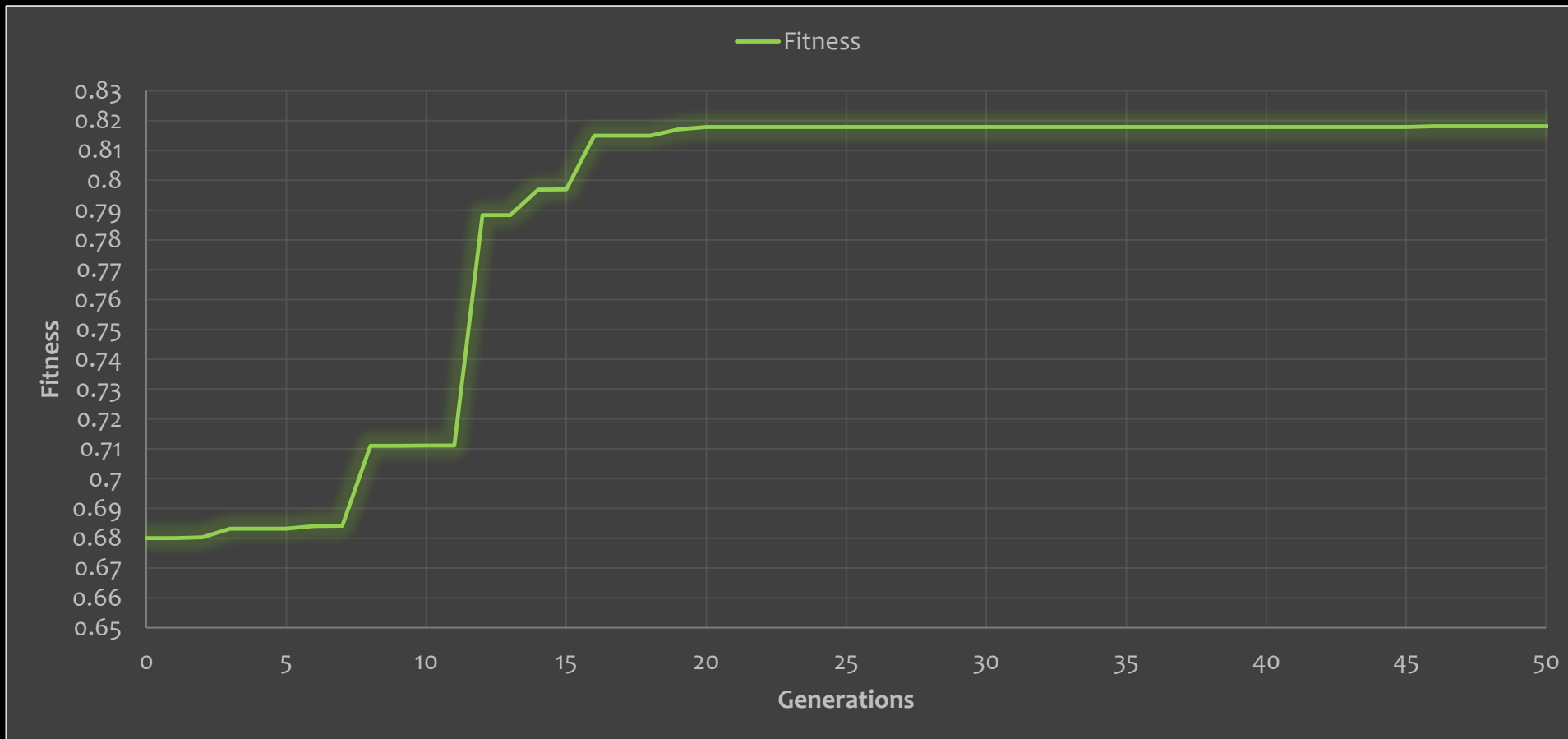


# Resultados





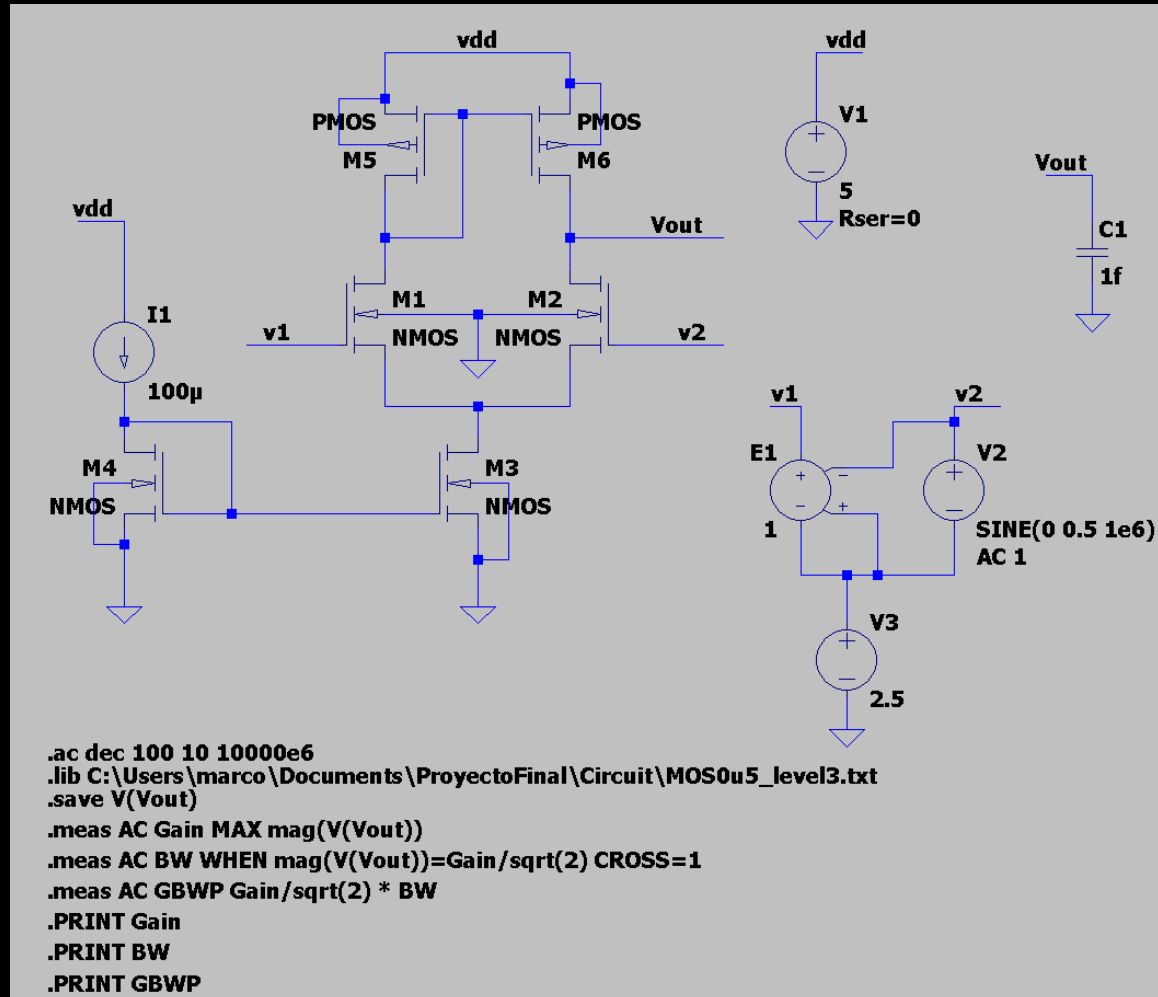
# Resultados - Fitness



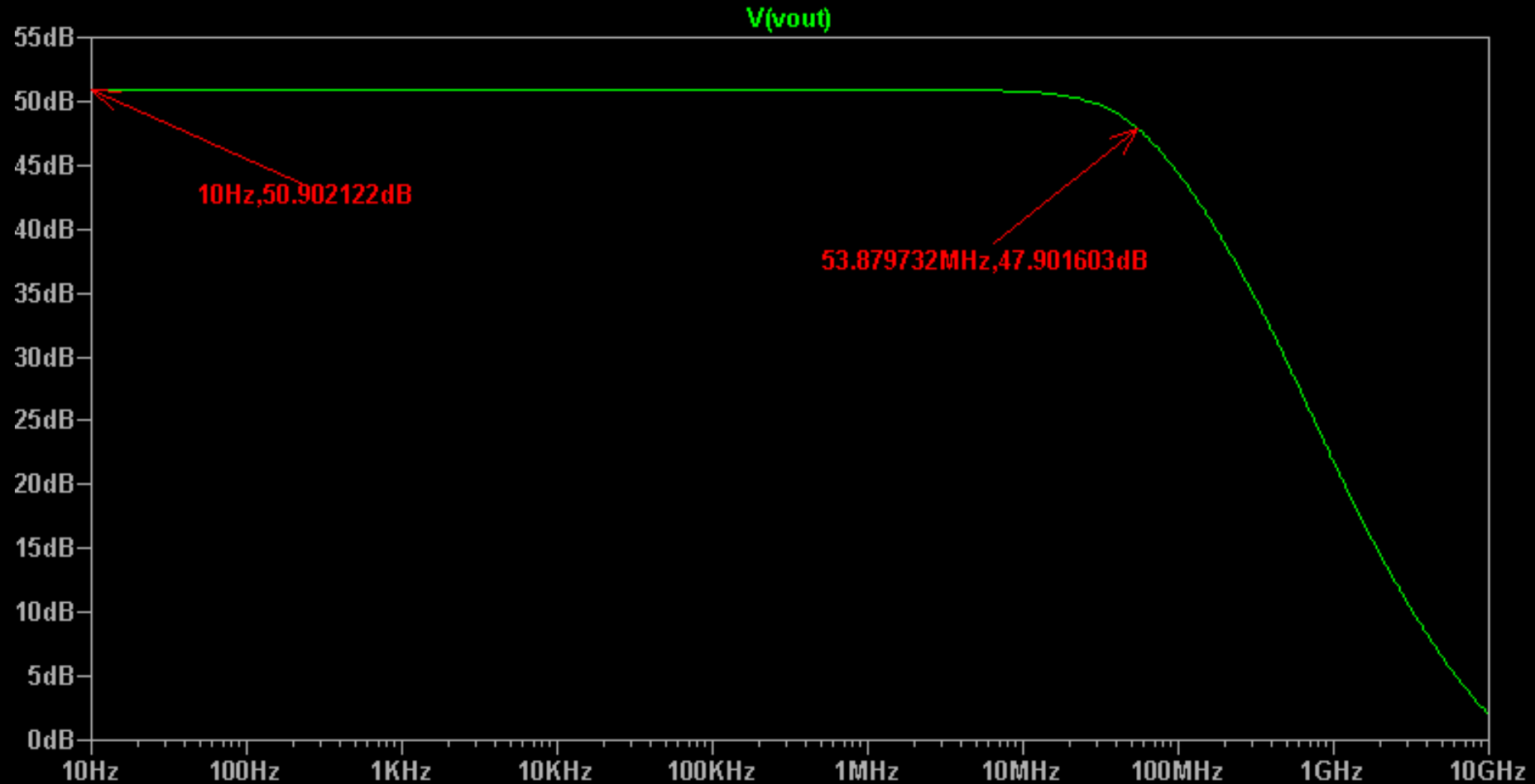
Best:  $f(22.3814619317, 1.2638109054) = [54000800.0, 50.9021]$ , fitness = 0.818138125

```
Run_80dB_50MHz_Results.txt
1  Generation: 0
2  --> BestIndividual found: 0.680037125
3  Generation: 1
4  --> BestIndividual found: 0.680037125
5  Generation: 2
6  --> BestIndividual found: 0.680269875
7  Generation: 3
8  --> BestIndividual found: 0.68322625
9  Generation: 4
10 Generation: 5
11 Generation: 6
12 --> BestIndividual found: 0.68413225
13 Generation: 7
14 --> BestIndividual found: 0.684141
15 Generation: 8
16 --> BestIndividual found: 0.710996875
17 Generation: 9
18 --> BestIndividual found: 0.711
19 Generation: 10
20 --> BestIndividual found: 0.711095875
21 Generation: 11
22 Generation: 12
23 --> BestIndividual found: 0.788309375
24 Generation: 13
25 Generation: 14
26 --> BestIndividual found: 0.796871875
27 Generation: 15
28 --> BestIndividual found: 0.796940625
29 Generation: 16
30 --> BestIndividual found: 0.814934375
31 Generation: 17
32 --> BestIndividual found: 0.814934375
33 Generation: 18
34 Generation: 19
35 --> BestIndividual found: 0.817119375
36 Generation: 20
37 --> BestIndividual found: 0.817899375
38 Generation: 21
39 Generation: 22
40 Generation: 23
41 Generation: 24
42 Generation: 25
43 Generation: 26
44 --> BestIndividual found: 0.81790625
45 Generation: 27
46 Generation: 28
47 Generation: 29
48 Generation: 30
49 --> BestIndividual found: 0.81790625
50 Generation: 31
51 Generation: 32
52 --> BestIndividual found: 0.81790625
53 Generation: 33
```

# Circuito Simulado



# Resultados de Simulación



# Trabajo a Futuro

- Correr el algoritmo genético tal como esta, evaluando de uno en uno los circuitos, toma mucho tiempo (1000 iteraciones por hora, aprox)
  - Como trabajo a futuro, se podría tener un netlist con 10 diferentes instancias el OTA con diferentes dimensiones, para correr la simulación de 10 en 10.
- Soportar diferentes arquitecturas de OTAs.
  - Se va a requerir modificar el código del algoritmo genético para poder generar mas de 2 phenotipos.
- Incluir mas mediciones en el Fitness
  - Comprobar que todos los transistores están en saturación.