

Tarea 1 - Electrónica Aplicada 1 - Alexander Daniel Rios

September 20, 2021

1 Tarea 1 - Electronica Aplicada 1

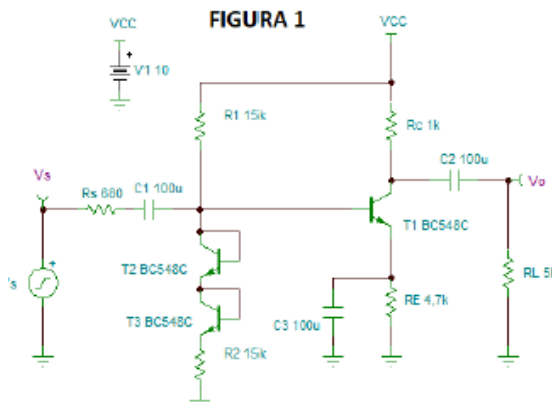
```
[3]: from IPython.display import Image

fig = Image(filename=('./enunciado.png'))
fig
```

[3]:

ENUNCIADO

El circuito de la figura 1 fue diseñado a una temperatura de $+25^{\circ}\text{C}$. Pero su rango de operación en el equipo en donde se desempeña es de $T = [-35^{\circ}\text{C}..+65^{\circ}\text{C}]$. Considerando que las leyes de variación con la temperatura de los componentes son:



Voltaje V_{CC} , aumenta 100mV cada $\Delta T = +20^{\circ}\text{C}$.

Voltaje $V_{BE}|_{T_0=25^{\circ}\text{C}} = 643\text{mV}$, $\frac{dV_{BE}}{dT} = -2.1 \frac{\text{mV}}{^{\circ}\text{C}}$.

Resistencias

$$R|_{T_f} = R|_{T_0} (1 + \alpha \Delta T), \quad \text{donde}$$
$$\Delta T = T_f - T_0 \text{ y } \alpha = 0.0005 \frac{1}{^{\circ}\text{C}}$$

$h_{FE}|_{T_0=25^{\circ}\text{C}} = 360$ y aumenta un 50% cada $\Delta T = +40^{\circ}\text{C}$.

1) Realizar un código MATLAB®, PYTHON® o similar que permita graficar:

1. $I_{CQ} = f(T)$
2. $V_{CEQ} = f(T)$
3. $A_{VS} = f(T)$, determinar $A_{VS}|_{T=25^{\circ}\text{C}}$
4. $R_{IA} = f(T)$
5. $V_{ce_max} = f(T)$

2) Simular el circuito en SPICE o TINA y verificar el valor de A_{VS} con el obtenido de $A_{VS}|_{T=25^{\circ}\text{C}}$ en el punto 1) ítem 3.

3) Extraer a conclusiones.

1.1 1) Resolucion - Script en Python

1.1.1 Modulos utilizados:

```
[4]: import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator
```

1.1.2 Constantes:

```
[5]: #=====CONSTANTES=====
TA = 25 #Temperatura ambiente en Celsius
TO = 273.15 #Temperatura ambiente en Kelvin
ALFA = 0.0005
K = 1.380649e-23 #Cte de Boltzmann
q = 1.602e-19 #Carga del electron
```

1.1.3 Funciones utilizadas:

```
[6]: def var_VCC(VCC_n, T):
    return VCC_n + (100e-3*(T-TA)/20)

def var_VBE(VBE_n, T):
    return VBE_n - (-2.1e-3*(T-TA))

def var_Resistencias(R_n, T):
    return R_n*(1+ALFA*(T-TA))

def var_hFE(hFE_n, T):
    return hFE_n + (0.5*hFE_n*(T-TA)/40)

def R_Paralelo(Ra, Rb):
    return (Ra*Rb) / (Ra+Rb)

def ICQ_T(VCC, VBE, R1, R2, RE, hFE, RB):
    return ( (VCC*(R2/(R1+R2)) ) + (VBE*(((2*R1)/(R1+R2))-1)) ) / ( RE + RB/hFE )

def VCEQ_T(VCC, ICQ, RC, RE):
    return VCC - ICQ * (RE + RC)

def rbe(VT, IBQ):
    ru = VT/IBQ #Resistencia de la juntura de un diodo
    rb = 0 #Resistencia del cuerpo(la considere nula)
    rbe = ru + rb
    return rbe
```

```

def AV_T(hFE, RD, RI):
    return -hFE*RD/RI

def AVS_T(RIA, RS, AV):
    return (AV*RIA)/(RIA+RS)

def VBB_T(VCC, VBE, R1, R2):
    return VCC*(R2/(R1+R2))+(2*VBE)*(R1/(R1+R2))

def VCE_MAX_T(VCEQ, ICQ, VCE_sat, RD):
    vomax1 = VCEQ - VCE_sat
    vomax2 = ICQ * RD
    if (vomax1 > vomax2).all():
        return vomax2
    else:
        return vomax1

```

1.1.4 Cálculo de factores que varían con la temperatura:

```

[7]: T = np.linspace(-35, 65, 1000) #Vector de temperaturas de -35 grados Celsius
                                     #a 65 grados Celsius con step=0.1

#####Resistencias normalizadas#####
R1_n = 15000
R2_n = 15000
RC_n = 1000
RE_n = 4700
RL_n = 5000
RS_n = 680

#####Tensiones#####
VCC_n = 10
VBE_n = 0.7
VCE_sat = 1

#####hFE del transistor#####
hFE_n = 360

#Tensiones, Corrientes y Componentes del circuito que varían con la temperatura

VCC = var_VCC(VCC_n, T)
VBE = var_VBE(VBE_n, T)
hFE = var_hFE(hFE_n, T)

R1 = var_Resistencias(R1_n, T)
R2 = var_Resistencias(R2_n, T)
RC = var_Resistencias(RC_n, T)
RE = var_Resistencias(RE_n, T)
RL = var_Resistencias(RL_n, T)

```

```

RS = var_Resistencias(RS_n, T)
RB = R_Paralelo(R1, R2)
RD = R_Paralelo(RC, RL)

ICQ = ICQ_T(VCC, VBE, R1, R2, RE, hFE, RB)

VCEQ = VCEQ_T(VCC, ICQ, RC, RE)

VT = K*(T0+T)/q
IBQ = ICQ/hFE
VBB = VBB_T(VCC, VBE, R1, R2)

RI = rbe(VT, IBQ)
RIA = R_Paralelo(RB, RI)

AV = AV_T(hFE, RD, RI)
AVS = AVS_T(RIA, RS, AV)

VCE_MAX = VCE_MAX_T(VCEQ, ICQ, VCE_sat, RD)
#=====
#=====Hallamos el punto para T=25°=====
i=0
for i,x in enumerate(T):
    if x > 24.9 and x <= 25:
        break

```

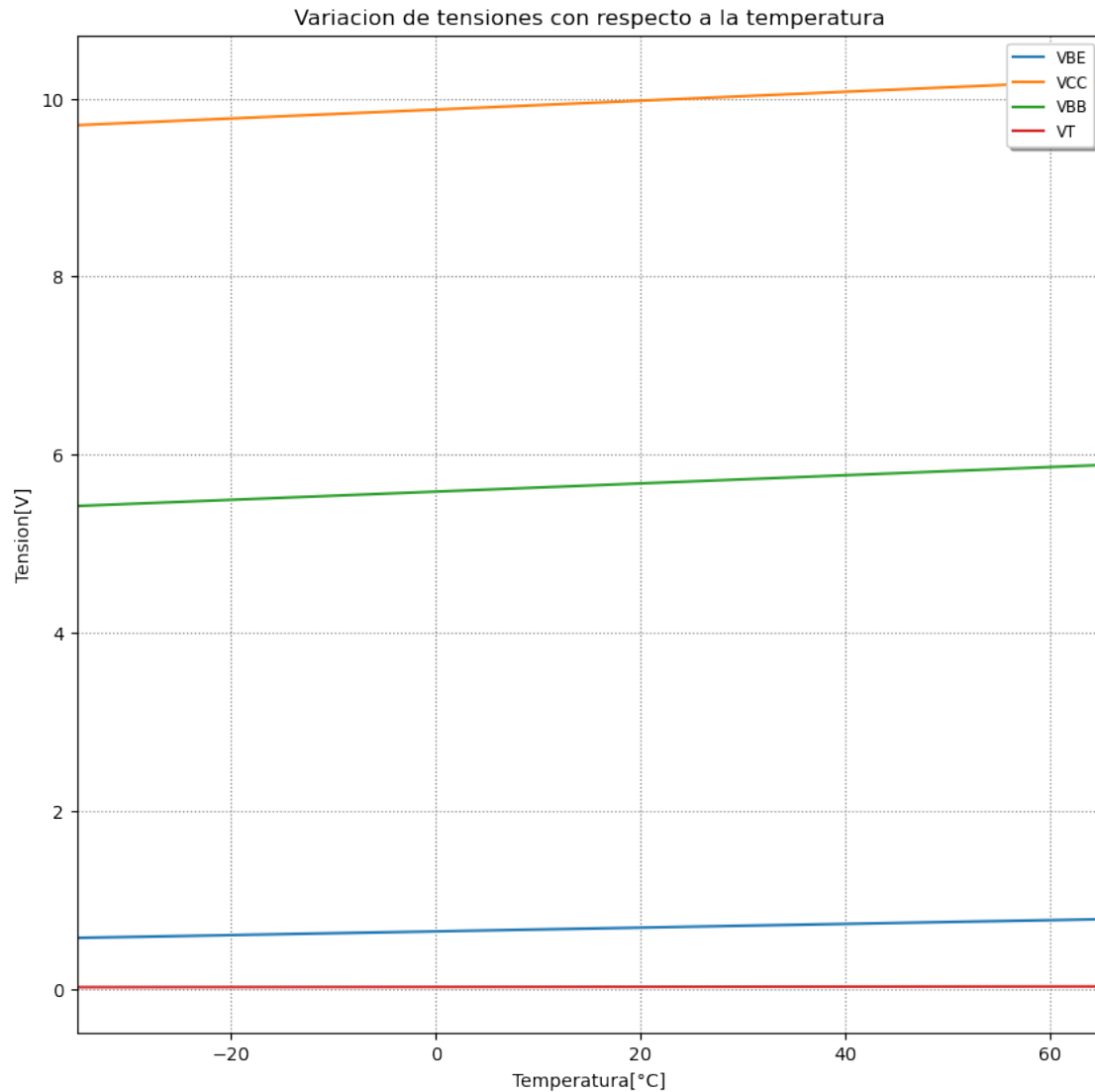
1.1.5 Graficos:

```

[8]: #%% Tensiones
plt.figure(1, figsize=(10, 10), dpi=100)
plt.plot(T, VBE, label='VBE')
plt.plot(T, VCC, label='VCC')
plt.plot(T, VBB, label='VBB')
plt.plot(T, VT, label='VT')
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de tensiones con respecto a la temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('Tension[V]')
plt.legend(loc='upper right', shadow=True, fontsize='small')

```

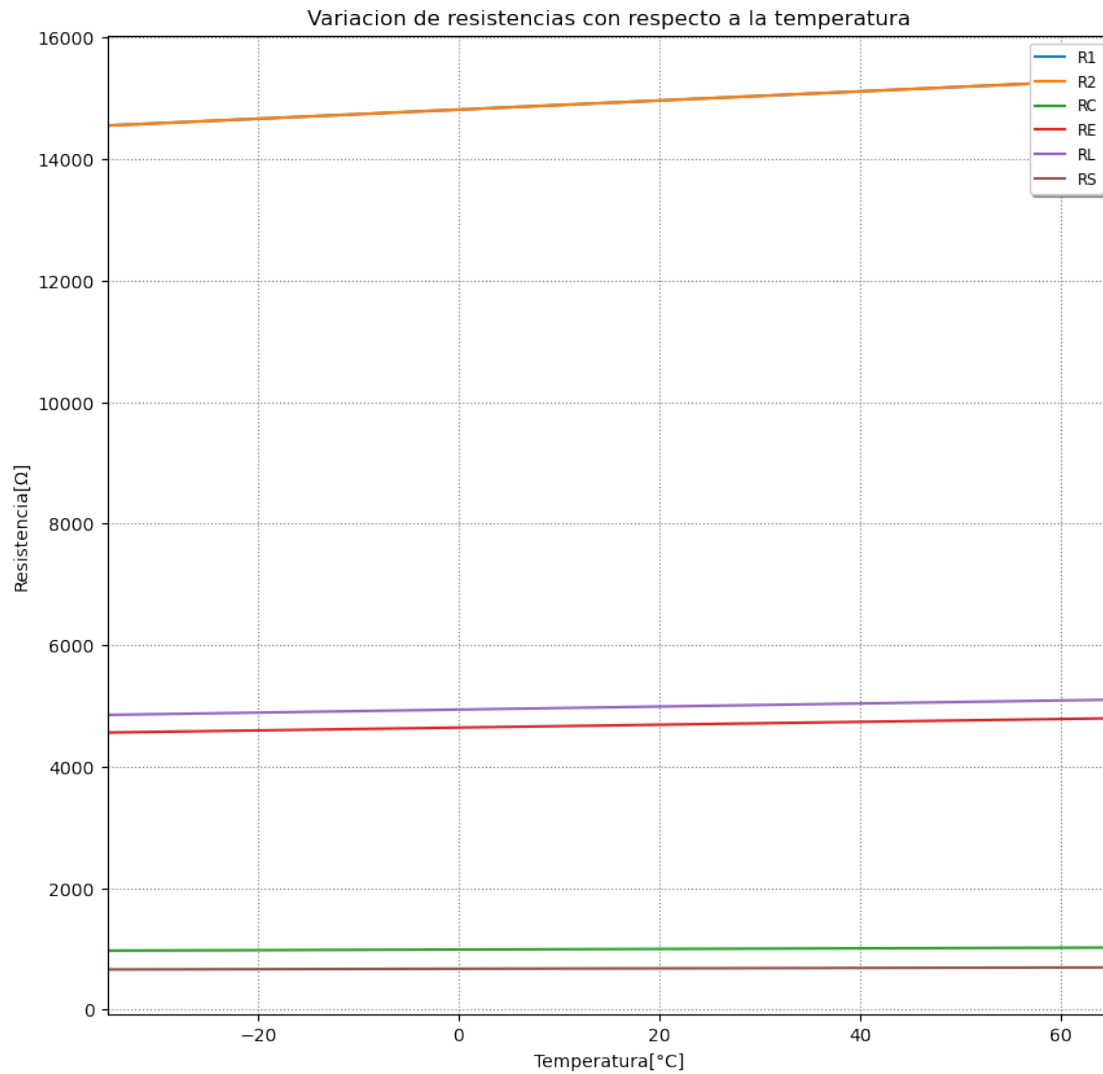
[8]: <matplotlib.legend.Legend at 0x20e3e257d30>



```
[9]: Resistencias
plt.figure(2, figsize=(10, 10), dpi=100)
plt.plot(T, R1, label='R1')
plt.plot(T, R2, label='R2')
plt.plot(T, RC, label='RC')
plt.plot(T, RE, label='RE')
plt.plot(T, RL, label='RL')
plt.plot(T, RS, label='RS')
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de resistencias con respecto a la temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('Resistencia[Ω]')
```

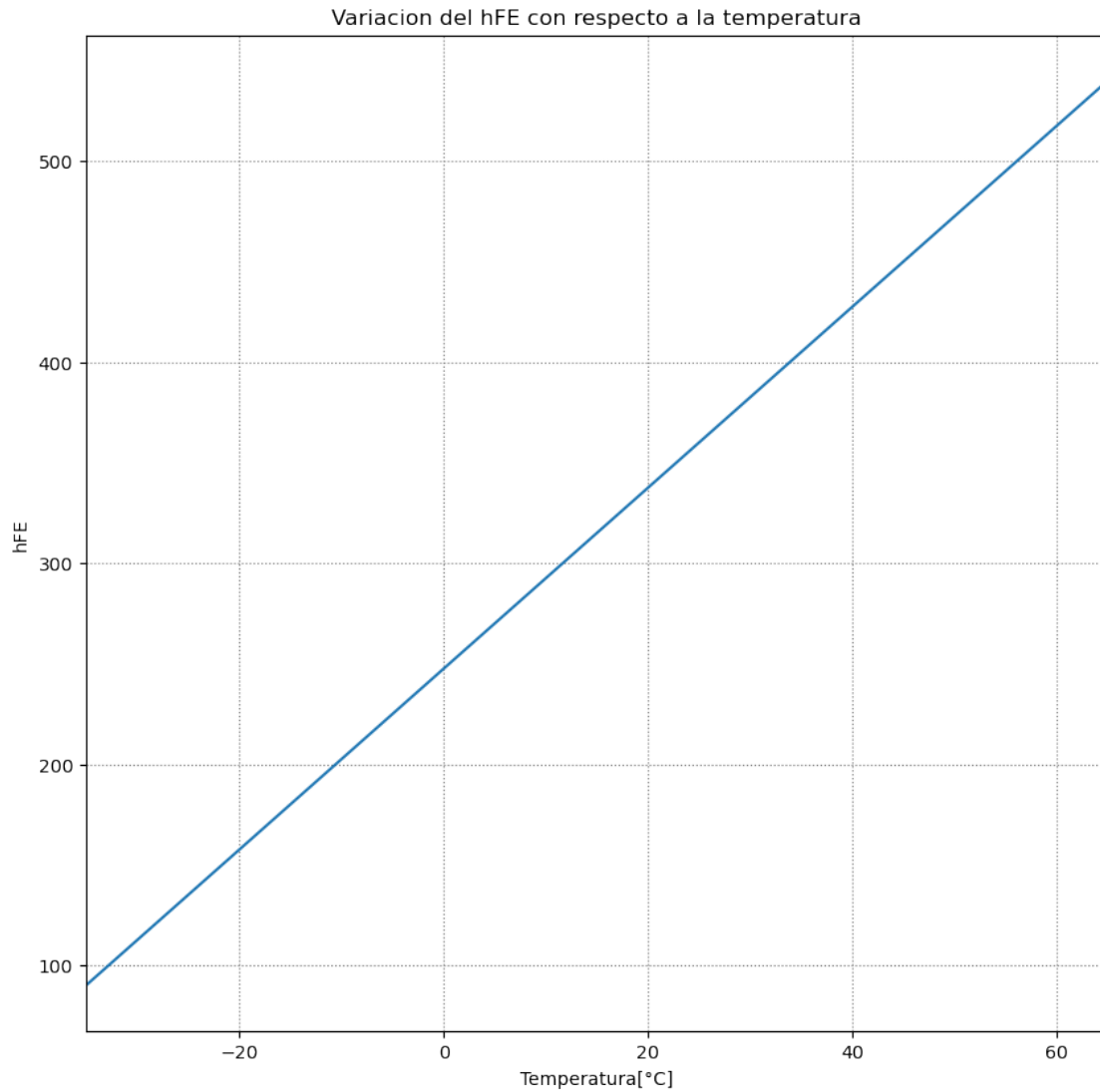
```
plt.legend(loc='upper right', shadow=True, fontsize='small')
```

[9]: <matplotlib.legend.Legend at 0x20e3f38e400>



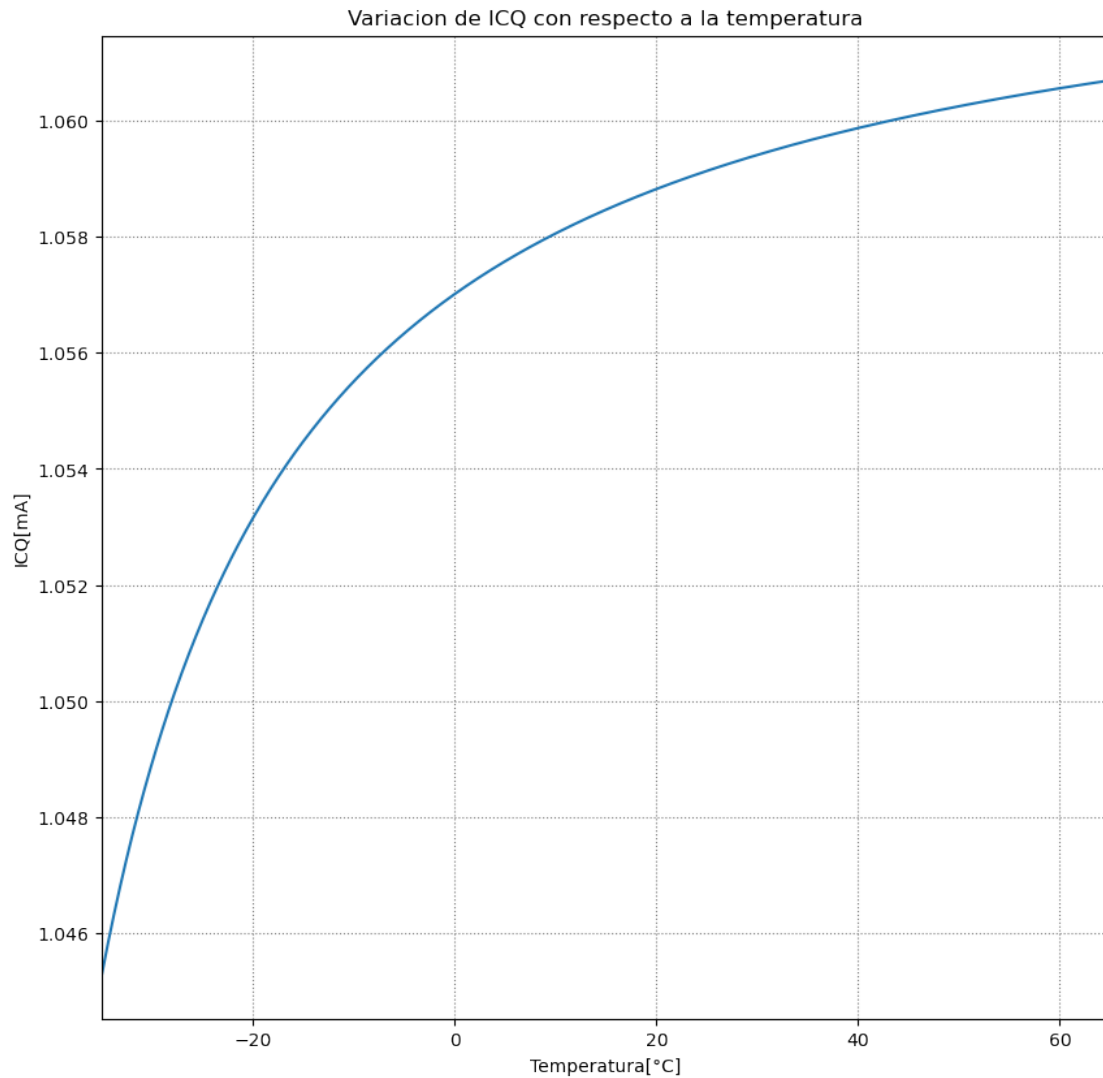
```
[10]: #%% hFE
plt.figure(3, figsize=(10, 10), dpi=100)
plt.plot(T, hFE)
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion del hFE con respecto a la temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('hFE')
```

[10]: Text(0, 0.5, 'hFE')



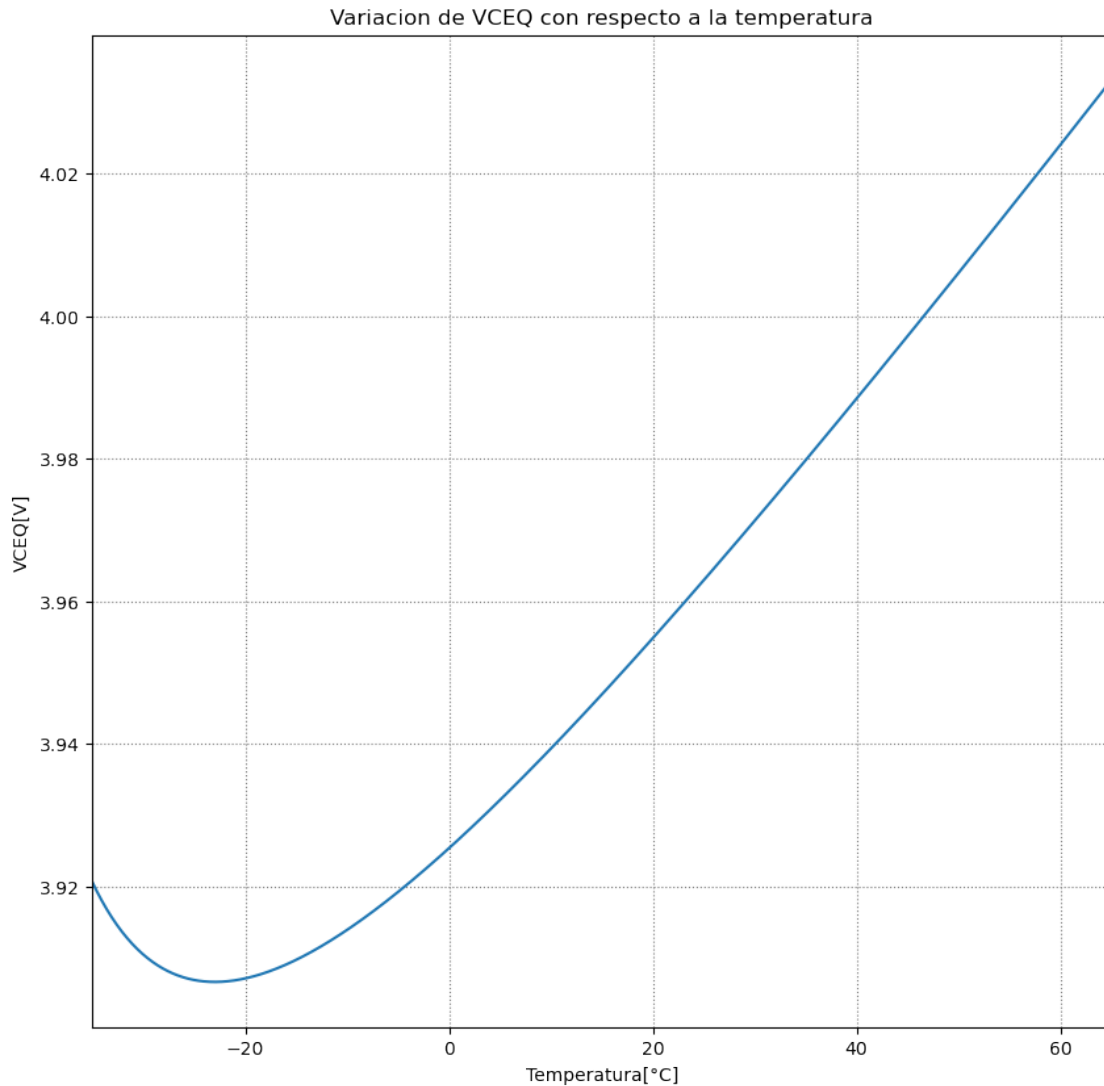
```
[11]: ICQ
      plt.figure(4, figsize=(10, 10), dpi=100)
      plt.plot(T, ICQ/1e-3) #mA
      plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
      plt.xlim(-35, 65)
      plt.title('Variacion de ICQ con respecto a la temperatura')
      plt.xlabel('Temperatura[°C]')
      plt.ylabel('ICQ[mA]')
```

```
[11]: Text(0, 0.5, 'ICQ[mA]')
```



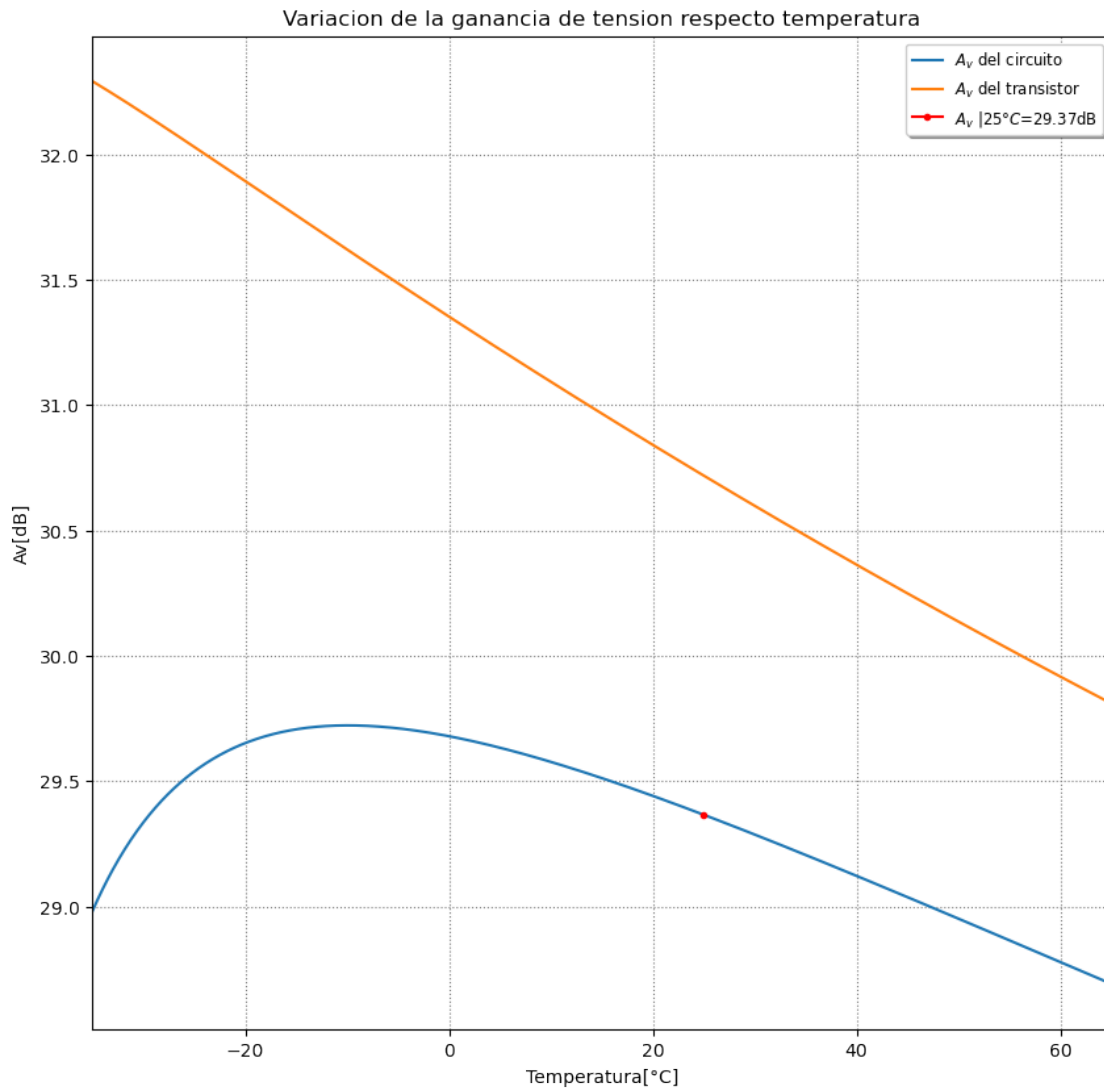
```
[12]: ### VCEQ
plt.figure(5, figsize=(10, 10), dpi=100)
plt.plot(T, VCEQ)
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de VCEQ con respecto a la temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('VCEQ[V]')
```

```
[12]: Text(0, 0.5, 'VCEQ[V]')
```

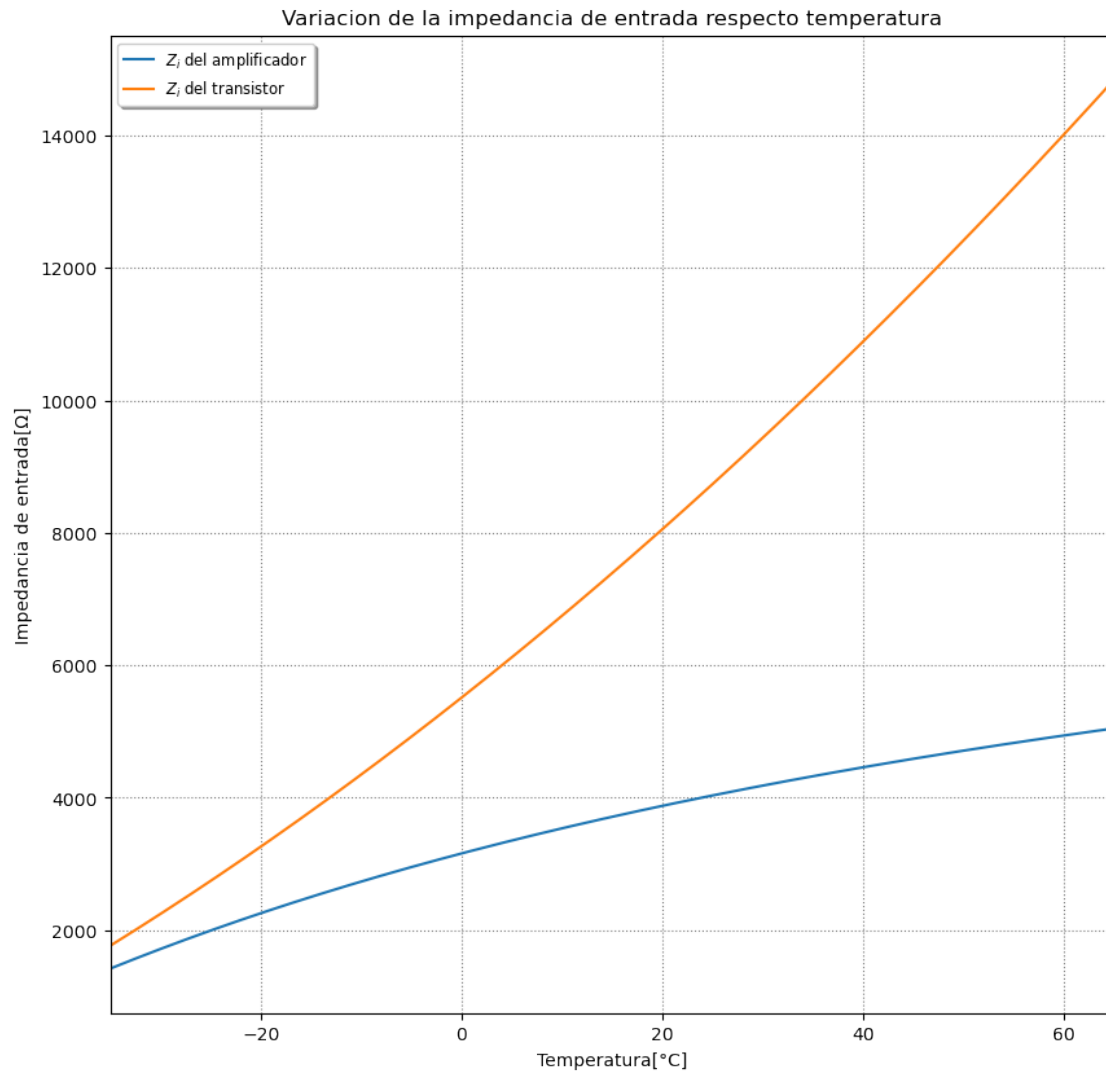
```
[13]: ### Ganancias de tension
fig = plt.figure(6, figsize=(10, 10), dpi=100)
ax = fig.add_subplot(111)
plt.plot(T, 20*np.log10(abs(AVS)), label='$A_v$ del circuito')
plt.plot(T, 20*np.log10(abs(AV)), label='$A_v$ del transistor')
plt.plot(T[i], 20*np.log10(abs(AVS[i])), color='red', marker=".", label=f'$A_v$ del transistor a 25°C = {round(20*np.log10(abs(AVS[i])), 2)}dB')
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de la ganancia de tension respecto temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('Av[dB]')
plt.legend(loc='upper right', shadow=True, fontsize='small')
```

[13]: <matplotlib.legend.Legend at 0x20e3f710df0>



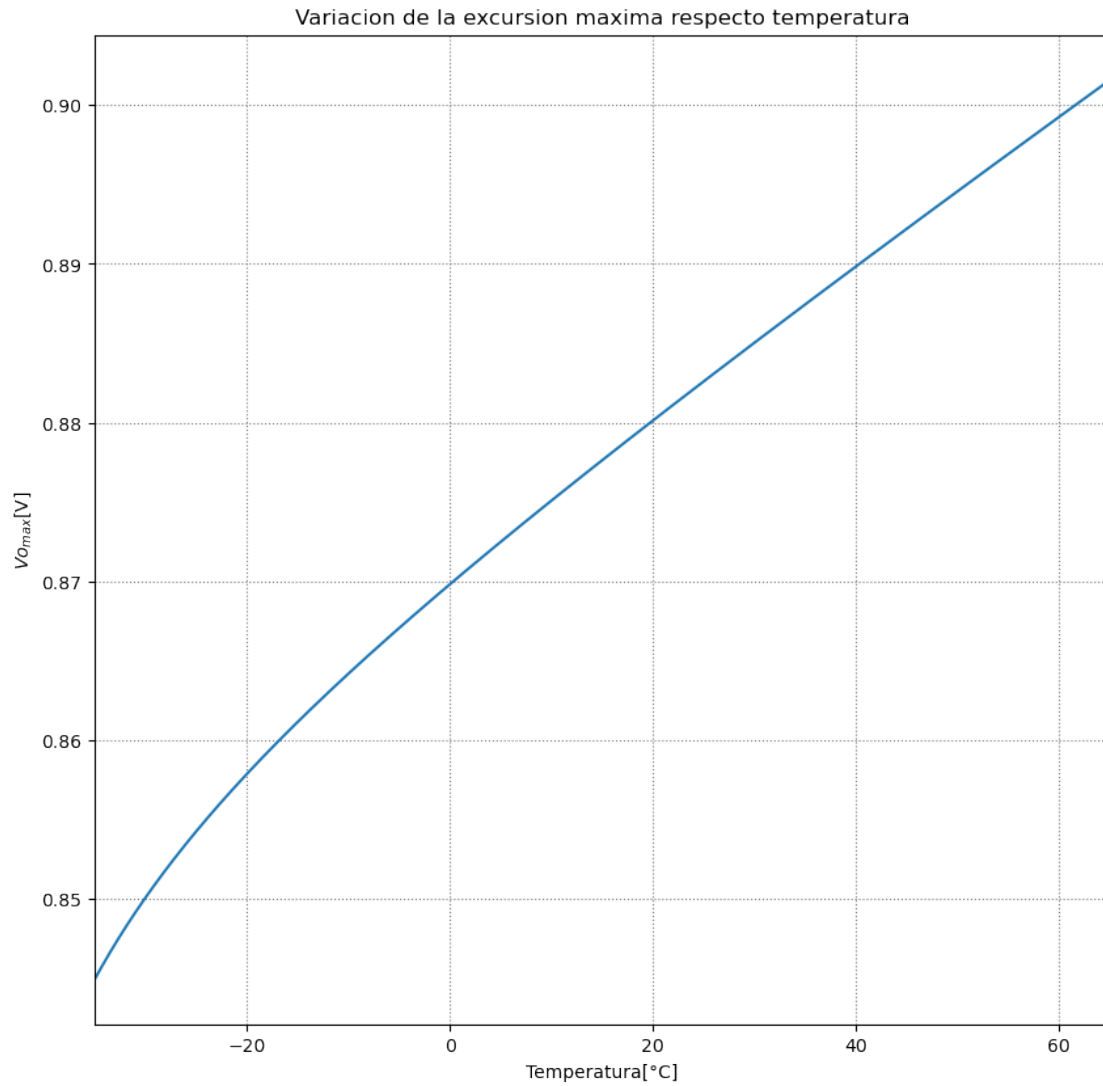
```
[14]: Impedancias de entrada
plt.figure(7, figsize=(10, 10), dpi=100)
plt.plot(T, RIA, label='$Z_i$ del amplificador')
plt.plot(T, RI, label='$Z_i$ del transistor')
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de la impedancia de entrada respecto temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('Impedancia de entrada[Ω]')
plt.legend(loc='upper left', shadow=True, fontsize='small')
```

[14]: <matplotlib.legend.Legend at 0x20e3f80cfa0>



```
[15]: ### Tension de salida
plt.figure(8, figsize=(10, 10), dpi=100)
plt.plot(T, VCE_MAX)
plt.grid(color='gray', linestyle='dotted', linewidth=0.8)
plt.xlim(-35, 65)
plt.title('Variacion de la excursion maxima respecto temperatura')
plt.xlabel('Temperatura[°C]')
plt.ylabel('$V_{o\_max}$[V]')

plt.show()
```

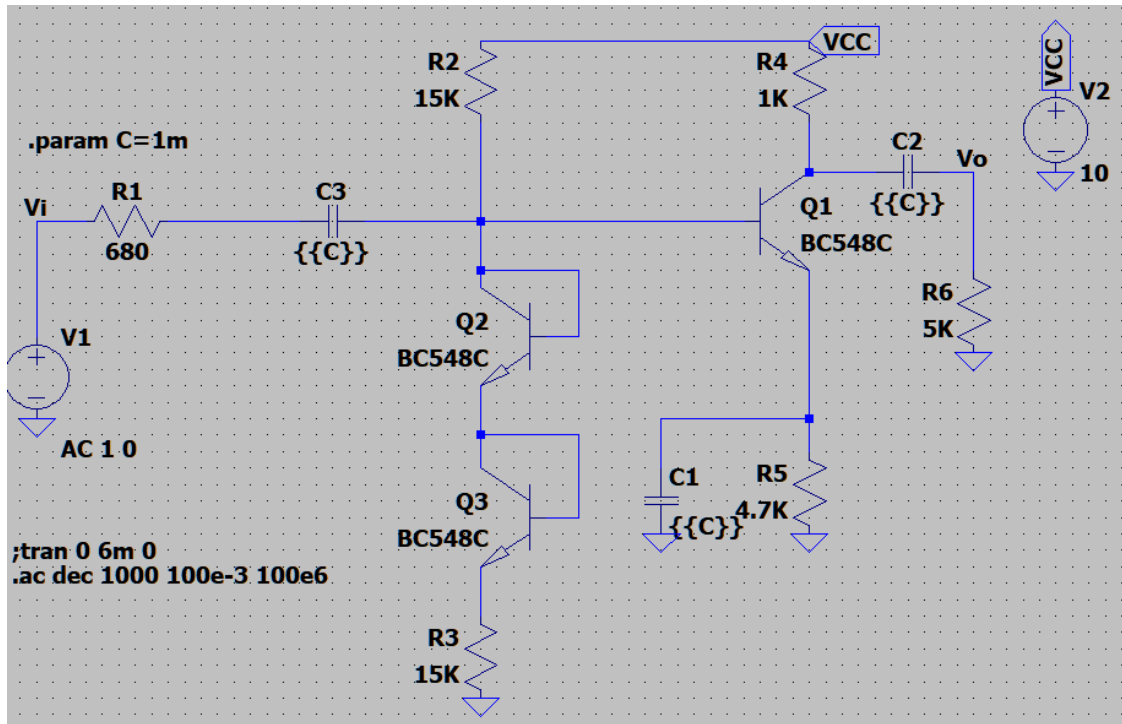


1.2 2) Simulacion en LTSpice

1.2.1 Circuito en LTSpice:

```
[16]: fig2 = Image(filename='./circuit_LTSpice.png')  
fig2
```

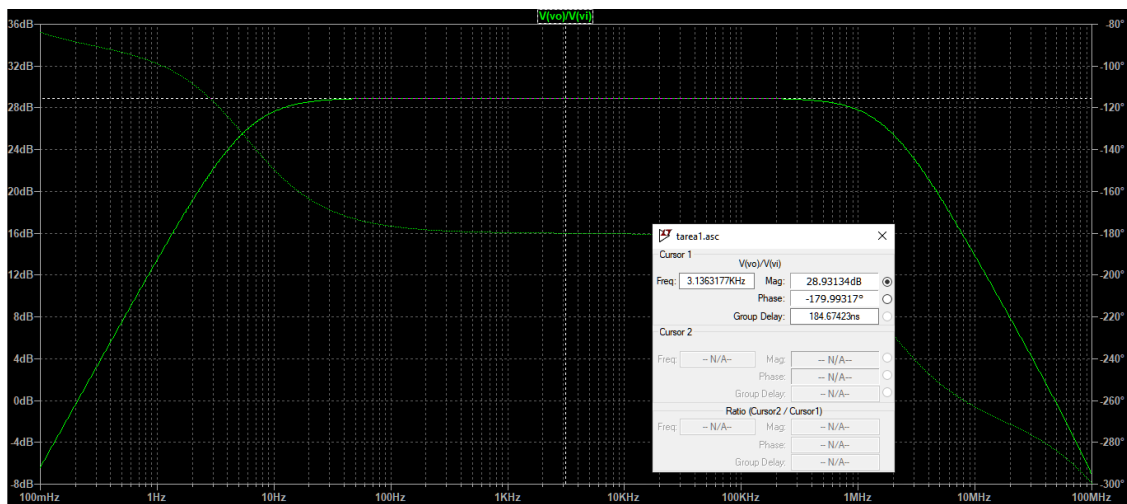
[16]:



1.2.2 Ganancia AVS del circuito:

```
[17]: fig = Image(filename=('./AV_LTSpice.png'))
fig
```

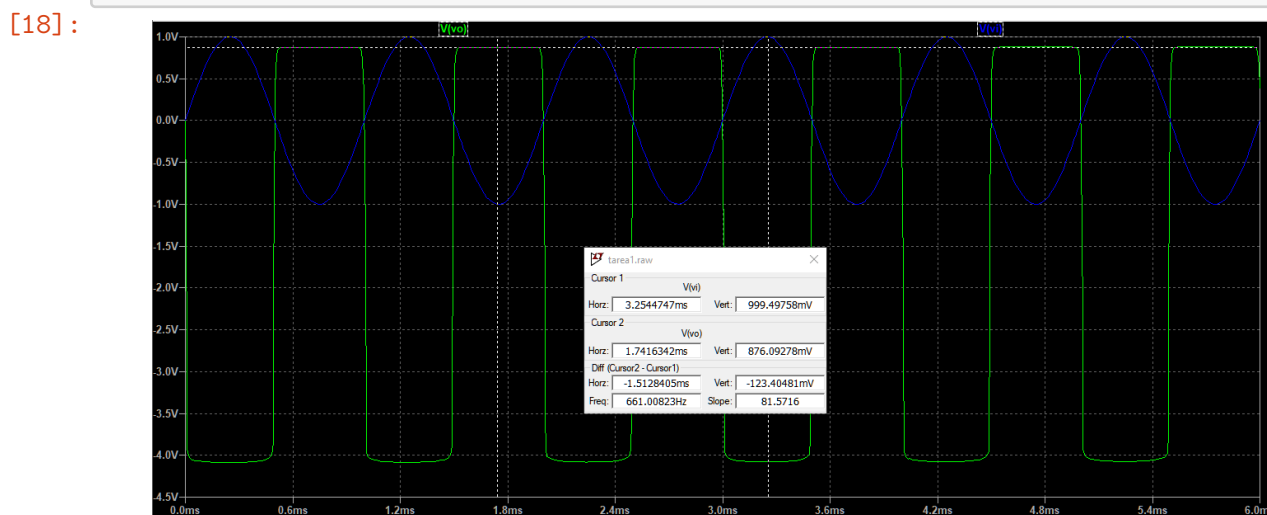
[17]:



1.3 3) Conclusiones

Como podemos observar hay una diferencia entre la ganancia A_{vs} calculada por el script en Python y el simulado en LTSpice, pero esa diferencia es mínima, por lo que puedo considerar que son similares. Luego como vimos en la teoría este circuito esta polarizado de forma tal que las variaciones frente a los cambios de temperatura sean mínimos, si observamos la grafica de IC_Q vs $Temperatura$ podemos ver que IC_Q varia muy poco, lo mismo sucede con respecto a VCE_Q , esto nos quiere decir que el punto de operación del circuito se mantendrá estable frente a los cambios de temperatura, validando de esta manera lo visto en teoría. Por otro lado, en el inciso 2) realice un análisis frecuencial para verificar la ganancia de tensión, pero si hacemos un análisis temporal podemos verificar que la excursión máxima para 25 grados Celsius también se verifica obteniendo un valor aproximado a 880mV para una entrada de 1V máximo, además también verificamos que la deformación se presenta primero en el semiciclo positivo lo que nos dice que la excursión máxima esta dada para que el transistor no entre en la region de corte, es decir por $IC_Q.R_D$.

```
[18]: fig = Image(filename=('./Vo_LTSpice.png'))
fig
```



Puede visitar este notebook de forma online en el siguiente enlace: [Tarea 1 - Electrónica Aplicada 1 - Alexander Daniel Rios](#)