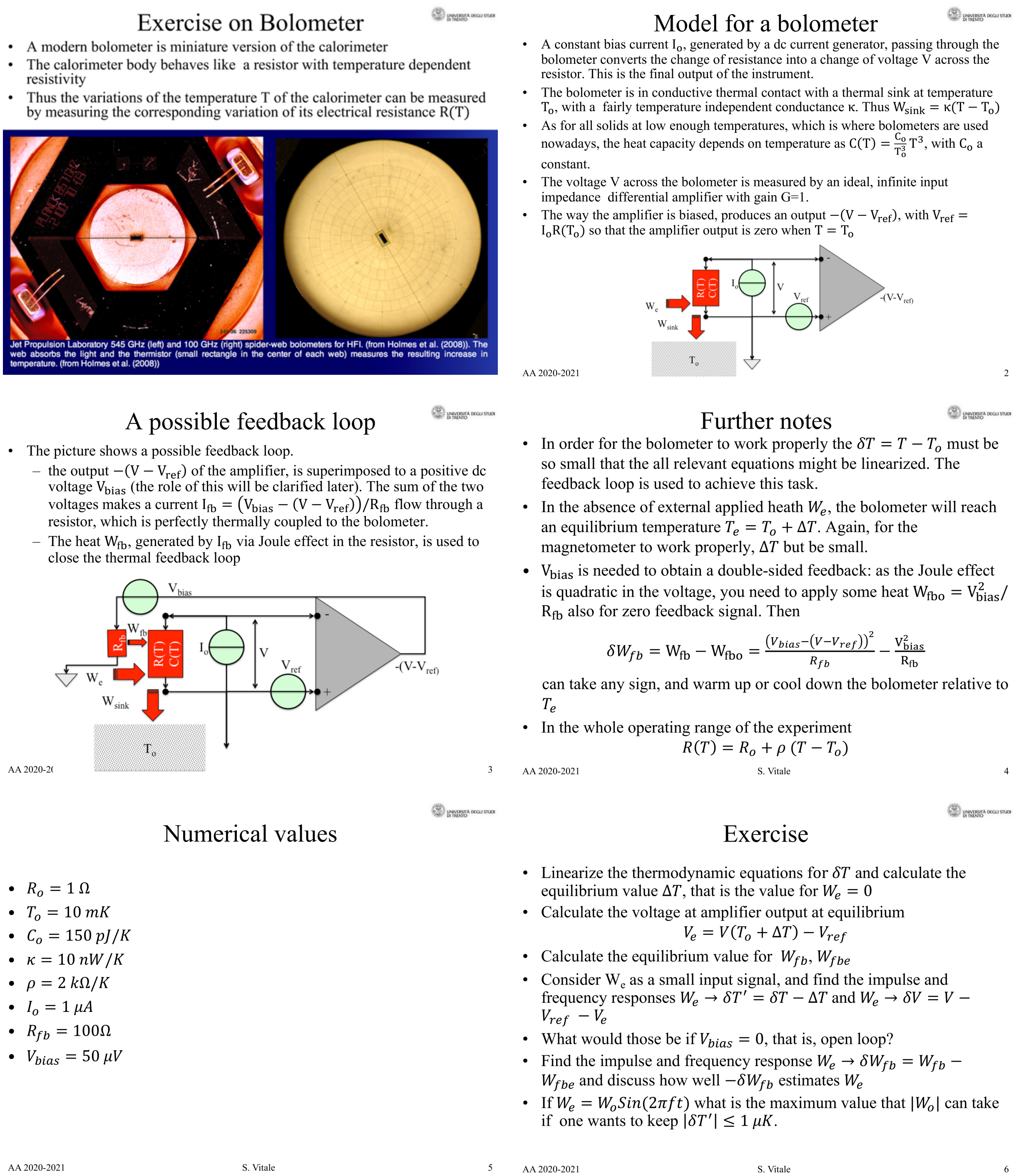
Exercise 05: Bolometer

- Gugliemo Grillo



Exercise on Bolometer

- A modern bolometer is miniature version of the calorimeter
- The calorimeter body behaves like a resistor with temperature dependent resistivity
- Thus the variations of the temperature T of the calorimeter can be measured by measuring the corresponding variation of its electrical resistance R(T)

Jet Propulsion Laboratory 545 GHz (left) and 100 GHz (right) spider-web bolometers for HFI. (from Holmes et al. (2008)). The web absorbs the light and the thermistor (small rectangle in the center of each web) measures the resulting increase in temperature. (from Holmes et al. (2008))

Model for a bolometer

- A constant bias current I_o, generated by a dc current generator, passing through the bolometer converts the change of resistance into a change of voltage V across the resistor. This is the final output of the instrument.
- The bolometer is in conductive thermal contact with a thermal sink at temperature T_o, with a fairly temperature independent conductance κ. Thus $W_{sink} = \kappa(T - T_o)$
- As for all solids at low enough temperatures, which is where bolometers are used nowadays, the heat capacity depends on temperature as $C(T) = \frac{C_o}{T_o^3}T^3$, with C_o a constant.
- The voltage V across the bolometer is measured by an ideal, infinite input impedance differential amplifier with gain G=1.
- The way the amplifier is biased, produces an output $-(V - V_{ref})$, with $V_{ref} = I_o R(T_o)$ so that the amplifier output is zero when $T = T_o$

AA 2020-2021 2

A possible feedback loop

- The picture shows a possible feedback loop.
  - the output $-(V - V_{ref})$ of the amplifier, is superimposed to a positive dc voltage V_{bias} (the role of this will be clarified later). The sum of the two voltages makes a current $I_{fb} = \left(V_{bias} - (V - V_{ref})\right)/R_{fb}$ flow through a resistor, which is perfectly thermally coupled to the bolometer.
  - The heat W_{fb}, generated by I_{fb} via Joule effect in the resistor, is used to close the thermal feedback loop

AA 2020-20 3

Further notes

- In order for the bolometer to work properly the $\delta T = T - T_o$ must be so small that the all relevant equations might be linearized. The feedback loop is used to achieve this task.
- In the absence of external applied heath W_e, the bolometer will reach an equilibrium temperature $T_e = T_o + \Delta T$. Again, for the magnetometer to work properly, ΔT but be small.
- V_{bias} is needed to obtain a double-sided feedback: as the Joule effect is quadratic in the voltage, you need to apply some heat $W_{fbo} = V_{bias}^2/R_{fb}$ also for zero feedback signal. Then

$$\delta W_{fb} = W_{fb} - W_{fbo} = \frac{\left(V_{bias} - (V - V_{ref})\right)^2}{R_{fb}} - \frac{V_{bias}^2}{R_{fb}}$$

can take any sign, and warm up or cool down the bolometer relative to T_e
- In the whole operating range of the experiment

$$R(T) = R_o + \rho\,(T - T_o)$$

AA 2020-2021 S. Vitale 4

Numerical values

- $R_o = 1\,\Omega$
- $T_o = 10\,mK$
- $C_o = 150\,pJ/K$
- $\kappa = 10\,nW/K$
- $\rho = 2\,k\Omega/K$
- $I_o = 1\,\mu A$
- $R_{fb} = 100\Omega$
- $V_{bias} = 50\,\mu V$

AA 2020-2021 S. Vitale 5

Exercise

- Linearize the thermodynamic equations for δT and calculate the equilibrium value ΔT, that is the value for $W_e = 0$
- Calculate the voltage at amplifier output at equilibrium

$$V_e = V(T_o + \Delta T) - V_{ref}$$
- Calculate the equilibrium value for W_{fb}, W_{fbe}
- Consider W_e as a small input signal, and find the impulse and frequency responses $W_e \to \delta T' = \delta T - \Delta T$ and $W_e \to \delta V = V - V_{ref} - V_e$
- What would those be if $V_{bias} = 0$, that is, open loop?
- Find the impulse and frequency response $W_e \to \delta W_{fb} = W_{fb} - W_{fbe}$ and discuss how well $-\delta W_{fb}$ estimates W_e
- If $W_e = W_o Sin(2\pi f t)$ what is the maximum value that $|W_o|$ can take if one wants to keep $|\delta T'| \leq 1\,\mu K$.

AA 2020-2021 S. Vitale 6

In [1]:
```
# packages used
import scipy.constants
from scipy import signal
from scipy.integrate import quad

import numpy as np

import matplotlib.pyplot as plt
```

```python
import seaborn as sns

from IPython.display import display, Math

# Given constans
R0 = 1 #Ohm
T0 = 10e-3 #K
C0 = 150e-12 # J/K
KAPPA = 10e-9 # W/K
RHO = 2e3 #Ohm/K
I0 = 1e-6 #A
Rfb = 100 #Ohm
Vbias = 50e-6 #V
```

Equilibrium Value

Let's start by writing the thermodynamical equation for the system both in terms of heat and voltages:

$$C_0 \frac{T^3}{T_0^3} \frac{d\delta T}{dt} = W_e - W_{sink} + W_{fb}$$

$$C_o \frac{(T + \delta T)^3}{T_0} \frac{d\delta T}{dt} = W_e - k\delta T + \frac{[V_{bias} - (V - V_{ref})]^2}{R_{fb}}$$

The second one, by substituting the relation $V(T) = I_0\rho(T - T_0) + I_0 R(T_0)$ and neglecting terms of order higher than linear, becomes:

$$C_o\left(3\frac{\delta T}{T_0} + 1\right)\frac{d\delta T}{dt} + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T = W_e + \frac{V_{bias}^2}{R_{fb}}$$

It's then possible to evaluate ΔT by imposing $W_e = 0$ and noting that at equilibrium $\frac{d\delta T}{dt} = 0$. Thus we have:

$$\Delta T = \frac{V_{bias}^2}{R_{fb}}\left[k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right]^{-1}$$

In [2]:
```python
DeltaT = Vbias**2/Rfb / (KAPPA+2*Vbias*RHO*I0/Rfb)
display(Math(r'T_e-T_0=\Delta T = {:.5e} K'.format(DeltaT)))
```

$$T_e - T_0 = \Delta T = 2.08333e - 03K$$

Amplifier's voltage output at equilibrium

The quantity desired is obtained through:

$$V_e = V(T_0 + \Delta T) - V_{ref} = \rho(T_0 + \Delta T)I_0 - \rho T_0 I_0 = \rho I_0 \Delta T$$

In [3]:
```python
Ve = RHO*I0*(DeltaT)
display(Math(r'V_e = {:.5e} V'.format(Ve)))
```

$$V_e = 4.16667e - 06V$$

Feedback heat at equilibrium

The heat generated through the feedback loop can be written as:

$$W_{fbe} = \frac{[V_{bias} - (V - V_{ref})]^2}{R_{fb}} = \frac{V_{bias}^2 - 2\rho\Delta T I_0 V_{bias} + (\rho I_0 \Delta T)^2}{R_{fb}} \simeq \frac{V_{bias}^2 - 2\rho\Delta T I_0 V_{bias}}{R_{fb}}$$

where we discarded the term depending on ΔT^2.

In [4]:
```
Wfbe = (Vbias**2 - 2*RHO*DeltaT*I0*Vbias)/Rfb
display(Math(r'W_{fbe} ='+' {:.5e} V'.format(Wfbe)))
```

$$W_{fbe} = 2.08333e - 11V$$

Impulse and frequency response

In order to evaluate both the impulse and the frequency response we assume the term $\frac{\delta T}{T_0} << 1$ and simplify the differential equation to:

$$C_o \frac{d\delta T(t)}{dt} + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T(t) = W_e(t) + \frac{V_{bias}^2}{R_{fb}}$$

Case 1: $W_e \rightarrow \delta T' = \delta T - \Delta T$

If we perform the desired change of variable the equation becomes:

$$C_o \frac{d\delta T'(t)}{dt'} + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T'(t) + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\Delta T = W_e(t) + \frac{V_{bias}^2}{R_{fb}}$$

$$\Rightarrow C_o \frac{d\delta T'(t)}{dt} + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T'(t) = W_e(t)$$

where we make use of the relation (4). Performing a Fourier transform we get:

$$C_o i\omega \delta T'(\omega) + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T'(\omega) = W_e(\omega)$$

$$[\text{Frequency response}]\ h(\omega) = \frac{\delta T'(\omega)}{W_e(\omega)} = \frac{1/C_0}{i\omega + \frac{k}{C_0} + 2\frac{V_{bias}}{R_{fb}C_0}}$$

$$[\text{Impulse response}]\ h(t) = \frac{\delta T'(t)}{W_e(t)} = \frac{1}{2\pi C_0} e^{-\left(\frac{k}{C_0} + 2\frac{V_{bias}}{R_{fb}C_0}\right)t}\theta(t)$$

Case 2: $W_e \rightarrow \delta V = V - V_{ref} - V_e$

We could substitute and redo all the calculation. Instead we note that:

$$\delta V(t) = I_0\rho(\delta T'(t) + \Delta T) - I_0\rho(T_0 + \Delta T - T_0) = I_0\rho\ \delta T'(t)$$

$$\rightarrow \delta V(\omega) = I_0\rho\ \delta T'(\omega)$$

We can then perform the substitution in the Fourier space and get:

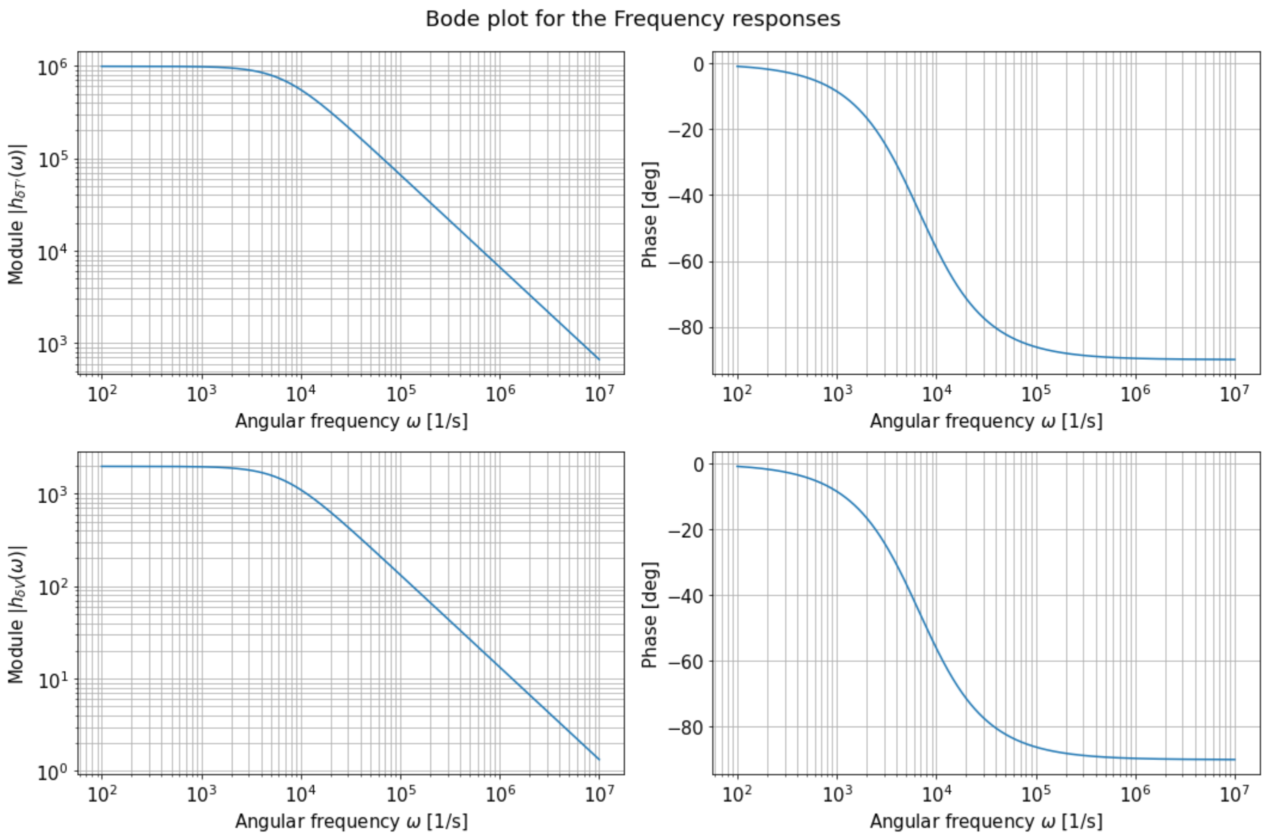
$$\text{[Frequency response]}\ h(\omega) = \frac{\delta V(\omega)}{W_e(\omega)} = \frac{I_0\rho/C_0}{i\omega + \frac{k}{C_0} + 2\frac{V_{bias}}{R_{fb}C_0}}$$

$$\text{[Impulse response]}\ h(t) = \frac{\delta V(t)}{W_e(t)} = \frac{I_0\rho}{2\pi C_0} e^{-\left(\frac{k}{C_0}+2\frac{V_{bias}}{R_{fb}C_0}\right)t}\theta(t)$$

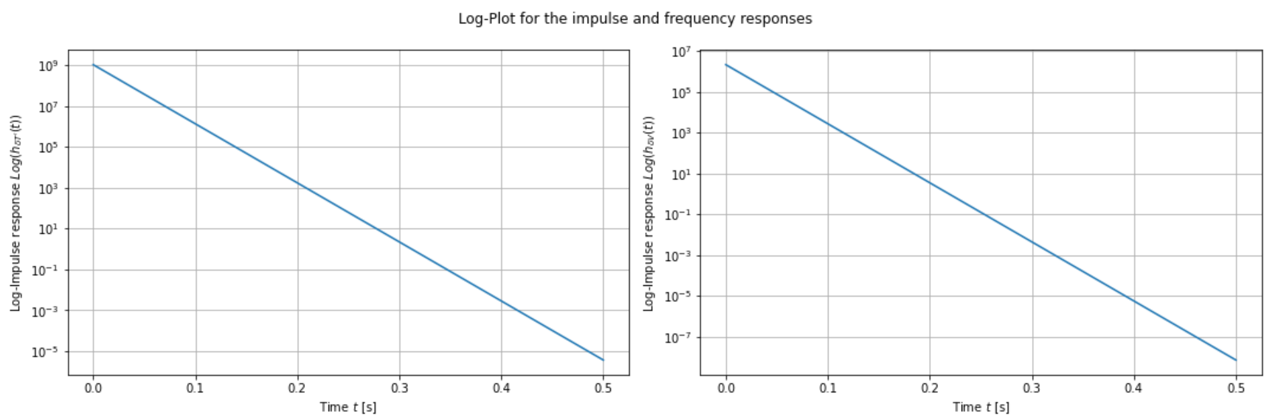
The plots for the impulse and frequency response are:

In [5]:
```
def h_T(omega):
    return 1/(1j*omega*C0 + KAPPA + 2*Vbias/Rfb)

def h_V(omega):
    return I0*RHO/(1j*omega*C0 + KAPPA + 2*Vbias/Rfb)

w = np.linspace(int(1e2), int(1e7), int(1e6))

points_h_T = h_T(w)
mag = np.abs(points_h_T)
phase = np.arctan(np.imag(points_h_T) / np.real(points_h_T))*180/np.pi

# Bode plot
plt.rcParams.update({'font.size': 15})
fig, axis = plt.subplots(2,2, figsize=(15, 10))
fig.suptitle('Bode plot for the Frequency responses')

axis[0][0].plot(w, mag)
axis[0][0].set_xscale('log')
axis[0][0].set_yscale('log')
axis[0][0].set_xlabel("Angular frequency $\omega$ [1/s]")
axis[0][0].set_ylabel("Module |$h_{\delta T'}(\omega$)|")
axis[0][0].grid(True, which="both")

axis[0][1].plot(w, phase)
axis[0][1].set_xscale('log')
axis[0][1].set_xlabel(r"Angular frequency $\omega$ [1/s]")
axis[0][1].set_ylabel(r"Phase [deg]")
axis[0][1].grid(True, which="both")


points_h_V = h_V(w)
mag = np.abs(points_h_V)
phase = np.arctan(np.imag(points_h_V) / np.real(points_h_V))*180/np.pi

axis[1][0].plot(w, mag)
axis[1][0].set_xscale('log')
axis[1][0].set_yscale('log')
axis[1][0].set_xlabel(r"Angular frequency $\omega$ [1/s]")
axis[1][0].set_ylabel(r"Module |$h_{\delta V}(\omega$)|")
axis[1][0].grid(True, which="both")

axis[1][1].plot(w, phase)
axis[1][1].set_xscale('log')
axis[1][1].set_xlabel(r"Angular frequency $\omega$ [1/s]")
axis[1][1].set_ylabel(r"Phase [deg]")
axis[1][1].grid(True, which="both")
```

```
plt.tight_layout()
plt.show()
```



In [6]:
```
def h_T(t):
    return np.exp(-(KAPPA/C0 + 2*Vbias/Rfb)*t)/(2*np.pi*C0)

def h_V(t):
    return I0*RHO/(2*np.pi*C0)*np.exp(-(KAPPA/C0 + 2*Vbias/Rfb)*t)

t = np.linspace(0, 0.5, int(1e6))

# Bode plot
plt.rcParams.update({'font.size': 10})
fig, axis = plt.subplots(1, 2, figsize=(15, 5))
fig.suptitle('Log-Plot for the impulse and frequency responses')

axis[0].plot(t, h_T(t=t))
axis[0].set_yscale('log')
axis[0].set_xlabel("Time $t$ [s]")
axis[0].set_ylabel("Log-Impulse response $Log(h_{\delta T'}(t))$")
axis[0].grid(True, which="both")


axis[1].plot(t, h_V(t=t))
axis[1].set_yscale('log')
axis[1].set_xlabel("Time $t$ [s]")
axis[1].set_ylabel("Log-Impulse response $Log(h_{\delta V}(t))$")
axis[1].grid(True, which="both")


plt.tight_layout()
plt.show()
```



Null V_{bias}

If $V_{bias} = 0$ the answer follow from the previous answers:

Case 1: $W_e \rightarrow \delta T' = \delta T - \Delta T$

$$h(\omega) = \frac{\delta T'(\omega)}{W_e(\omega)} = \frac{1/C_0}{i\omega + \frac{k}{C_0}}$$

$$h(t) = \frac{\delta T'(t)}{W_e(t)} = \frac{1}{2\pi C_0} e^{-\frac{k}{C_0}t}\theta(t)$$

Case 2: $W_e \rightarrow \delta V = V - V_{ref} - V_e$

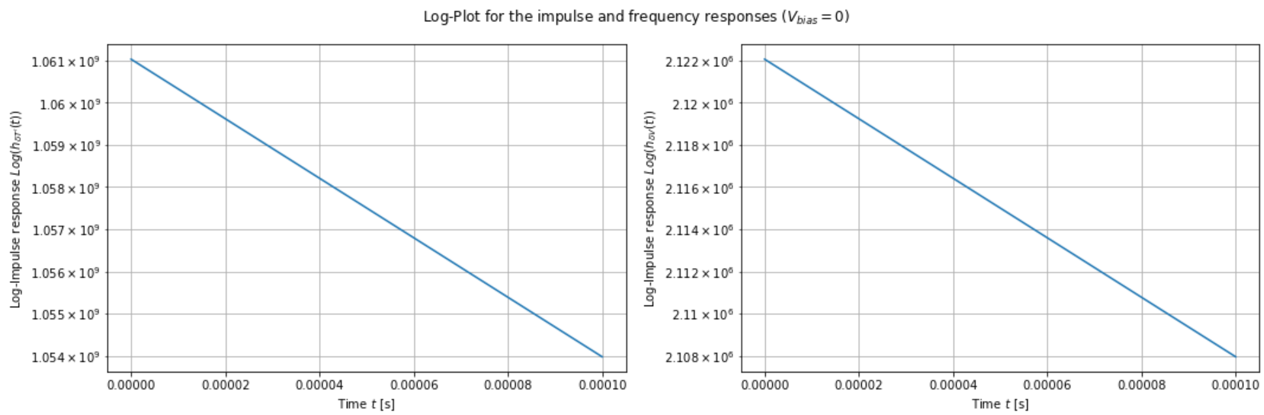
$$h(\omega) = \frac{\delta V(\omega)}{W_e(\omega)} = \frac{I_0\rho/C_0}{i\omega + \frac{k}{C_0}}$$

$$h(t) = \frac{\delta V(t)}{W_e(t)} = \frac{I_0\rho}{2\pi C_0} e^{-\frac{k}{C_0}t}\theta(t)$$

In [7]:

```
def h_T(t):
    return np.exp(-(KAPPA/C0)*t)/(2*np.pi*C0)

def h_V(t):
    return I0*RHO/(2*np.pi*C0)*np.exp(-(KAPPA/C0)*t)

t = np.linspace(0, 1e-4, int(1e6))

# Bode plot
plt.rcParams.update({'font.size': 10})
fig, axis = plt.subplots(1, 2, figsize=(15, 5))
fig.suptitle('Log-Plot for the impulse and frequency responses ($V_{bias}=0$)')

axis[0].plot(t, h_T(t=t))
axis[0].set_yscale('log')
axis[0].set_xlabel("Time $t$ [s]")
axis[0].set_ylabel("Log-Impulse response $Log(h_{\delta T'}(t))$")
axis[0].grid(True, which="both")
```

```
axis[1].plot(t, h_V(t=t))
axis[1].set_yscale('log')
axis[1].set_xlabel("Time $t$ [s]")
axis[1].set_ylabel("Log-Impulse response $Log(h_{\delta V}(t))$")
axis[1].grid(True, which="both")


plt.tight_layout()
plt.show()
```



Case: $W_e \to \delta W_{fb} = W_{fb} - W_{fbe}$

To start, let's write the relation between δT and δW_{fb}:

$$\delta W_{fb} = W_{fb} - W_{fbe} = \frac{[V_{bias} - (V - V_{ref})]^2}{R_{fb}} - \frac{V_{bias}^2 - 2\rho \Delta T I_o V_{bias} + (\rho I_0 \Delta T)}{R_{fb}} = \frac{-2V_{bias}\rho I_0(\delta}{}$$

$$\delta W_{fb} = \frac{-2V_{bias}\rho I_0}{R_{fb}}\delta T + \frac{2V_{bias}\rho I_0}{R_{fb}}\Delta T + \mathcal{O}\big[(\delta T^2 - \Delta T^2)\big]$$

$$\to \delta T = -\frac{R_{fb}}{2\rho V_{bias} I_0}\delta W_{fb} + \Delta T$$

Where in (22) the $\mathcal{O}\big[(\delta T^2 - \Delta T^2)\big]$ term was neglected. This is done because $\mathcal{O}\big[(\delta T^2 - \Delta T^2)\big] = \mathcal{O}\big[(T_0^2 - T_e^2 + 2T\,\Delta T)\big]$ is negligible. An esteem for the error at equilibrium is provided before the plot.

The differential equation then becomes:

$$-\frac{C_0 R_{fb}}{2V_{bias}\rho I_0}\frac{d\,\delta W_{fb}(t)}{dt} - \big(k + 2\frac{V_{bias}}{R_{fb}\rho I_0}\big)\frac{R_{fb}}{2V_{bias}\rho I_0}\,\delta W_{fb} = W_e(t) + \big[\frac{V_{bias}^2}{R_{fb}} - \big(k + 2\frac{V_{bias}}{R_{fb}\rho I_0}\big)\,\Delta T$$

Where the term between square brackets on the right is zero due to (4). The equation can be rewritten as:

$$\frac{d\,\delta W_{fb}(t)}{dt} + \big(\frac{k}{C_0} + 2\frac{V_{bias}\rho I_0}{C_0 R_{fb}}\big)\,\delta W_{fb}(t) = -\frac{2V_{bias}\rho I_0}{C_0 R_{fb}}W_e(t)$$

Defining $\tau = \frac{2V_{bias}\rho I_0}{C_0 R_{fb}}$ and $\delta W_{fb} \to -\delta W_{fb}$:

$$\frac{d\,\delta W_{fb}(t)}{dt} + \left(\frac{k}{C_0} + \tau\right)\,\delta W_{fb}(t) = \tau W_e(t)$$

$$i\omega\,\delta W_{fb}(\omega) + \left(\frac{k}{C_0} + \tau\right)\,\delta W_{fb}(\omega) = \tau W_e(\omega)$$

The frequency and impulse responses are then:

$$h_{\delta W_{fb}}(\omega) = \frac{W_{fb}(\omega)}{W_e(\omega)} = \frac{\tau}{i\omega + \left(\frac{k}{C_0} + \tau\right)}$$

$$h_{\delta W_{fb}}(t) = \frac{W_{fb}(t)}{W_e(t)} = \frac{\tau}{2\pi}\theta(t)e^{-\left(\frac{k}{C_0}+\tau\right)t}$$

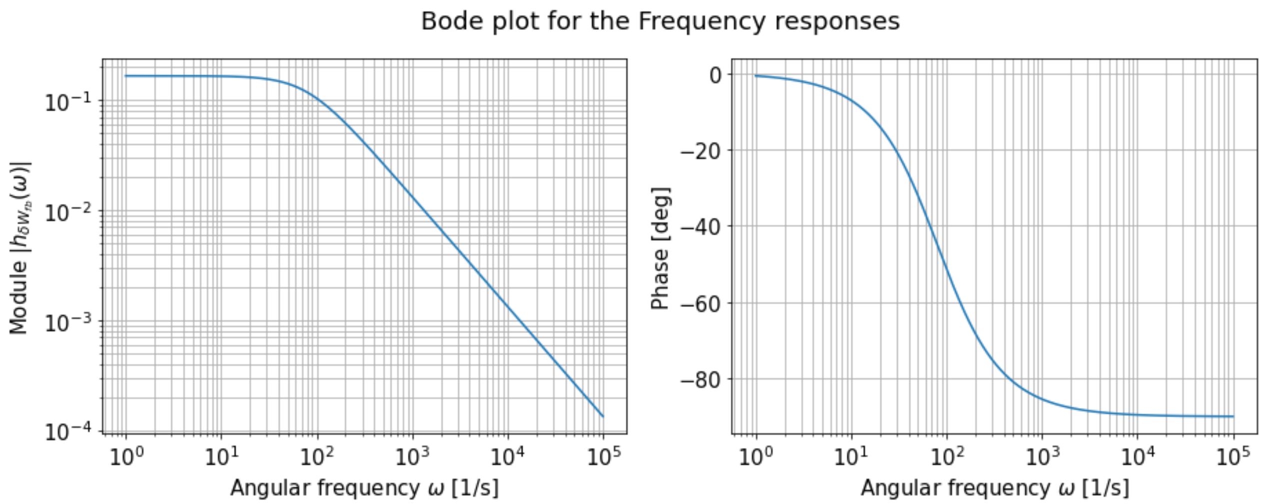
In [33]:

```
Te = T0 + DeltaT
error = lambda T: (RHO**2 * I0**2 /Rfb)*(T0**2 - Te**2 + 2*T*DeltaT)

display(Math(r'error({:.5e})\simeq{:.5e} W'.format(Te, error(Te))))
```

$$error(1.20833e-02) \simeq 1.73611e-13W$$

In [9]:

```
tau = 2*Vbias*RHO*I0 /(C0*Rfb)

def h_W(omega):
    return -1*tau / (1j*omega+(KAPPA/C0 + tau) )

w = np.linspace(int(1e0), int(1e5), int(1e6))

points = h_W(w)
mag = np.abs(points)
phase = np.arctan(np.imag(points) / np.real(points))*180/np.pi

# Bode plot
plt.rcParams.update({'font.size': 15})
fig, axis = plt.subplots(1,2, figsize=(15, 5))
fig.suptitle('Bode plot for the Frequency responses')

axis[0].plot(w, mag)
axis[0].set_xscale('log')
axis[0].set_yscale('log')
axis[0].set_xlabel("Angular frequency $\omega$ [1/s]")
axis[0].set_ylabel("Module |$h_{\delta W_{fb}}(\omega$)|")
axis[0].grid(True, which="both")

axis[1].plot(w, phase)
axis[1].set_xscale('log')
axis[1].set_xlabel(r"Angular frequency $\omega$ [1/s]")
axis[1].set_ylabel(r"Phase [deg]")
axis[1].grid(True, which="both")
```



Maximum value for $|W_0|$

In order to answer to the last problem, let's rewrite equation (8) with the given values:

$$C_o\frac{d\delta T'(t)}{dt} + \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T'(t) = W_0 sin(2\pi f t)$$

Now we perform some simplification assuming that:

```
\begin{enumerate}
   \item We are only interested in the behaviour near the maxima
   \item The relation between $\delta T'(t)$ and $W_e(t)$ is linear (this can be assumed because we ar
\end{enumerate}
```

With these hypothesis the maxima of $W_e(t)$ correspond to the maxima of $\delta T'(t)$ and it's derivative with respect to the time is zero. The expression can then be simplified to:

$$\left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)\delta T'(t) = W_0 \rightarrow \delta T'(t) = W_0\left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right)^{-1} \leq 1\mu K$$

And the maximum value for W_0 is:

$$W_0 = \left(k + 2\frac{V_{bias}}{R_{fb}}\rho I_o\right) * 1\mu K$$

In [10]:

```
W0 = KAPPA + 2* Vbias*RHO*I0 / Rfb
display(Math(r'|W_{0}| ='+' {:.5e} W'.format(W0)))
```

$$|W_0| = 1.20000e - 08W$$