Superconducting transmission line resonators for quantum information processing

Gleb Fedorov
Methods of experimental physics    April 17, 2020
1 Introduction
- Superconducting qubits and circuit QED
- Physical quantum oscillators
- LC-oscillator
- Theory of transmission lines
- Transmission line resonators (TLR)
- Equivalent parameters of TLR
- S-parameters
- Some numerical examples
- More complex features of real devices

2 Designing TLR
- How the samples are made
- EDA software
- Simulating your design

3 Measuring TLR
- Sample connection
- Lab equipment
- Measurement process
- Data processing in Python
Superconducting qubits and circuit QED

Introduction

What do people call qubits?

Formally, any two-state system can be used as a qubit:

$$|\psi\rangle = \alpha |a\rangle + \beta |b\rangle, \quad \alpha, \beta \in \mathbb{C},$$

$$|\alpha|^2 + |\beta|^2 = 1,$$

global phase does not matter

But how to measure one?

Depends on the system. We will talk about SC quits.
Superconducting qubits and circuit QED

Introduction

Current technique is based on cavity QED:\(^1\):

The Hamiltonian of the system:

\[ H = \hbar \omega_r \left( a^\dagger a + \frac{1}{2} \right) \]

\[ + \frac{\hbar \Omega}{2} \sigma^z + \hbar g (a^\dagger \sigma^- + a \sigma^+) \]

Dispersive limit:

\[ H \approx \hbar \left[ \omega_r + \frac{g^2}{\Delta} \sigma^z \right] a^\dagger a + \frac{\hbar}{2} \left[ \Omega + \frac{g^2}{\Delta} \right] \sigma^z. \]

\(^1\)A. Blais et al., Physical Review A 69, 062320 (2004)
In such configuration, the transmission of the Fabri-Perot cavity will depend on the qubit state:
Superconducting qubits and circuit QED

Introduction
Since 2004, many types of devices were used for the role of a cavity:

Notch (shunting) type

Lumped-element LC
Since 2004, many types of devices were used for the role of a cavity:
A common feature of all electrical resonators is that they may be represented as LC-circuits (in terms of their impedance)

Resonance angular frequency:

$$\omega_0 = \frac{1}{\sqrt{LC}} = 2\pi f_0$$

Series impedance:

$$Z_s(\omega) = jL \frac{\omega^2 - \omega_0^2}{\omega}$$

Parallel impedance:

$$Z_p(\omega) = -j \frac{1}{C} \frac{\omega}{\omega^2 - \omega_0^2}$$
LC-oscillator

Introduction

Quality factors: internal and external

\[ R^* = \frac{1 + \omega^2 C_k^2 (Z_0/2)^2}{\omega^2 C_k^2 (Z_0/2)} \]

\[ C^* = \frac{C_k}{1 + \omega^2 C_k^2 (Z_0/2)^2} \approx C_k \text{ (in our case).} \]

\[ Q_l^{-1} = Q_i^{-1} + Q_e^{-1}, \]

\[ Q_i = \omega (C + C^*) R_{in}, \]

\[ Q_e = \omega (C + C^*) R^*. \]
Theory of transmission lines

Introduction

\[ \frac{\partial}{\partial x} v(z, t) = -L \frac{\partial}{\partial t} i(z, t) - R i(z, t) \]  

\[ \frac{\partial}{\partial x} i(z, t) = -C \frac{\partial}{\partial t} v(z, t) - G v(z, t) \]

---

\(^2\)D. M. Pozar, *Microwave engineering*, (John Wiley & Sons, 2009)
Theory of transmission lines

**Introduction**

For harmonic signals \( i(z, t) = I(z)e^{i\omega t} \), \( V(z, t) = V(z)e^{i\omega t} \) follow the wave equations:

\[
\frac{\partial^2 V}{\partial t^2} - u^2 \frac{\partial^2 V}{\partial z^2} = 0 \tag{5}
\]

\[
\frac{\partial^2 I}{\partial t^2} - u^2 \frac{\partial^2 I}{\partial z^2} = 0, \tag{6}
\]

where \( u \) – the speed of light in the line, 
\[
u = \alpha + j\beta = \sqrt{(R + j\omega L)(G + j\omega C)}.
\]

Voltage across the line is 
\[
v(z, t) = V_0^+ \cos(\omega t - \beta z)e^{-\alpha z} + V_0^- \cos(\omega t + \beta z)e^{\alpha z},
\]

were \( V_0^+(-) \) are the waves travelling backward and forward along \( z \) axis.

\[
Z_0 = \sqrt{\frac{R + j\omega L}{G + j\omega C}}
\]
The impedance of a line terminated by some impedance $Z_L$ (load):

$$Z_{in} = Z_0 \frac{Z_L + jZ_0 \tan ul}{Z_0 + jZ_L \tan ul}$$
Transmission line resonators (TLR)

Introduction

Using the formula above, we can find the equivalent parameters of the resonators based on transmission lines:

Halfwave open

\[ Z_{in} = \frac{Z_0}{\alpha l + j(\Delta \omega \pi / \omega_0)} \]

Halfwave shorted

\[ Z_{in} = R + 2jL\Delta \omega \]

Quarterwave

\[ Z_{in} = \frac{1}{1/R + 2k\Delta \omega C} \]
### Equivalent parameters of TLR

**Introduction**

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Series Resonator</th>
<th>Parallel Resonator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input impedance/admittance</td>
<td>$Z_{in} = R + j\omega L - j \frac{1}{\omega C}$</td>
<td>$Y_{in} = \frac{1}{R} + j\omega C - j \frac{1}{\omega L}$</td>
</tr>
<tr>
<td></td>
<td>$\simeq R + j \frac{2RQ_0 \Delta \omega}{\omega_0}$</td>
<td>$\simeq \frac{1}{R} + j \frac{2Q_0 \Delta \omega}{R \omega_0}$</td>
</tr>
<tr>
<td>Power loss</td>
<td>$P_{loss} = \frac{1}{2}</td>
<td>I</td>
</tr>
<tr>
<td>Stored magnetic energy</td>
<td>$W_m = \frac{1}{4}</td>
<td>I</td>
</tr>
<tr>
<td>Stored electric energy</td>
<td>$W_e = \frac{1}{4}</td>
<td>I</td>
</tr>
<tr>
<td>Resonant frequency</td>
<td>$\omega_0 = \frac{1}{\sqrt{LC}}$</td>
<td>$\omega_0 = \frac{1}{\sqrt{LC}}$</td>
</tr>
<tr>
<td>Unloaded $Q$</td>
<td>$Q_0 = \frac{\omega_0 L}{R} = \frac{1}{\omega_0 RC}$</td>
<td>$Q_0 = \omega_0 RC = \frac{R}{\omega_0 L}$</td>
</tr>
<tr>
<td>External $Q$</td>
<td>$Q_e = \frac{\omega_0 L}{R_L}$</td>
<td>$Q_e = \frac{R_L}{\omega_0 L}$</td>
</tr>
</tbody>
</table>
Equivalent parameters of TLR

*Introduction*

Fundamental mode equivalent parameters:

<table>
<thead>
<tr>
<th></th>
<th>Quarter wave</th>
<th>Halfwave short</th>
<th>Halfwave open</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l$</td>
<td>$\lambda/4$</td>
<td>$\lambda/2$</td>
<td>$\lambda/2$</td>
</tr>
<tr>
<td>$\omega_0 = 2\pi\beta/\lambda$</td>
<td>$\pi\beta/2l$</td>
<td>$\pi\beta/l$</td>
<td>$\pi\beta/l$</td>
</tr>
<tr>
<td>$R_r$</td>
<td>$Z_0/\alpha l$</td>
<td>$Z_0\alpha l$</td>
<td>$Z_0/\alpha l$</td>
</tr>
<tr>
<td>$L_r$</td>
<td>$1/\omega_0^2 C_r$</td>
<td>$Z_0\pi/2\omega_0$</td>
<td>$1/\omega_0^2 C_r$</td>
</tr>
<tr>
<td>$C_r$</td>
<td>$\pi/4\omega_0 Z_0$</td>
<td>$1/\omega_0^2 L_r$</td>
<td>$\pi/2\omega_0 Z_0$</td>
</tr>
</tbody>
</table>
Calculating S-parameters from voltages and currents determined by Kirchhoff laws:

\[ V_{1,2} = V_{1,2}^+ + V_{1,2}^-; \]
\[ I_{1,2} = I_{1,2}^+ + I_{1,2}^- = \frac{V_{1,2}^+-V_{1,2}^-}{Z_0}, \]
\[ V_{1,2}^\pm = \frac{1}{2}(V_{1,2} \pm Z_0I_{1,2}). \]

\[
\begin{pmatrix}
V_1^- \\
V_2^-
\end{pmatrix} =
\begin{pmatrix}
S_{11} & S_{12} \\
S_{21} & S_{22}
\end{pmatrix}
\begin{pmatrix}
V_1^+ \\
V_2^+
\end{pmatrix}
\]

If \( V = 1, \)
\[ V_1^+ = \frac{1}{2}(V_{1} + Z_0I_1) = \]
\[ \frac{1}{2}(V - Z_0I_1 + Z_0I_1) = 1/2. \]
\[ V_2^- = \frac{1}{2}(V_2 - Z_0I_2) = V_2. \]
\[ S_{21} = 2V_2 \]
Some numerical examples

Introduction

Using LT-Spice:
More complex features of real devices

Introduction

Bifurcation effects from nonlinearity (kinetic inductance)
More complex features of real devices

Introduction

Interaction with dipole two-level defects
More complex features of real devices

Introduction

Power-dependence of quality factors (saturation of two-level defects)
1 Introduction
- Superconducting qubits and circuit QED
- Physical quantum oscillators
- LC-oscillator
- Theory of transmission lines
- Transmission line resonators (TLR)
- Equivalent parameters of TLR
- S-parameters
- Some numerical examples
- More complex features of real devices

2 Designing TLR
- How the samples are made
- EDA software
- Simulating your design

3 Measuring TLR
- Sample connection
- Lab equipment
- Measurement process
- Data processing in Python
How the samples are made

Designing TLR

Mask lithography:

Direct lase write:

\[ M \, \text{Göppl et al., Journal of Applied Physics 104, 113904 (2008)} \]
How the samples are made

Designing TLR

Mask lithography:

Example result:

\[ M \text{ Göppl et al., Journal of Applied Physics 104, 113904 (2008)} \]
EDA software

Designing TLR

Everything from IC design, but simplified.

The LayoutEditor

is the most popular software to edit designs for MEMS and IC fabrication. It is also often used for Multi-Chip-Modules (MCM), Chip-on-Board (COB), Low temperature co-fired ceramics (LTCC), Monolithic Microwave Integrated Circuits (MMIC), printed circuit boards (PCB), thick film technology, thin film technology or any other technology using photomasks. The LayoutEditor can freely be used to access our services like the supplier with photomasks.

Give it a try and be prepared to be impressed!

Learn More  Download
EDA software
Designing TLR

KLayout - Your Mask Layout Friend

View  Edit  Generate  Analyze

Fast and accurate viewing of large layout files
Draw, modify, and transform hierarchical layout
Script-based generation, multiple layout version
Ensure, verify, measure layout and check analysis results

Layers

- 0/0
- 1/0
- 2/0
- 3/0
- 4/0
- 5/0
- 6/0
- 7/0
- 8/0
- 9/0
- 10/0
- 11/0
- 12/0
- 13/0
- 14/0
- 15/0

Color
Frame color
Useless
Animation
Style
Visibility
EDA software
Designing TLR

Scripting for design automation:

```python
import matplotlib.pyplot as plt
from math import cos, sin, atan2, pi, copy
from matplotlib.transforms import Affine2D
from matplotlib.transforms import Affine2D
from matplotlib.transforms import Affine2D
from matplotlib.transforms import Affine2D
from matplotlib.transforms import Affine2D

from .BaseClasses import ComplexBase
from .BaseClasses import RealPath, CPW
from .BaseClasses import Cell_type_1, CPW_arc

class CPWResonator:
    def __init__(self, origin, cpw_parameters, turn_radius, frequency, k, wavelength_fraction=1/4, coupling_length=900*3, offset_length=2900*3, meander_periods=4, neck_length=1800*3, end_primitive=None, trans_in=None):
        ...,
        A CPW resonator of wavelength fraction with automatic length calculation from given frequency.

        It's also possible to attach a primitive to the end of the resonator which should provide a get_phase_shift() method to calculate it's effective length as if it was just a straight CPW piece.

def place(self, cell, layer):
    N, R = self._meander_periods, self._turn_radius

    meander_length = (self._length - self._coupling_length - self._offset_length - 2*pi*R/N/4 - N*pi*R - 2*pi*R/2 - R - 2*pi*R/4 - self._neck_length)/(2*N+3/2)
    shape = "RLRL"*N+"RLRL"*R

    segment_lengths = [self._coupling_length + R, self._offset_length + 2*R] + [meander_length+R] + R + self._neck_length//R
    turn_angles = [pi/2, pi/2] + R + [-pi, pi]+[-pi, pi/2]

    line = CPW_RL_Path(self._origin, shape, self._cpw_parameters, self._turn_radius, segment_lengths, turn_angles, self._trans_in)
    line.place(cell, layer)

    self.start = line.connections[0]
    self.end = line.connections[1]
    self.alpha_start = line.angle_connections[0]
    self.alpha_end = line.angle_connections[1]
```

25 / 34
Several numeric packages exist
We use HFSS and Sonnet Suite.

Example results from Sonnet

HFSS interface
Introduction
- Superconducting qubits and circuit QED
- Physical quantum oscillators
- LC-oscillator
- Theory of transmission lines
- Transmission line resonators (TLR)
- Equivalent parameters of TLR
- S-parameters
- Some numerical examples
- More complex features of real devices

Designing TLR
- How the samples are made
- EDA software
- Simulating your design

Measuring TLR
- Sample connection
- Lab equipment
- Measurement process
- Data processing in Python
Sample connection

Measuring TLR
Lab equipment

Measuring TLR
Remote control of the VNA:
1. Making a wide frequency scan

```python
znb = Znb("VISA ADDRESS FROM NI-MAX SOFTWARE")

znb.select_S_param("S21")
znb.set_freq_limits(1e9, 10e9)
znb.set_power(-60)

znb.prepare_for_stb()
znb.sweep_single()
znb.wait_for_stb()

freqs, data = znb.get_frequencies(), znb.get_sdata()
```
For example, here one can see a scan (amplitude shown) of 6 notch-type resonators:

Resonator areas are shown with coloured crosses
2. Scanning each peak for a range of powers

\[ \text{powers} = \text{linspace}(-60, -10, 51) \]

\[ \text{full\_data} = {} \]

\[ \text{for resonator\_area in resonator\_areas:} \]
\[ \quad \text{znb.set\_freq\_limits(*resonator\_area)} \]

\[ \text{resonator\_data} = [] \]

\[ \text{for power in powers:} \]
\[ \quad \text{znb.set\_power(power)} \]
\[ \quad \text{znb.prepare\_for\_stb()} \]
\[ \quad \text{znb.sweep\_single()} \]
\[ \quad \text{znb.wait\_for\_stb()} \]
\[ \quad \text{resonator\_data.append(znb.get\_sdata())} \]

\[ \text{full\_data[resonator\_area]} = \text{resonator\_data} \]
Data processing in Python

Measuring TLR

Fitting the model for the notch-type resonator:

\[ S_{21}^{\text{notch}}(f_p) = a e^{i\alpha} e^{2\pi i f_p \tau} \left[ 1 - \frac{Q_l/Q'_e}{1 + 2iQ_l(f_p/f_r - 1)} \right], \]

Extracting the quality factors vs power:

I. $Q_e = 0.50$, $Q_{sf}/Q_{hp} = 4.47/10.33 \times 10^5$

II. $Q_e = 0.54$, $Q_{sf}/Q_{hp} = 3.27/9.79 \times 10^5$

III. $Q_e = 0.89$, $Q_{sf}/Q_{hp} = 2.66/9.14 \times 10^5$

IV. $Q_e = 0.82$, $Q_{sf}/Q_{hp} = 2.41/8.44 \times 10^5$

V. $Q_e = 1.06$, $Q_{sf}/Q_{hp} = 2.40/8.34 \times 10^5$

VI. $Q_e = 0.91$, $Q_{sf}/Q_{hp} = 2.73/8.35 \times 10^5$