

DOI: <https://doi.org/10.30898/1684-1719.2026.6.11>

УДК: 621.317.335.3; 537.868.3; 53.096; 53.083.2

СПОСОБ ВЫЧИСЛЕНИЯ ДИЭЛЕКТРИЧЕСКИХ ПАРАМЕТРОВ ЗАМОРОЖЕННЫХ БИОЛОГИЧЕСКИХ ТКАНЕЙ В РЕЗОНАНСНЫХ ИЗМЕРЕНИЯХ

А.В. Коннов ¹, Г.А. Белозеров ¹, А.Д. Лангман ², В.А. Сотсков ¹

¹ Всероссийский научно-исследовательский институт холодильной промышленности –
филиал «Федеральный научный центр пищевых систем им. В.М. Горбатова» РАН,
127422, Москва, ул. Костякова, 12

² Московский физико-технический институт
(национальный исследовательский университет),
141700, Долгопрудный, Институтский пер., д. 9.

Статья поступила в редакцию 30 мая 2026 г.

Аннотация. Рассмотрен новый подход к обработке результатов резонансных измерений диэлектрических свойств замороженных биологических тканей при температурах от -20 до -1°C в диапазоне частот 100–200 МГц в условиях фазовых переходов вымороженной воды, основанный на вычислении характеристического сопротивления резонатора с помощью 3D программы моделирования электродинамических систем. Предлагаемый способ не требует дополнительного эталонного диэлектрика. Возмущение резонансной частоты измерительной ячейки может достигать 100 %. Проведено сравнение точности определения диэлектрических параметров замороженных биологических тканей предложенным способом и ранее использованным методом.

Ключевые слова: замороженная биологическая ткань, фазовые переходы, диэлектрическая проницаемость, тангенс угла диэлектрических потерь, электромагнитные волны, очень высокие частоты, измерительная ячейка, резонатор, характеристическое сопротивление, 3D моделирование.

Финансирование: исследование выполнено в рамках Государственного задания ФГБНУ «ФНЦ пищевых систем им. В.М. Горбатова» РАН № FGUS-2025-0004.

Автор для переписки: Коннов Александр Викторович,
konnov.alexander@mail.ru

Введение

Резонансный метод широко применяется при измерении диэлектрических свойств различных диэлектриков в диапазонах радиочастот от единиц мегагерц до сотен гигагерц. [1]. Наиболее известная форма этого метода – метод малых возмущений [2]. В [3] рассмотрен вариант резонансного метода, специально разработанный для измерения диэлектрических параметров замороженных биотканей (аморфных несовершенных диэлектриков) при фазовых переходах. В этом методе для вычисления комплексной диэлектрической проницаемости (ДП) замороженной биоткани требуется дополнительное измерение резонансной частоты диэлектрика с известной ДП (эталонного диэлектрика). В [3] в качестве эталонного диэлектрика был использован фторопласт Ф-4, ДП которого известна с погрешностью 10 %, что вносит систематическую погрешность в расчеты диэлектрических параметров замороженных биотканей. Данное обстоятельство может существенно повлиять на результатах расчета ДП вблизи криоскопической температуры (температуры полного оттаивания замороженных биотканей), в силу скачкообразного роста ДП в этой области температур от значений 5–7 до значений 30–70.

В предлагаемом ниже способе обработки результатов резонансных измерений дополнительный эталонный диэлектрик не применяется, и эта погрешность устраняется при сохранении достоинств метода [3], таких как отсутствие жестких требований к малости уходов резонансной частоты в резонаторе измерительной ячейки с образцом и к форме самого образца.

1. Диэлектрическая проницаемость аморфных диэлектриков, размещенных в диэлектрической технологической оснастке

Для проведения измерений образец биоткани плотно без воздушных зазоров размещается в диэлектрической технологической оснастке (ДТО), предварительно замораживается до -20°C , а затем помещается в емкостной зазор резонатора измерительной ячейки, которая находится в холодильной камере с регулируемой температурой. После термостатирования образца биоткани при заданной отрицательной температуре производится измерение резонансной частоты резонатора измерительной ячейки.

ДТО представляет собой полый тонкостенный диэлектрический цилиндр с толщиной стенки δ_c и внутренним радиусом r_0 , закрытый сверху и снизу тонкими диэлектрическими (полиэтиленовыми) пленками толщиной $\delta_f / 2$ каждая с известной относительной диэлектрической проницаемостью ε_f . Высота цилиндра ДТО равна $d - \delta_f$, где d – высота емкостного зазора, причем $\delta_f \ll d$ и $\delta_f \ll r_0$.

Известно, что собственная круговая частота ω_0 рабочего вида колебаний резонатора (контура) без диэлектрика (с воздушным заполнением) связана с его собственной индуктивностью L_0 и емкостью C_0 следующим образом:

$$L_0 C_0 = \frac{1}{\omega_0^2}. \quad (1)$$

Если резонатор возмущен внесением в его емкостной зазор образца диэлектрика, то связь ухода круговой частоты $\Delta\omega$ и изменения емкости ΔC определяется из (1) как:

$$L_0 (C_0 + \Delta C) = \frac{1}{(\omega_0 - \Delta\omega)^2}. \quad (2)$$

Выражая из (1) индуктивность L_0 , после подстановки в (2), получим:

$$C_0 = \frac{\Delta C (\omega_0 - \Delta\omega)^2}{\Delta\omega (2\omega_0 - \Delta\omega)} \quad (3)$$

Суммарная измеряемая ДП образца диэлектрика в ДТО ε_Σ определяется как результат последовательного соединения трех плоских конденсаторов, заполненных разнородными диэлектриками: биотканью и двумя диэлектрическими пленками:

$$\varepsilon_\Sigma = \frac{\varepsilon_x \varepsilon_f}{\varepsilon_x k_f + \varepsilon_f (1 - k_f)}, \quad (4)$$

где ε_x – ДП замороженной биоткани, $k_f = \delta_f / d$ – коэффициент заполнения емкостного зазора d диэлектрическими пленками, $k_f \in [0, 1]$.

При введении диэлектрика с ДП ε_Σ в емкостной зазор в область однородного поля его емкость изменится на ΔC :

$$\Delta C = \varepsilon_0 (\varepsilon_\Sigma - 1) \frac{S}{d}, \quad (5)$$

здесь $S = \pi r_0^2$ – площадь торцевой поверхности измеряемого диэлектрика, $\varepsilon_0 = 8.854 \cdot 10^{-12}$ Ф/м – диэлектрическая постоянная вакуума.

По определению характеристическое сопротивление рабочего вида колебаний резонатора (контура) $\rho = \sqrt{\frac{L_0}{C_0}} = \frac{1}{\omega_0 C_0}$.

Отсюда

$$C_0 = \frac{1}{\omega_0 \rho}. \quad (6)$$

Подставляя (5) и (6) в (3), получим

$$\frac{1}{\omega_0 \rho} = \varepsilon_0 (\varepsilon_\Sigma - 1) \frac{S}{d} \frac{(\omega_0 - \Delta\omega)^2}{\Delta\omega (2\omega_0 - \Delta\omega)}, \quad (7)$$

Откуда следует, что

$$\varepsilon_\Sigma = 1 + \frac{d}{S} \frac{\Delta\omega (2\omega_0 - \Delta\omega)}{\rho \varepsilon_0 \omega_0 (\omega_0 - \Delta\omega)^2}. \quad (8)$$

Из (4) ДП измеряемого диэлектрика

$$\varepsilon_x = \frac{1 - k_f}{\frac{1}{\varepsilon_\Sigma} - \frac{k_f}{\varepsilon_f}}. \quad (9)$$

Пусть диэлектрик – фторопласт без ДТО, тогда $\varepsilon_\Sigma = \varepsilon_\phi$ и $\Delta\omega = \Delta\omega_\phi$.

В этом случае (8) примет вид:

$$\varepsilon_\phi = 1 + \frac{d}{S} \frac{\Delta\omega_\phi(2\omega_0 - \Delta\omega_\phi)}{\rho \varepsilon_0 \omega_0 (\omega_0 - \Delta\omega_\phi)^2}. \quad (10)$$

Из (10) выражение

$$\frac{d}{S} \frac{1}{\rho \varepsilon_0 \omega_0} = (\varepsilon_\phi - 1) \frac{(\omega_0 - \Delta\omega_\phi)^2}{\Delta\omega_\phi(2\omega_0 - \Delta\omega_\phi)}. \quad (11)$$

Подставляя (11) в (8) и полагая в (9) $k_f = 0$ (диэлектрик без ДТО), получим:

$$\varepsilon_x = 1 + (\varepsilon_\phi - 1) \frac{(\omega_0 - \Delta\omega_\phi)^2}{\Delta\omega_\phi(2\omega_0 - \Delta\omega_\phi)} \frac{\Delta\omega_x(2\omega_0 - \Delta\omega_x)}{(\omega_0 - \Delta\omega_x)^2}. \quad (12)$$

Так как $\Delta\omega_x = \omega_0 - \omega_x$ и $\Delta\omega_\phi = \omega_0 - \omega_\phi$, то (12) можно переписать как:

$$\varepsilon_x = 1 + (\varepsilon_\phi - 1) \frac{\omega_\phi^2}{\omega_x^2} \frac{(\omega_0^2 - \omega_x^2)}{(\omega_0^2 - \omega_\phi^2)}. \quad (13)$$

Формула (13) с точностью до обозначений совпадает с формулой, полученной в [3] другим способом, для твердых диэлектриков.

Для вычисления ε_x по формуле (9) с учетом (8) следует определить значение характеристического сопротивления ρ рабочего вида колебаний резонатора.

Его можно определить двумя способами:

1) Из формулы (10) для фторопласта

$$\rho = \frac{d}{S} \frac{(\omega_0^2 - \omega_\phi^2)}{\varepsilon_0(\varepsilon_\phi - 1)\omega_0\omega_\phi^2}, \quad (14)$$

но тогда этот способ эквивалентен рассмотренному в работе [3].

2) характеристическое сопротивление ρ можно определить численно, используя 3D программу моделирования электродинамических систем в режиме расчета собственных частот, с помощью имеющегося в ней макроса (подпрограммы) [4]. Макрос вычисляет полевым методом характеристическое сопротивление, которое в этом случае представляется в виде [5]:

$$\rho = \frac{\left[\int_l E_l(\zeta) d\zeta \right]^2}{2\omega_0 W}, \quad (15)$$

здесь l – линия, проведенная параллельно направлению вектора электрического высокочастотного (ВЧ) поля в однородной области емкостного зазора (Рис. 1), W – энергия, запасенная в резонаторе за период колебания, $E_l(\zeta)$ – амплитуда ВЧ поля рабочего вида колебаний в емкостном зазоре. Для однородного ВЧ поля интеграл в (15) вычисляется аналитически $\int_l E_l(\zeta) d\zeta = E_i d$, где E_i – амплитуда ВЧ поля, взятая в любой точке в однородной области емкостного зазора. Отметим также, что в режиме расчета собственных частот $W \equiv 1$ Дж.

Расчет в режиме собственных частот по 3D программе моделирования электродинамических систем распределения ВЧ электрических полей на рабочем виде колебаний по геометрии резонатора [3] дает характеристическое сопротивление $\rho = 119.7$ Ом на резонансной частоте $f_0 = 221.9$ МГц (Рис. 1).

Полученная в результате численного расчета собственная резонансная частота несколько ниже экспериментально измеренной ($f_0 = 224$ МГц), так как при 3D моделировании в режиме собственных частот из полости резонатора были убраны элементы связи с входным и выходным трактами.

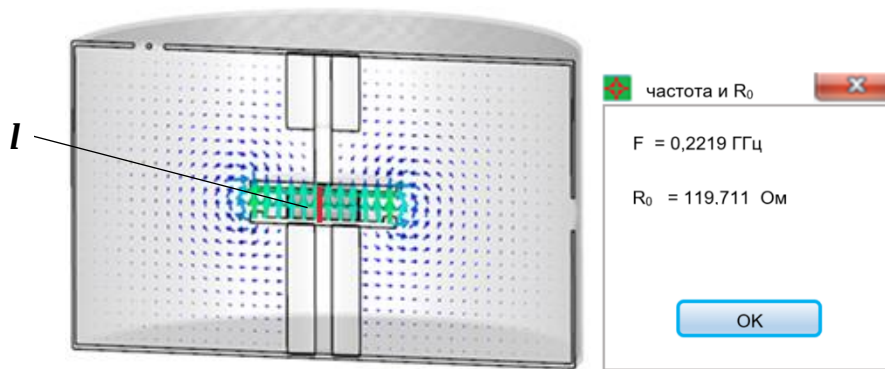


Рис. 1. Распределение ВЧ полей, собственная частота f_0 и ρ рабочего вида колебаний резонатора.

Характеристическое сопротивление можно определить по формуле (14) для различных значений ДП фторопласта. Так для образца фторопласта с $\varepsilon_\phi = 1.95$ высотой 20 мм, $\varnothing 50$ мм без ДТО $\rho = 125.0$ Ом при $f_\phi = 209.3$ МГц и $f_0 = 224.0$ МГц. При той же геометрии образца фторопласта, но с $\varepsilon_\phi = 1.99$ $\rho = 120.0$ Ом, что достаточно близко к значению, рассчитанному численно по 3D программе моделирования электродинамических систем. “Подстройку” ДП эталонного диэлектрика по известному характеристическому сопротивлению рабочего вида колебаний можно рассматривать как один из способов уточнения значения его ДП.

Отсюда следует, что погрешность экспериментального определения характеристического сопротивления с помощью формулы (14) не менее 4 % и определяется разбросом значений ДП промышленных образцов фторопласта.

Для устранения этой погрешности в определении ДП предпочтительно использовать формулы (9) и (8) с численно рассчитанным значением ρ без использования дополнительного эталонного диэлектрика.

2. Диэлектрические потери

Диэлектрические потери в диэлектрике учитываются введением комплексной ДП:

$$\varepsilon = \varepsilon' - j\varepsilon'', \quad (16)$$

здесь ε' – ДП, а ε'' – фактор диэлектрических потерь измеряемых диэлектриков.

В [3] получена формула для ε'' .

$$\varepsilon_x'' = \frac{(\varepsilon_\Sigma + \xi)}{Q_x} - \frac{f_0(1 + \xi)}{f_x Q_0}, \quad (17)$$

здесь индекс «х» обозначает величины, относящиеся к измеряемому образцу диэлектрика в ДТО: значениям резонансной частоты f_x , добротности резонатора Q_x и мнимой части комплексной ДП, а индекс «0» – к значениям частоты f_0 и добротности резонатора Q_0 , заполненного сухим воздухом, $\xi = \frac{C_{ex}}{C_i}$ – параметр краевого поля, $C_0 = C_i + C_{ex}$, где C_i – емкость области, занимаемой диэлектриком, C_{ex} – оставшаяся емкость резонатора на рабочем виде колебаний.

Из формулы (6)

$$C_0 = C_i + C_{ex} = C_i(1 + \xi) = \frac{1}{\omega_0 \rho}. \quad (18)$$

Отсюда

$$\xi = \frac{d}{2\pi \varepsilon_0 S f_0 \rho} - 1. \quad (19)$$

Тангенс угла диэлектрических потерь диэлектрика по определению:

$$\operatorname{tg} \delta_x = \frac{\varepsilon_x''}{\varepsilon_x'}, \quad (20)$$

где ε_x'' вычисляется по формуле (17), а ε_x' – по формуле (9).

3. Зависимости диэлектрических параметров некоторых замороженных биотканей от температуры в диапазоне частот 150 ± 30 МГц

Ниже, на рисунках 2 – 5, приведены результаты вычислений ДП и $\operatorname{tg} \delta$ различных замороженных биологических тканей, на основе экспериментальных данных резонансных измерений, полученных в [6], по формулам (8, 9) и (17–20) настоящей работы с использованием вычисленного значения характеристического сопротивления резонатора ρ и, для сопоставления, по формулам, приведенным в работе [3].

Алгоритм обработки массивов экспериментальных данных различных замороженных биологических тканей (резонансные частоты, ширины полос пропускания, температуры) двумя способами реализован на алгоритмическом языке “Питон” [7], листинг кода которого приведен в Приложении 1.

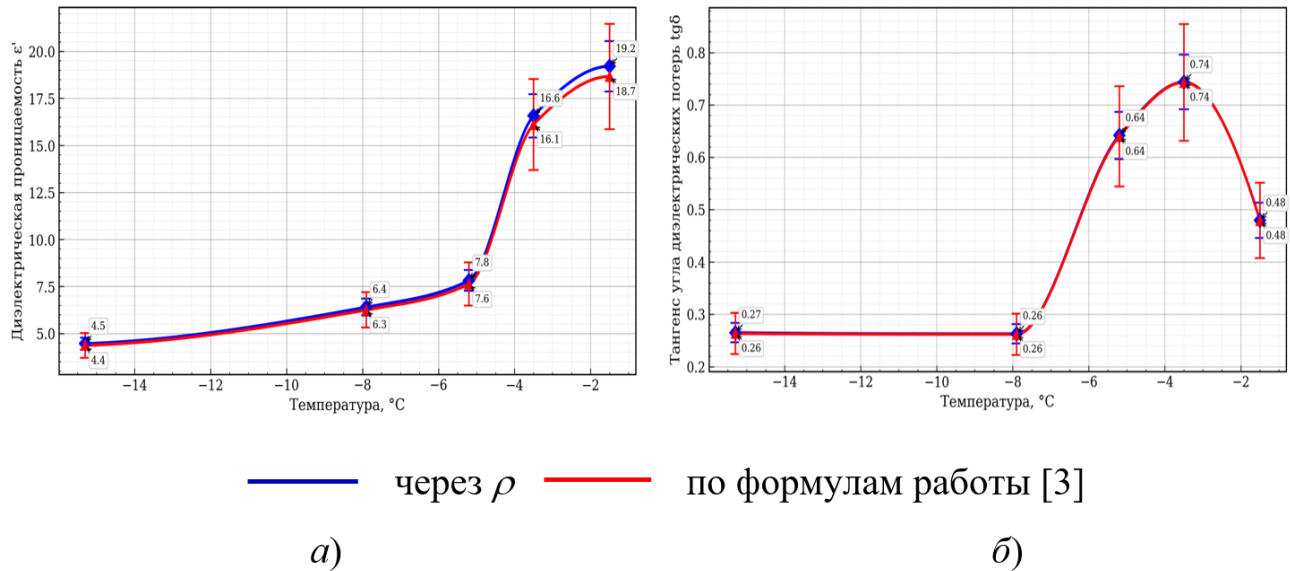


Рис. 2. Зависимости ДП (а) и $\tan \delta$ (б) сырой постной говядины (влажность 67 %) от температуры.

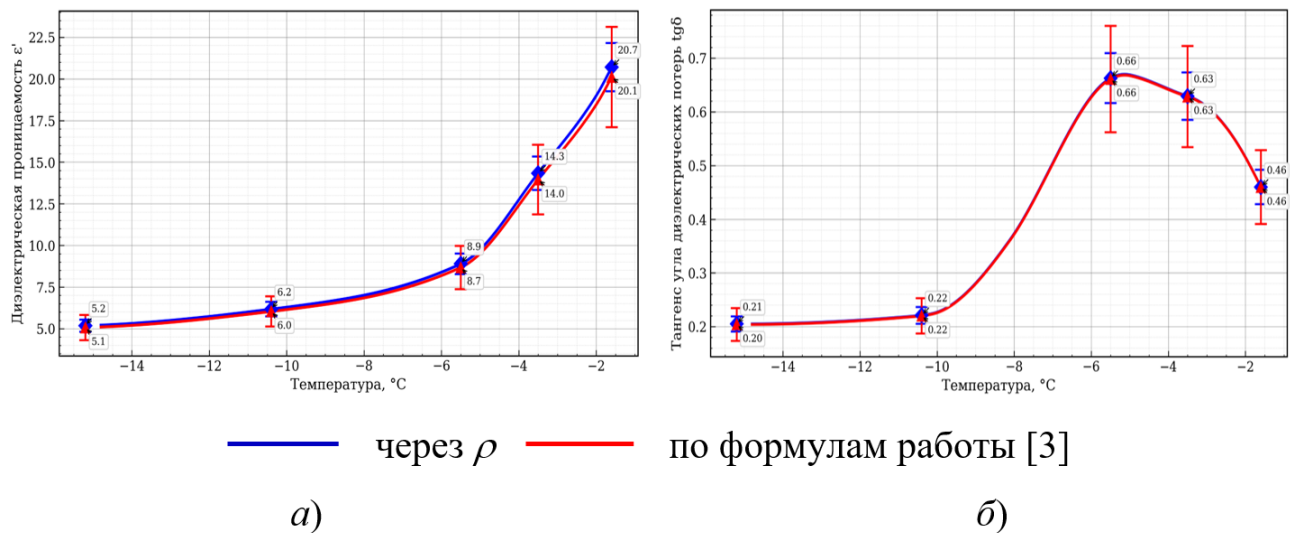


Рис. 3. Зависимости ДП (а) и $\tan \delta$ (б) карбонада (влажность 73 %) от температуры.

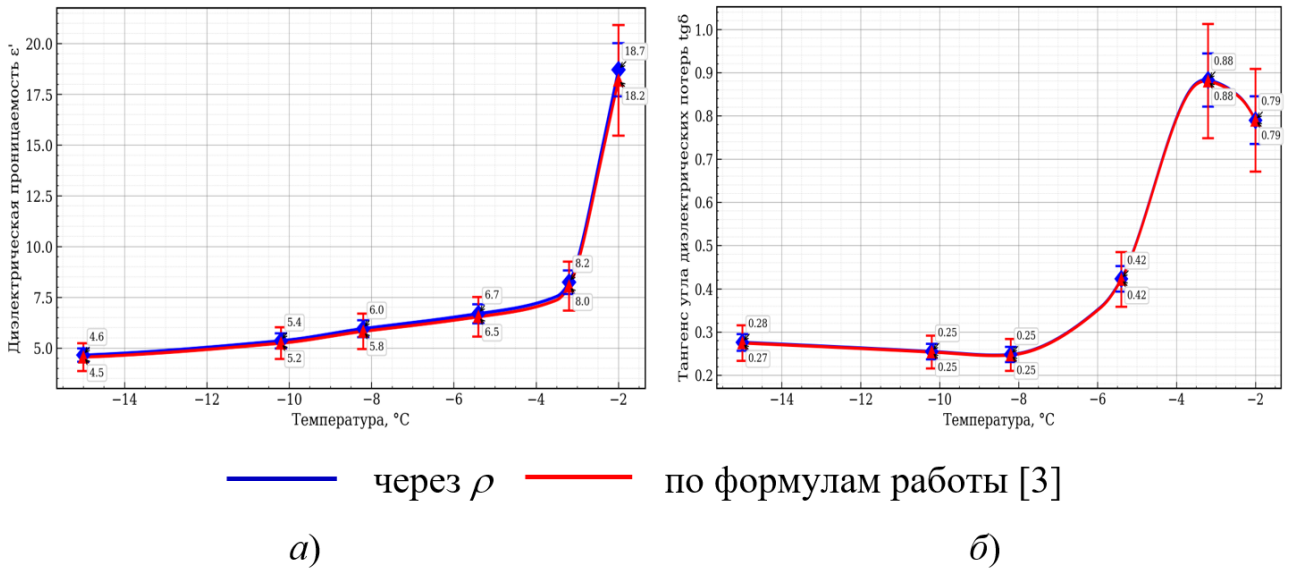


Рис. 4. Зависимости ДП (а) и $\tan \delta$ (б) свиной шейки (влажность 68,2 %) от температуры.

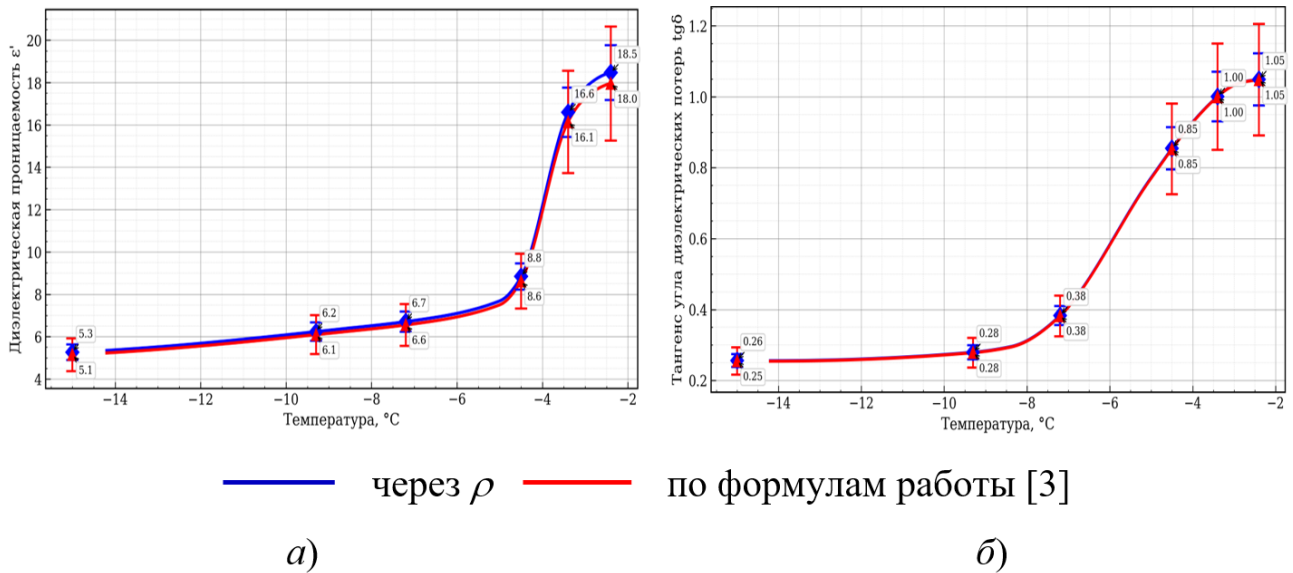


Рис. 5. Зависимости ДП (а) и $\tan \delta$ (б) спинки форели (влажность 60 %, жирность 16 %) от температуры.

На рисунках 2–5 кривые, выделенные синим цветом, построены по формулам (8, 9) и (17–20) настоящей работы, а – красным – по формулам работы [3].

Заключение

1) Предложенный подход к определению резонансным методом диэлектрических параметров замороженных биологических тканей при температурах от -20 до -1°C в диапазоне частот 100–200 МГц в условиях фазовых переходов вымороженной воды, основанный на вычислении характеристического сопротивления резонатора с помощью 3D программы моделирования электродинамических систем, упрощает экспериментальные исследования, так как не требует измерений параметров дополнительного эталонного диэлектрика, и исключает погрешности, связанные с технологическим разбросом параметров самих эталонных материалов.

2) Выполнено трехмерное математическое моделирование резонатора измерительной ячейки, рассчитаны частота и распределение ВЧ электрических полей на рабочем виде колебаний, по которому вычислено значение его характеристического сопротивления.

3) Разработан алгоритм и программа обработки массивов данных резонансных измерений предложенным способом, основанным на вычислении характеристического сопротивления резонатора, и методом работы [3].

4) Температурные зависимости диэлектрических параметров замороженных биологических тканей, как полученные в представленной работе, так и в [6], хорошо соответствуют друг другу в пределах погрешности измерений.

Литература

1. Venkatesh M.S. and Raghavan G.S.V. An overview of dielectric properties measuring techniques // Canadian Biosystems Engineering. – 2005. – V. 47. – P. 7.15–7.30.
2. Никольский В.В. Теория электромагнитного поля. – М.: УРСС, 2020, С. 400.
3. Коннов А.В., Белозеров Г.А., Сотсков В.А. Определение диэлектрических характеристик замороженных биологических тканей в условиях протекающих в них фазовых переходов. // Журнал радиоэлектроники. – 2026. – № 3. <https://doi.org/10.30898/1684-1719.2026.3.12>
4. Grigoriev A.D., Salimov R.V., Tikhonov R.I. Modeling of Complicated Microwave Devices by RFS Code // Int. Conf. APEDE 2012, Sept.19–20, Saratov, 2012. – P. 175–182.
5. Хаби В.С. Измерение характеристического сопротивления резонатора с бессеточным зазором // Электронная техника. Серия 1. Электроника СВЧ. – 1971. – № 3. – С. 138–140.
6. Белозеров А.Г., Коннов А.В., Белозеров Г. А, Сотсков В.А., Онацкий И.Т. Исследование диэлектрических свойств замороженных мясных и рыбных полуфабрикатов при квазистатическом отеплении в условиях фазового перехода // Все о мясе. – 2026. – № 6 (в печати).
7. Лутц М. Изучаем Python. Т1.– СПб.: Диалектика, 2019, С. 832.

Для цитирования:

Коннов А.В., Белозеров Г.А., Лангман А.Д., Сотсков В.А. Способ вычисления диэлектрических параметров замороженных биологических тканей в резонансных измерениях // Журнал радиоэлектроники. – 2026. – № 6. <https://doi.org/10.30898/1684-1719.2026.6.11>

Приложение 1.

""""

Листинг программы подготовлен как приложение к тексту статьи: «Способ вычисления диэлектрических характеристик замороженных биологических тканей в резонансных измерениях»

Для обработки экспериментальных данных применяются два подхода:

1. Расчет по смещению резонансной частоты и характеристическому сопротивлению ρ резонатора.
2. Расчет по смещению резонансной частоты с использованием диэлектрической проницаемости и резонансной частоты эталонного диэлектрика-фторопласта.

""""

```
import math
from io import BytesIO
from pathlib import Path
from collections import defaultdict

import matplotlib.pyplot as plt
import numpy as np
from scipy.interpolate import PchipInterpolator

try:
    from docx import Document
    from docx.shared import Inches
    DOCX_AVAILABLE = True
except ImportError:
    Document = None
    Inches = None
    DOCX_AVAILABLE = False

# =====
# 1. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ: резонансные частоты, полосы пропускания и температуры
# =====
# Для каждого набора задаются:
# T : температура образца, °C
# fx : резонансная частота, МГц
# df : полоса пропускания, МГц
data_sets = {
    1: {
        "title": "говядина",
        "T": [-15.3, -7.9, -5.2, -3.5, -1.5],
        "fx": [186.67, 173.05, 164.99, 133.81, 127.89],
        "df": [21.58, 23.76, 58.39, 71.7, 46.23],
    },
    2: {
        "title": "карбонад_1",
        "T": [-15.2, -10.4, -5.5, -3.5, -1.6],
        "fx": [181.4, 174.0, 159.4, 145.9, 135.1],
        "df": [18.29, 19.44, 57.01, 60.3, 29.71],
    },
    3: {
        "title": "карбонад_2",
        "T": [-14.8, -10.1, -5.5, -3.5, -2.0, -1.5],
        "fx": [181.51, 174.08, 159.34, 130.82, 120.36, 115.31],
        "df": [17.5, 20.7, 63.4, 56.2, 60.0, 60.0],
    },
    4: {
        "title": "карбонад_3",
        "T": [-15.3, -10.6, -8.1, -5.0, -1.5],
        "fx": [180.72, 175.01, 171.0, 157.48, 125.22],
        "df": [17.87, 20.18, 25.76, 67.68, 39.85],
    },
    5: {
        "title": "Фореель_1",
        "T": [-15.0, -9.3, -7.2, -4.5, -3.4, -2.4],
```

```

    "fx": [184.71, 175.79, 172.64, 163.89, 134.8, 129.7],
    "df": [20.96, 22.54, 27.65, 65.2, 79.9, 93.91],
  },
  6: {
    "title": "форель_2",
    "T": [-1.3, -14.2, -8.4, -5.0, -3.1, -2.4, -1.2],
    "fx": [126.91, 176.79, 171.25, 164.76, 130.57, 129.18, 127.8],
    "df": [77.92, 22.92, 29.31, 81.8, 113.5, 106.9, 90.0],
  },
  7: {
    "title": "шейка_1",
    "T": [-15.0, -10.2, -8.2, -5.4, -3.2, -2.0],
    "fx": [186.91, 181.78, 175.93, 172.45, 168.94, 126.0],
    "df": [27.7, 24.71, 23.19, 42.14, 86.78, 87.6],
  },
  8: {
    "title": "шейка_2",
    "T": [-15.0, -10.6, -8.9, -5.9, -3.5, -2.4, -1.5],
    "fx": [183.56, 178.6, 176.87, 171.28, 163.36, 136.58, 128.4],
    "df": [17.49, 19.04, 20.18, 27.01, 72.02, 70.02, 51.2],
  },
  9: {
    "title": "шпик",
    "T": [-14.8, -10.4, -8.5, -5.5, -3.3, -1.3, 1.3],
    "fx": [206.85, 203.71, 202.05, 197.56, 191.81, 184.25, 176.73],
    "df": [48.64, 46.62, 46.67, 47.3, 48.16, 39.85, 30.45],
  },
}
# =====
# 2. ФИЗИЧЕСКИЕ КОНСТАНТЫ И ПАРАМЕТРЫ ИЗМЕРИТЕЛЬНОЙ ЯЧЕЙКИ
# =====
# RHO_0      : характеристическое сопротивление резонатора ρ
# F0_MHZ / F0_HZ : резонансная частота пустого резонатора.
# PLATE_AREA  : площадь торцевой поверхности диэлектрика, м².
# EPSILON_0   : электрическая постоянная вакуума, Ф/м.
# SAMPLE_GAP  : емкостной зазор резонатора, м.
# FILM_THICKNESS : толщина защитной пленки, м.
# FILM_EPSILON : диэлектрическая проницаемость пленки.
# Q0          : добротность пустого резонатора.
# EPS_FLUOR   : диэлектрическая проницаемость фторопласта.
RHO_0 = 120.0
F0_MHZ = 223.0
F0_HZ = F0_MHZ * 1e6
PLATE_AREA = 0.0019635
EPSILON_0 = 8.854e-12
SAMPLE_GAP = 0.02
FILM_THICKNESS = 0.0002
FILM_EPSILON = 2.5
Q0 = 85.0

F_AIR_MHZ = 221.34
F_FLUOR_MHZ = 209.16
EPS_FLUOR = 2.0

H2 = 20e-3
H_PL = 0.2e-3
# =====
# 3. ОЦЕНКИ ПОГРЕШНОСТЕЙ
# =====
# Эти значения используются для построения интервалов погрешностей на графиках.
ERROR_E1_PERCENT = 10
ERROR_E2_PERCENT = 20
ERROR_TG_RHO_PERCENT = 10

```

```
ERROR_TG_EXP_PERCENT = 20
```

```
# =====
```

```
# 4. ФУНКЦИИ ДЛЯ РАСЧЕТА  $\epsilon'$  И  $\text{tg}\delta$ 
```

```
# =====
```

```
def eps_from_frequency_method(
```

```
    fx_mhz,
    f_air_mhz=F_AIR_MHZ,
    f_fluor_mhz=F_FLUOR_MHZ,
    eps_fluor=EPS_FLUOR,
    h2=H2,
    h_pl=H_PL,
```

```
):
```

```
    """
```

Расчет действительной части диэлектрической проницаемости ϵ' с использованием эталонного диэлектрика-фторопласта.

```
    """
```

```
    r_c = h2 / (h2 - 2 * h_pl)
    b_sigma = 1.0 + (eps_fluor - 1.0) * (
        f_fluor_mhz**2 * (f_air_mhz**2 - fx_mhz**2)
    ) / (
        fx_mhz**2 * (f_air_mhz**2 - f_fluor_mhz**2)
    )
    return 1.0 / (r_c / b_sigma - (r_c - 1.0) / eps_fluor)
```

```
def eps_from_rho_method(fx_mhz, film_epsilon=FILM_EPSILON):
```

```
    """
```

Расчет действительной части диэлектрической проницаемости ϵ' с использованием ρ .

```
    """
```

```
    fx_hz = fx_mhz * 1e6
    w = 2 * math.pi * fx_hz
    w0 = 2 * math.pi * F0_HZ
    dw = -(w - w0)

    a_value = (
        (dw * (2 * w0 - dw)) / (RHO_0 * w0 * (w0 - dw) ** 2)
        + (EPSILON_0 * PLATE_AREA) / (SAMPLE_GAP + 2 * FILM_THICKNESS)
    )

    return (
        (
            (a_value**(-1) * EPSILON_0 * film_epsilon * PLATE_AREA
             - 2 * FILM_THICKNESS)
        ) ** -1
    ) * SAMPLE_GAP * film_epsilon
```

```
def tg_delta_from_frequency_method(
```

```
    fx_mhz,
    df_mhz,
    eps_prime,
    eps_fluor=EPS_FLUOR,
    f_air_mhz=F_AIR_MHZ,
    f_fluor_mhz=F_FLUOR_MHZ,
    f0_mhz=F0_MHZ,
    q0=Q0,
```

```
):
```

```
    """
```

Расчет тангенса угла диэлектрических потерь $\text{tg}\delta$ с использованием эталонного диэлектрика-фторопласта.

```
    """
```

```
    xi_exp = (eps_fluor * f_fluor_mhz**2 - f_air_mhz**2) / (
        f_air_mhz**2 - f_fluor_mhz**2
    )
    eps_double_prime = (
        (eps_prime + xi_exp) * (df_mhz / fx_mhz)
```

```

    - (f0_mhz * (1.0 + xi_exp)) / (q0 * fx_mhz)
)
return eps_double_prime / eps_prime

def xi_rho():
    """ Параметр краевого поля  $\xi$  для  $\rho$ -метода. """
    return SAMPLE_GAP / (
        2 * math.pi * EPSILON_0 * PLATE_AREA * F0_HZ * RHO_0
    ) - 1.0

def tg_delta_from_rho_method(
    fx_mhz,
    df_mhz,
    eps_prime,
    xi_value,
    f0_hz=F0_HZ,
    q0=Q0,
):
    """
    Расчет тангенса угла диэлектрических потерь  $\text{tg}\delta$  через  $\rho$ .
    """
    fx_hz = fx_mhz * 1e6
    df_hz = df_mhz * 1e6
    qx = fx_hz / df_hz
    eps_double_prime = (
        (eps_prime + xi_value) / qx
        - (f0_hz * (1.0 + xi_value)) / (fx_hz * q0)
    )
    return eps_double_prime / eps_prime
# =====
# 5. ПОДГОТОВКА И УСРЕДНЕНИЕ ДАННЫХ
# =====
def prepare_dataset(raw, film_epsilon=FILM_EPSILON):
    """
    Подготавливает данные одного набора:
    1. усредняет частоты по одинаковым температурам;
    2. рассчитывает  $\epsilon'(T)$  двумя методами.
    """
    temperature_groups = defaultdict(list)
    for temperature, frequency in zip(raw["T"], raw["fx"]):
        temperature_groups[temperature].append(frequency)

    t_vals, fx_vals, e_rho_vals, e_exp_vals = [], [], [], []
    for temperature in sorted(temperature_groups):
        fx_mean = np.mean(temperature_groups[temperature])
        t_vals.append(temperature)
        fx_vals.append(fx_mean)
        e_rho_vals.append(eps_from_rho_method(fx_mean, film_epsilon))
        e_exp_vals.append(eps_from_frequency_method(fx_mean))

    return (
        np.array(t_vals),
        np.array(fx_vals),
        np.array(e_rho_vals),
        np.array(e_exp_vals),
    )

def mean_bandwidth_by_temperature(raw):
    """Усреднение полосы пропускания  $\Delta f$  по одинаковым температурам."""
    df_groups = defaultdict(list)
    for temperature, bandwidth in zip(raw["T"], raw["df"]):
        df_groups[temperature].append(bandwidth)
    return np.array([np.mean(df_groups[t]) for t in sorted(df_groups)], dtype=float)

```

```

def prepare_tg_dataset(raw, film_epsilon=FILM_EPSILON):
    """
    Подготавливает  $\text{tg}\delta(T)$  для одного набора данных двумя методами:
    через  $\rho$  и  $\epsilon$  использованием эталонного диэлектрика.
    """
    t_vals, fx_vals, e_rho_vals, e_exp_vals = prepare_dataset(raw, film_epsilon)
    df_vals = mean_bandwidth_by_temperature(raw)
    xi_value = xi_rho()

    tg_rho_vals = []
    tg_exp_vals = []

    for fx_mhz, df_mhz, e_rho, e_exp in zip(fx_vals, df_vals, e_rho_vals, e_exp_vals):
        tg_rho_vals.append(tg_delta_from_rho_method(fx_mhz, df_mhz, e_rho, xi_value))
        tg_exp_vals.append(tg_delta_from_frequency_method(fx_mhz, df_mhz, e_exp))

    return np.array(t_vals), np.array(tg_rho_vals), np.array(tg_exp_vals)

def smooth_curve(x, y, points=300):
    """Сглаживание кривой монотонным интерполятором PCHIP."""
    x = np.array(x, dtype=float)
    y = np.array(y, dtype=float)

    unique_x = np.unique(x)
    if len(unique_x) < len(x):
        y = np.array([np.mean(y[x == x_value]) for x_value in unique_x], dtype=float)
        x = unique_x

    order = np.argsort(x)
    x_sorted = x[order]
    y_sorted = y[order]

    interpolator = PchipInterpolator(x_sorted, y_sorted)
    x_smooth = np.linspace(x_sorted.min(), x_sorted.max(), points)
    return x_smooth, interpolator(x_smooth)

def average_eps_processed(processed):
    """
    Усреднение набора кривых  $\epsilon(T)$  в общем температурном диапазоне.
    """
    left = max(t.min() for t, _, _ in processed)
    right = min(t.max() for t, _, _ in processed)
    x_common = np.linspace(left, right, 300)

    e_rho_list = [PchipInterpolator(t, e_rho)(x_common) for t, e_rho, _ in processed]
    e_exp_list = [PchipInterpolator(t, e_exp)(x_common) for t, _, e_exp in processed]

    t_ref = processed[0][0]
    e_rho_points = np.mean([PchipInterpolator(t, e_rho)(t_ref) for t, e_rho, _ in processed], axis=0)
    e_exp_points = np.mean([PchipInterpolator(t, e_exp)(t_ref) for t, _, e_exp in processed], axis=0)

    return x_common, np.mean(e_rho_list, axis=0), np.mean(e_exp_list, axis=0), t_ref, e_rho_points, e_exp_points

def average_tg_processed(processed):
    """
    Усреднение набора кривых  $\text{tg}\delta(T)$  в общем температурном диапазоне.
    """
    left = max(t.min() for t, _, _ in processed)
    right = min(t.max() for t, _, _ in processed)
    x_common = np.linspace(left, right, 300)

    tg_rho_list = [PchipInterpolator(t, tg_rho)(x_common) for t, tg_rho, _ in processed]

```

```

tg_exp_list = [PchipInterpolator(t, tg_exp)(x_common) for t, _, tg_exp in processed]

t_ref = processed[0][0]
tg_rho_points = np.mean([PchipInterpolator(t, tg_rho)(t_ref) for t, tg_rho, _ in processed], axis=0)
tg_exp_points = np.mean([PchipInterpolator(t, tg_exp)(t_ref) for t, _, tg_exp in processed], axis=0)

return x_common, np.mean(tg_rho_list, axis=0), np.mean(tg_exp_list, axis=0), t_ref, tg_rho_points, tg_exp_points
# =====
# 6. ГРУППИРОВКА ОБРАЗЦОВ ПО ТИПАМ
# =====
product_groups = {
    "Говядина": [data_sets[1]],
    "Карбонады": [data_sets[2], data_sets[3], data_sets[4]],
    "Форе́ли": [data_sets[5], data_sets[6]],
    "Шейки": [data_sets[7], data_sets[8]],
    "Шпик": [data_sets[9]],
}
# =====
# 7. НАСТРОЙКА ОФОРМЛЕНИЯ ГРАФИКОВ
# =====
def configure_publication_style():
    """Глобальные параметры matplotlib для публикационного оформления."""
    plt.rcParams.update(
        {
            "font.family": "serif",
            "font.size": 12,
            "axes.linewidth": 1.2,
            "xtick.direction": "in",
            "ytick.direction": "in",
            "xtick.major.size": 5,
            "ytick.major.size": 5,
            "legend.frameon": False,
        }
    )

def style_axes(ax, title, ylabel):
    """Единое оформление осей, сетки и легенды."""
    ax.set_title(title, fontsize=15, fontweight="bold", pad=14)
    ax.set_xlabel("Температура, °C", fontsize=13)
    ax.set_ylabel(ylabel, fontsize=13)
    ax.grid(True, which="major", linestyle="-", linewidth=0.7, alpha=0.55, color="gray")
    ax.minorticks_on()
    ax.grid(True, which="minor", linestyle=":", linewidth=0.5, alpha=0.35, color="gray")
    ax.legend(loc="upper center", bbox_to_anchor=(0.5, -0.16), ncol=2, frameon=False, fontsize=11)
# =====
# 8. ПОСТРОЕНИЕ ГРАФИКОВ  $\varepsilon(T)$ 
# =====
def annotate_series(ax, x_vals, y_vals, fmt, offset_y):
    """Подписывает основные экспериментальные точки на графике."""
    for x_value, y_value in zip(x_vals, y_vals):
        ax.annotate(
            fmt.format(y_value),
            xy=(x_value, y_value),
            xytext=(6, offset_y),
            textcoords="offset points",
            fontsize=9,
            color="black",
            bbox=dict(boxstyle="round,pad=0.25", fc="white", ec="lightgray", alpha=0.95),
            arrowprops=dict(arrowstyle="->", color="black", lw=0.9),
        )

def plot_eps_group(ax, group_title, raw_list):
    """

```

Строит график $\varepsilon(T)$ для одной группы образцов.

Для одиночного образца используется исходный набор,

для нескольких образцов строится усредненная кривая.

"""

```

if len(raw_list) == 1:
    t_vals, _, e_rho_vals, e_exp_vals = prepare_dataset(raw_list[0], FILM_EPSILON)

    x_rho, y_rho = smooth_curve(t_vals, e_rho_vals)
    x_exp, y_exp = smooth_curve(t_vals, e_exp_vals)

    ax.plot(x_rho, y_rho, "b-", linewidth=2.5, label="через  $\rho$ ")
    ax.plot(x_exp, y_exp, "r-", linewidth=2.5, label="эксперимент")

    ax.errorbar(
        t_vals,
        e_rho_vals,
        yerr=np.array(e_rho_vals) * ERROR_E1_PERCENT / 100,
        fmt="D",
        color="blue",
        markersize=8,
        capsize=5,
        elinewidth=1.5,
        capthick=1.5,
    )
    ax.errorbar(
        t_vals,
        e_exp_vals,
        yerr=np.array(e_exp_vals) * ERROR_E2_PERCENT / 100,
        fmt="^",
        color="red",
        markersize=8,
        capsize=5,
        elinewidth=1.5,
        capthick=1.5,
    )

    annotate_series(ax, t_vals, e_rho_vals, "{:.1f}", 12)
    annotate_series(ax, t_vals, e_exp_vals, "{:.1f}", -15)
    title = group_title
else:
    processed = [
        (t_vals, e_rho_vals, e_exp_vals)
        for t_vals, _, e_rho_vals, e_exp_vals in (
            prepare_dataset(raw, FILM_EPSILON) for raw in raw_list
        )
    ]
    x_common, e_rho_mean, e_exp_mean, t_ref, e_rho_points, e_exp_points = average_eps_processed(processed)

    ax.plot(x_common, e_rho_mean, "b-", linewidth=2.5, label="через  $\rho$ ")
    ax.plot(x_common, e_exp_mean, "r-", linewidth=2.5, label="эксперимент")

    ax.errorbar(
        t_ref,
        e_rho_points,
        yerr=np.array(e_rho_points) * ERROR_E1_PERCENT / 100,
        fmt="D",
        color="blue",
        markersize=9,
        capsize=6,
        elinewidth=1.8,
        capthick=1.8,
    )
    ax.errorbar(

```

```

t_ref,
e_exp_points,
yerr=np.array(e_exp_points) * ERROR_E2_PERCENT / 100,
fmt="^",
color="red",
markersize=9,
capsize=6,
elinewidth=1.8,
capthick=1.8,
)

annotate_series(ax, t_ref, e_rho_points, "{:.1f}", 12)
annotate_series(ax, t_ref, e_exp_points, "{:.1f}", -15)
title = group_title

style_axes(ax, title, "Диэлектрическая проницаемость  $\epsilon$ ")
# =====
# 9. ПОСТРОЕНИЕ ГРАФИКОВ  $\text{tg}\delta(T)$ 
# =====
def plot_tg_group(ax, group_title, raw_list):
    """
    Строит график  $\text{tg}\delta(T)$  для одной группы образцов.
    Для одиночного образца используется исходный набор,
    для нескольких образцов строится усредненная кривая.
    """
    if len(raw_list) == 1:
        t_vals, tg_rho_vals, tg_exp_vals = prepare_tg_dataset(raw_list[0], FILM_EPSILON)

        x_rho, y_rho = smooth_curve(t_vals, tg_rho_vals)
        x_exp, y_exp = smooth_curve(t_vals, tg_exp_vals)

        ax.plot(x_rho, y_rho, "b-", linewidth=2.5, label="через  $\rho$ ")
        ax.plot(x_exp, y_exp, "r-", linewidth=2.5, label="эксперимент")

        ax.errorbar(
            t_vals,
            tg_rho_vals,
            yerr=np.abs(np.array(tg_rho_vals)) * ERROR_TG_RHO_PERCENT / 100,
            fmt="D",
            color="blue",
            markersize=8,
            capsize=5,
            elinewidth=1.5,
            capthick=1.5,
        )
        ax.errorbar(
            t_vals,
            tg_exp_vals,
            yerr=np.abs(np.array(tg_exp_vals)) * ERROR_TG_EXP_PERCENT / 100,
            fmt="^",
            color="red",
            markersize=8,
            capsize=5,
            elinewidth=1.5,
            capthick=1.5,
        )

        annotate_series(ax, t_vals, tg_rho_vals, "{:.2f}", 12)
        annotate_series(ax, t_vals, tg_exp_vals, "{:.2f}", -15)
        title = f"{group_title}  $\text{tg}\delta$ "
    else:
        processed = [prepare_tg_dataset(raw, FILM_EPSILON) for raw in raw_list]
        x_common, tg_rho_mean, tg_exp_mean, t_ref, tg_rho_points, tg_exp_points = average_tg_processed(processed)

```

```

ax.plot(x_common, tg_rho_mean, "b-", linewidth=2.5, label="через  $\rho$ ")
ax.plot(x_common, tg_exp_mean, "r-", linewidth=2.5, label="эксперимент")

ax.errorbar(
    t_ref,
    tg_rho_points,
    yerr=np.abs(np.array(tg_rho_points)) * ERROR_TG_RHO_PERCENT / 100,
    fmt="D",
    color="blue",
    markersize=9,
    capsize=6,
    elinewidth=1.8,
    capthick=1.8,
)
ax.errorbar(
    t_ref,
    tg_exp_points,
    yerr=np.abs(np.array(tg_exp_points)) * ERROR_TG_EXP_PERCENT / 100,
    fmt="^",
    color="red",
    markersize=9,
    capsize=6,
    elinewidth=1.8,
    capthick=1.8,
)

annotate_series(ax, t_ref, tg_rho_points, "{:.2f}", 12)
annotate_series(ax, t_ref, tg_exp_points, "{:.2f}", -15)
title = f"{group_title} tg $\delta$ "

style_axes(ax, title, "Тангенс угла диэлектрических потерь tg $\delta$ ")
# =====
# 10. ОСНОВНАЯ ПРОГРАММА
# =====
def add_figure_to_document(document, figure, caption):
    """Сохраняет построенный график в Word-документ."""
    if document is None:
        return
    image_buffer = BytesIO()
    figure.savefig(image_buffer, format="png", dpi=300, bbox_inches="tight")
    image_buffer.seek(0)
    document.add_paragraph(caption)
    document.add_picture(image_buffer, width=Inches(6.5))

def main():
    """
    Последовательно выводит отдельные графики  $\epsilon(T)$  и tg $\delta(T)$  для всех групп образцов.
    """
    configure_publication_style()
    output_dir = Path("figures_for_article")
    output_dir.mkdir(exist_ok=True)

    document = None
    if DOCX_AVAILABLE:
        document = Document()
        document.add_heading("Графики диэлектрической проницаемости и тангенса потерь", level=1)
    else:
        print("Библиотека python-docx не установлена: Word-файл создан не будет.")
        print("Графики будут показаны на экране и сохранены как PNG в папку figures_for_article.")

    for group_title, raw_list in product_groups.items():
        fig, ax = plt.subplots(figsize=(9.0, 6.0))

```

```

plot_eps_group(ax, group_title, raw_list)
fig.tight_layout()
add_figure_to_document(document, fig, f"{group_title}. График  $\varepsilon(T)$ .")
fig.savefig(output_dir / f"{group_title}_eps.png", dpi=300, bbox_inches="tight")

fig, ax = plt.subplots(figsize=(9.0, 6.0))
plot_tg_group(ax, group_title, raw_list)
fig.tight_layout()
add_figure_to_document(document, fig, f"{group_title}. График  $\operatorname{tg}\delta(T)$ .")
fig.savefig(output_dir / f"{group_title}_tg.png", dpi=300, bbox_inches="tight")

if document is not None:
    document.save("Графики_статьи.docx")
    print("Word-файл сохранен: Графики_статьи.docx")

plt.show()

if __name__ == "__main__":
    main()

```