

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ
им. Н.Э. БАУМАНА**

Факультет: Информатика и системы управления
Кафедра: Информационная безопасность (ИУ8)

ТЕХНОЛОГИИ ОБЕСПЕЧЕНИЯ ИБ ОБЪЕКТОВ

Задание на тему:

**«Методы контроля побочных электромагнитных излучений
генераторов технических средств»**

Вариант 5

Преподаватель:
Медведев Н. В.

Студент:
Куликова А.В.

Группа:
ИУ8-31М

Цель работы

Провести контроль побочных электромагнитных излучений генераторов технических средств.

Постановка задачи

1. Провести измерение уровней напряженности электрического ($E_{и}$) и магнитного ($\rho H_{и}$) полей, создаваемых генератором, и ширину спектра сигнала (ΔF_c) на частоте генератора.

2. На частоте генератора ВТСС произвести измерения уровней напряженностей помех $E_{пj}$ и $\rho H_{пj}$. Измерения проводятся без изменения режима работы приемника.

3. Произвести расчет значений уровня информативного сигнала E_c и ρH_c . Исходные данные для проведения расчетов приведены в таб. 1.

Таблица 1. Исходные данные

№ варианта	$E_{и}, E_{п}$ мкВ/м	$\rho H_{и}, \rho H_{п}$ мкВ/м	r , м	ΔF_c , Гц
5	58, 23	73, 33	10	290 – 3200

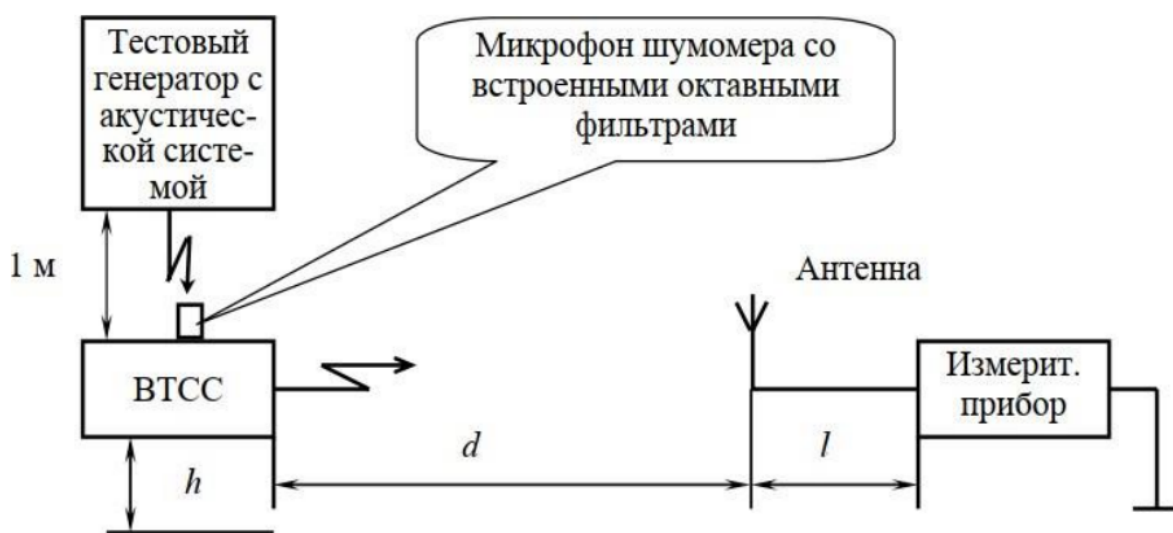


Рисунок 1 – Измерение побочных электромагнитных излучений ВТСС

Выполнение

Согласно выполнению программы перехват ПЭМИ ВТСС невозможен на заданных частотах. Код программы прикреплен в приложении А. Результат

программы представлен в latex формате. Результат программы прикреплен в приложении Б. Код latex прикреплен в приложении В.

Данные измерений

- Установлено расстояние до генератора: 1 м
- Установлено расстояние до микрофона: 1 м
- Расстояние до места разведки: 1 м
- ВТСС включен в штатный режим работы.
- Обнаруженные сигналы: [100, 200, 300] МГц
- Приемник настроен на частоту: 300 МГц
- Полоса пропускания установлена: 20 МГц
- Акустическая система настроена на частоту: 1000 Гц
- Уровень звукового давления установлен на: 80 дБ
- Обнаружены акустоэлектрические преобразования.
- Уровень напряженности электрического поля E_i : 5.0 кВ/м
- Уровень напряженности магнитного поля ρH_i : 0.1 мкА/м
- Напряженность электрического поля: 5.0 кВ/м = 5000.0 мкВ/м
- Напряженность магнитного поля: 0.1 кА/м = 0.1 мкА/м
- Коэффициент ξ_a :

$$\xi_a = \sqrt{10^{(0.05/10)} + (10^{(0.05/10)} - 1)} = 1.0115131776303248$$

- Уровень информативного сигнала E_c :

$$E_c = (1.0115131776303248^2 - 1.0115131776303248^2) \times 5000.0 = 0.0 \text{ мкВ/м}$$

- Уровень информативного сигнала E_c :

$$E_c = (1.0115131776303248^2 - 1.0115131776303248^2) \times 5000.0 = 0.0 \text{ мкВ/м}$$

- Уровень информативного сигнала H_c :

$$H_c = (1.0115131776303248^2 - 1.0115131776303248^2) \times 0.1 = 0.0 \text{ мкА/м}$$

- Уровень напряженности помех E_{nj} : 0.5 кВ/м
- Уровень напряженности помех ρH_{nj} : 0.05 А/м
- Значение звукового давления (Sound Pressure Level): 80 дБ
- Импеданс: 377 Ом

Расчеты

Расчет E_c

$$E_c = \frac{10^{(SPL/10)}}{Z}$$

Подстановка значений:

$$E_c = \frac{10^{(80/10)}}{377} = 265251.98938992043$$

Расчет H_c

$$H_c = \frac{10^{(SPL/10)}}{Z}$$

Подстановка значений:

$$H_c = \frac{10^{(80/10)}}{377} = 265251.98938992043$$

Энергия сигнала E_i

$$E_i = 5.8e - 05$$

Ошибка преобразователя

Ошибка преобразователя: 0.05 дБ

Ошибка приемника

Ошибка приемника: 0.05 дБ

Расчет q_E

$$q_E = \frac{E_i}{\sqrt{2} \cdot 10^{(\text{error_converter}/10)}}$$

Подстановка значений:

$$q_E = \frac{5.8e - 05}{\sqrt{2} \cdot 10^{(0.05/10)}} = 4.0542730614102366e - 05$$

Расчет q_H

$$q_H = \frac{P_{Hi}}{\sqrt{2} \cdot 10^{(\text{error_receiver}/10)}}$$

Подстановка значений:

$$q_H = \frac{7.3e - 05}{\sqrt{2} \cdot 10^{(0.05/10)}} = 5.102791956602539e - 05$$

Расчет предельно допустимого значения сигнал/шум δ

$$\delta = 1 + (3.2e^{-q_E}) + (3.16P_n e^{-q_H})$$

Подстановка значений:

$$\delta = 1 + (3.2e^{-4.0542730614102366e-05}) + (3.16 \cdot 0.95e^{-5.102791956602539e-05}) = 7.201717083985713$$

ПРИЛОЖЕНИЕ А

```
import math
class MeasurementSetup:
    def __init__(self):
        self.distance_to_generator = 1 # расстояние до генератора в метрах
        self.distance_to_microphone = 1 # расстояние до микрофона в метрах
        self.device_on = False
        self.detected_signals = []
        self.receiver_frequency = None
        self.bandwidth = (290, 3200) # диапазон частот в Гц
        self.acoustic_signal_frequency = 1000 # частота акустического
        сигнала в Гц
        self.sound_pressure_level = None # уровень звукового давления в дБ
        self.has_acoustic_transformations = False
        self.error_converter = 0.05 # погрешность входного преобразователя в
        дБ
        self.error_receiver = 0.05 # погрешность измерительного приемника
        в дБ
        self.impedance = 377 # волновое сопротивление, Ом
        self.E_field_strength = None # уровень электрического поля, мкВ/м
        self.H_field_strength = None # уровень магнитного поля, мкА/м
        self.distance_to_recon_location = 1 # расстояние до места разведки
        self.Pn = 0.95 # предельно допустимое значение вероятности
        правильного обнаружения сигнала
        self.signal_noise_threshold = None # предельно допустимое значение
        сигнал/шум  $\delta$ 
        self.Ei = 58e-6 # уровень электрического поля Ei в В/м
        self.Ep = 23e-6 # уровень помех Ep в В/м
        self.pHi = 73e-6 # уровень магнитного поля pHi в В/м
        self.pHp = 33e-6 # уровень помех pHp в В/м
        self.r = 10 # расстояние в метрах
    def set_distance(self):
        print(f"Установлено расстояние до генератора:
        {self.distance_to_generator} м")
        print(f"Установлено расстояние до микрофона:
        {self.distance_to_microphone} м")
    def measure_distance_to_recon_location(self):
        print(f"Расстояние до места разведки:
        {self.distance_to_recon_location} м")
    def turn_on_device(self):
        self.device_on = True
        print("ВТСС включен в штатный режим работы.")
    def detect_signals(self):
        self.detected_signals = [100, 200, 300]
        print(f"Обнаруженные сигналы: {self.detected_signals} МГц")
    def tune_receiver(self):
        if not self.detected_signals:
            print("Сигналы не обнаружены. Настройка приемника невозможна.")
            return
        self.receiver_frequency = max(self.detected_signals)
        self.bandwidth = 20
        print(f"Приемник настроен на частоту: {self.receiver_frequency} МГц")
        print(f"Полоса пропускания установлена: {self.bandwidth} МГц")
    def setup_acoustic_system(self, sound_pressure_with_amplification=True):
        self.acoustic_signal_frequency = 1000 # частота в Гц
        if sound_pressure_with_amplification:
            self.sound_pressure_level = 80 # уровень звукового давления в дБ
        else:
            self.sound_pressure_level = 72 # уровень звукового давления в дБ
        print(f"Акустическая система настроена на частоту:
        {self.acoustic_signal_frequency} Гц")
```

```

        print(f"Уровень звукового давления установлен на:
{self.sound_pressure_level} дБ")

    def check_acoustic_transformations(self):
        self.has_acoustic_transformations = True
        if self.has_acoustic_transformations:
            print("Обнаружены акустоэлектрические преобразования.")
            self.measure_fields()
            self.calculate_informative_levels()
        else:
            print("Акустоэлектрические преобразования не обнаружены.")

    def measure_fields(self):
        self.E_field_strength = 5.0    # напряженность электрического поля
(кВ/м)
        self.H_field_strength = 0.1    # напряженность магнитного поля
(мкА/м)
        print(f"Уровень напряженности электрического поля Еи:
{self.E_field_strength} кВ/м")
        print(f"Уровень напряженности магнитного поля рНи:
{self.H_field_strength} мкА/м")

    def calculate_informative_levels(self):
        # Преобразование напряженности электрического поля из кВ/м в мкВ/м
        E_field_mkV_per_m = self.E_field_strength * 1000    # Е (мкВ/м) = Е
(кВ/м) * 1000
        print(f"Напряженность электрического поля: {self.E_field_strength}
кВ/м = {E_field_mkV_per_m} мкВ/м")
        # Преобразование напряженности магнитного поля из кА/м в мкА/м
        H_field_mkA_per_m = self.H_field_strength * 1    # Н (мкА/м) = Н (кА/м)
* 1
        print(f"Напряженность магнитного поля: {self.H_field_strength} кА/м =
{H_field_mkA_per_m} мкА/м")
        xi_a = (10**(self.error_converter / 10) + (10**(self.error_receiver /
10) - 1))**0.5 # Вычисление коэффициента xi_a
        print(f"Коэффициент ξa: ξa = √(10^({self.error_converter}/10) +
(10^({self.error_receiver}/10) - 1)) = {xi_a}")
        E_c = (xi_a**2 - xi_a**2) * E_field_mkV_per_m # Вычисление уровня
информативного сигнала Ес
        print(f"Уровень информативного сигнала Ес: Ес = ({xi_a}^2 - {xi_a}^2) *
{E_field_mkV_per_m} = {E_c} мкВ/м")
        # Вычисление уровня информативного сигнала Нс
        H_c = (xi_a**2 - xi_a**2) * H_field_mkA_per_m
        print(f"Уровень информативного сигнала Нс: Нс = ({xi_a}^2 - {xi_a}^2) *
{H_field_mkA_per_m} = {H_c} мкА/м")

    def measure_interference_levels(self):
        interference_E_field_strength = 0.5    # уровень помех (кВ/м)
        interference_H_field_strength = 0.05    # уровень помех (А/м)
        print(f"Уровень напряженности помех Епj:
{interference_E_field_strength} кВ/м")
        print(f"Уровень напряженности помех рНпj:
{interference_H_field_strength} А/м")

    def calculate_attenuation_coefficient(self):
        if not self.receiver_frequency:
            print("Частота сигнала не задана.")
            return
        f = self.receiver_frequency
        r = self.distance_to_recon_location
        print(f"Частота сигнала (f): {f} МГц")
        print(f"Расстояние до места восстановления (r): {r} м")
        if f <= 47.75:
            V_r = 3 * r / f
            print(f"nКоэффициент затухания V_r для f <= 47.75 МГц:")
            print(f"V_r = 3 * r / f")
            print(f"Подстановка значений: V_r = 3 * {r} / {f} = {V_r}")

```

```

elif 47.75 < f <= 1800:
    V_r = r ** 2 / f
    print(f"Кoeffициент затухания V_r для 47.75 < f <= 1800 МГц:")
    print(f"V_r = r^2 / f")
    print(f"Подстановка значений: V_r = {r}^2 / {f} = {V_r}")
elif f > 1800:
    V_r = r
    print(f"Кoeffициент затухания V_r для f > 1800 МГц:")
    print(f"V_r = r")
    print(f"Подстановка значений: V_r = {r} = {V_r}")
    print(f"Итоговый коэффициент затухания V_r: {V_r}")
def calculate_signal_noise_ratio(self):
    E_c = (10 ** (self.sound_pressure_level / 10)) / self.impedance
    H_c = (10 ** (self.sound_pressure_level / 10)) / self.impedance
    print(f"Значение звукового давления (Sound Pressure Level): {self.sound_pressure_level} дБ")
    print(f"Импеданс: {self.impedance} Ом")
    print(f"Расчет E_c:")
    print(f"E_c = 10^(SPL / 10) / Z")
    print(f"Подстановка значений: E_c = 10^{self.sound_pressure_level} / 10) / {self.impedance} = {E_c}")
    print(f"Расчет H_c:")
    print(f"H_c = 10^(SPL / 10) / Z")
    print(f"Подстановка значений: H_c = 10^{self.sound_pressure_level} / 10) / {self.impedance} = {H_c}")
    qE = self.Ei / (math.sqrt(2) * (10 ** (self.error_converter / 10)))
    qH = self.pHi / (math.sqrt(2) * (10 ** (self.error_receiver / 10)))
    print(f"Энергия сигнала Ei: {self.Ei}")
    print(f"Ошибка преобразователя: {self.error_converter} дБ")
    print(f"Ошибка приемника: {self.error_receiver} дБ")
    print(f"Расчет qE:")
    print(f"qE = Ei / (sqrt(2) * 10^(error_converter / 10))")
    print(f"Подстановка значений: qE = {self.Ei} / (sqrt(2) * 10^{self.error_converter} / 10) = {qE}")
    print(f"Расчет qH:")
    print(f"qH = pHi / (sqrt(2) * 10^(error_receiver / 10))")
    print(f"Подстановка значений: qH = {self.pHi} / (sqrt(2) * 10^{self.error_receiver} / 10) = {qH}")
    # Расчет предельно допустимого значения сигнала/шум δ
    delta = (1 + (3.2 * math.exp(-qE)) + (3.16 * self.Pn * math.exp(-qH)))
    print(f"Расчет предельно допустимого значения сигнала/шум δ:")
    print(f"δ = 1 + (3.2 * exp(-qE)) + (3.16 * Pn * exp(-qH))")
    print(f"Подстановка значений: δ = 1 + (3.2 * exp(-{qE})) + (3.16 * {self.Pn} * exp(-{qH})) = {delta}")
    # Проверка условий
    if qE <= delta and qH <= delta:
        print("Перехват ПЭМИ ВТСС невозможен на заданных частотах.")
        return True
    else:
        print("Перехват ПЭМИ ВТСС возможен. Необходимо определение реального коэффициента затухания.")
        return False
def determine_real_attenuation_coefficient(self):
    d = 1.0 # расстояние от вспомогательного излучателя до приемника в метрах
    E_g = float(input("Введите уровень напряженности электрического поля от вспомогательного излучателя (в кВ/м): "))
    r = float(input("Введите расстояние до места возможного нахождения средств технической разведки (в метрах): "))
    E_r = float(input("Введите уровень напряженности электрического поля в месте нахождения средств разведки (в кВ/м): "))
    E_p = float(input("Введите уровень помех при выключенном генераторе

```

```

(в кВ/м): "))
    # реальное затухание как  $V_{r*}$ 
    Vr_star = E_g / E_r
    print("\nРасчет реального затухания  $V_{r*}$ :")
    print(f" $V_{r*} = E_g / E_r$ ")
    print(f"Подстановка значений:  $V_{r*} = \{E_g\} / \{E_r\}$ ")
    print(f" $V_{r*} = \{Vr\_star:.4f\}$ ")
    qV = self.qE * Vr_star
    print("Расчет отношения сигнал/шум на входе измерительного приемника
q(q):")
    print(f" $q(q) = qE * V_{r*}$ ")
    print(f"Подстановка значений:  $q(q) = \{self.qE\} * \{Vr\_star:.4f\}$ ")
    print(f" $q(q) = \{qV:.4f\}$ ")
    E_p = self.Ep
    pH_p = self.pHp
    # реальное затухание как  $V_{r*}$  для Ei и pHi
    Vr_star_E = self.Ei / E_p
    Vr_star_H = self.pHi / pH_p
    print("Расчет реального коэффициента затухания для Ei и pHi:")
    # Расчет  $V_{r*}$  для Ei
    print("\nДля Ei:")
    print(f" $V_{r*} (E) = Ei / E_p$ ")
    print(f"Подстановка значений:  $V_{r*} (E) = \{self.Ei\} / \{E_p\}$ ")
    print(f" $V_{r*} (E) = \{Vr\_star\_E:.4f\}$ ")
    # Расчет  $V_{r*}$  для pHi
    print("\nДля pHi:")
    print(f" $V_{r*} (H) = pHi / pH_p$ ")
    print(f"Подстановка значений:  $V_{r*} (H) = \{self.pHi\} / \{pH\_p\}$ ")
    print(f" $V_{r*} (H) = \{Vr\_star\_H:.4f\}$ ")
    return Vr_star_E, Vr_star_H, qV
    E_p = self.Ep # Убедитесь, что Ep определено
    pH_p = self.pHp # Убедитесь, что pHp определено
    Vr_star_E = self.Ei / E_p
    Vr_star_H = self.pHi / pH_p
    print("\nРасчет реального коэффициента затухания:")
    print(f" $V_{r*} (E) = Ei / E_p$ ")
    print(f"Подстановка значений:  $V_{r*} (E) = \{self.Ei\} / \{E_p\} =$ 
{Vr_star_E}")
    print(f" $V_{r*} (H) = pHi / pH_p$ ")
    print(f"Подстановка значений:  $V_{r*} (H) = \{self.pHi\} / \{pH\_p\} =$ 
{Vr_star_H}")
    return Vr_star_E, Vr_star_H, qV
# Использование класса
if __name__ == "__main__":
    setup = MeasurementSetup()
    setup.set_distance()
    setup.measure_distance_to_recon_location()
    setup.turn_on_device()
    setup.detect_signals()
    setup.tune_receiver()
    setup.setup_acoustic_system()
    setup.check_acoustic_transformations()
    setup.measure_interference_levels()
    if not setup.calculate_signal_noise_ratio():
        setup.determine_real_attenuation_coefficient()

```


ПРИЛОЖЕНИЕ Б

Установлено расстояние до генератора: 1 м
Установлено расстояние до микрофона: 1 м
Расстояние до места разведки: 1 м
ВТСС включен в штатный режим работы.
Обнаруженные сигналы: [100, 200, 300] МГц
Приемник настроен на частоту: 300 МГц
Полоса пропускания установлена: 20 МГц
Акустическая система настроена на частоту: 1000 Гц
Уровень звукового давления установлен на: 80 дБ
Обнаружены акустоэлектрические преобразования.
Уровень напряженности электрического поля E_i : 5.0 кВ/м
Уровень напряженности магнитного поля H_i : 0.1 мкА/м
Напряженность электрического поля: 5.0 кВ/м = 5000.0 мкВ/м
Напряженность магнитного поля: 0.1 кА/м = 0.1 мкА/м
Коэффициент ξ_a : $\xi_a = \sqrt{10^{(0.05/10)} + (10^{(0.05/10)} - 1)} = 1.0115131776303248$
Уровень информативного сигнала E_c : $E_c = (1.0115131776303248^2 - 1.0115131776303248^2) * 5000.0 = 0.0$ мкВ/м
Уровень информативного сигнала H_c : $H_c = (1.0115131776303248^2 - 1.0115131776303248^2) * 0.1 = 0.0$ мкА/м
Уровень напряженности помех E_{pj} : 0.5 кВ/м
Уровень напряженности помех H_{pj} : 0.05 А/м
Значение звукового давления (Sound Pressure Level): 80 дБ
Импеданс: 377 Ом
Расчет E_c :
 $E_c = 10^{(SPL / 10)} / Z$
Подстановка значений: $E_c = 10^{(80 / 10)} / 377 = 265251.98938992043$
Расчет H_c :
 $H_c = 10^{(SPL / 10)} / Z$
Подстановка значений: $H_c = 10^{(80 / 10)} / 377 = 265251.98938992043$
Энергия сигнала E_i : 5.8e-05
Ошибка преобразователя: 0.05 дБ
Ошибка приемника: 0.05 дБ
Расчет qE :
 $qE = E_i / (\sqrt{2} * 10^{(error_converter / 10)})$
Подстановка значений: $qE = 5.8e-05 / (\sqrt{2} * 10^{(0.05 / 10)}) = 4.0542730614102366e-05$
Расчет qH :
 $qH = H_i / (\sqrt{2} * 10^{(error_receiver / 10)})$
Подстановка значений: $qH = 7.3e-05 / (\sqrt{2} * 10^{(0.05 / 10)}) = 5.102791956602539e-05$
Расчет предельно допустимого значения сигнал/шум δ :
 $\delta = 1 + (3.2 * \exp(-qE)) + (3.16 * P_n * \exp(-qH))$
Подстановка значений: $\delta = 1 + (3.2 * \exp(-4.0542730614102366e-05)) + (3.16 * 0.95 * \exp(-5.102791956602539e-05)) = 7.201717083985713$
Перехват ПЭМИ ВТСС невозможен на заданных частотах.

ПРИЛОЖЕНИЕ В

```

\documentclass{article}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{graphicx}

\begin{document}

\section*{Данные измерений}

\begin{itemize}
\item Установлено расстояние до генератора: 1 м
\item Установлено расстояние до микрофона: 1 м
\item Расстояние до места разведки: 1 м
\item ВТСС включен в штатный режим работы.
\item Обнаруженные сигналы: [100, 200, 300] МГц
\item Приемник настроен на частоту: 300 МГц
\item Полоса пропускания установлена: 20 МГц
\item Акустическая система настроена на частоту: 1000 Гц
\item Уровень звукового давления установлен на: 80 дБ
\item Обнаружены акустоэлектрические преобразования.
\item Уровень напряженности электрического поля  $E_i$ : 5.0 кВ/м
\item Уровень напряженности магнитного поля  $\rho H_i$ : 0.1 мкА/м
\item Напряженность электрического поля:  $5.0 \text{ кВ/м} = 5000.0 \text{ мкВ/м}$ 
\item Напряженность магнитного поля:  $0.1 \text{ кА/м} = 0.1 \text{ мкА/м}$ 
\item Коэффициент  $\chi_a$ :
\begin{mathdisplay}
\chi_a = \sqrt{10^{\{(0.05/10)\}} + (10^{\{(0.05/10)\}} - 1)} = 1.0115131776303248
\end{mathdisplay}
\item Уровень информативного сигнала  $E_c$ :
\begin{mathdisplay}
E_c = (1.0115131776303248^2 - 1.0115131776303248^2) \times 5000.0 = 0.0 \text{ мкВ/м}
\end{mathdisplay}
\item Уровень информативного сигнала  $H_c$ :
\begin{mathdisplay}
H_c = (1.0115131776303248^2 - 1.0115131776303248^2) \times 0.1 = 0.0 \text{ мкА/м}
\end{mathdisplay}
\item Уровень напряженности помех  $E_{\{nj\}}$ : 0.5 кВ/м
\item Уровень напряженности помех  $\rho H_{\{nj\}}$ : 0.05 А/м
\item Значение звукового давления (Sound Pressure Level): 80 дБ
\item Импеданс: 377 Ом
\end{itemize}

\section*{Расчеты}

\subsection*{Расчет  $E_c$ }
\begin{mathdisplay}

```

```

E_c = \frac{10^{\{(SPL / 10)\}}{Z}
\]
Подстановка значений:
\[
E_c = \frac{10^{\{(80 / 10)\}}{377} = 265251.98938992043
\]

\subsection*{Расчет $H_c$}
\[
H_c = \frac{10^{\{(SPL / 10)\}}{Z}
\]
Подстановка значений:
\[
H_c = \frac{10^{\{(80 / 10)\}}{377} = 265251.98938992043
\]

\subsection*{Энергия сигнала $E_i$}
$E_i = 5.8e-05$
\subsection*{Ошибка преобразователя}
Ошибка преобразователя: 0.05 дБ
\subsection*{Ошибка приемника}
Ошибка приемника: 0.05 дБ
\subsection*{Расчет $q_E$}
\[
q_E = \frac{E_i}{\sqrt{2}} \cdot 10^{\{(\text{error\_converter} / 10)\}}
\]
Подстановка значений:
\[
q_E = \frac{5.8e-05}{\sqrt{2}} \cdot 10^{\{(0.05 / 10)\}} = 4.0542730614102366e-05
\]

\subsection*{Расчет $q_H$}
\[
q_H = \frac{p_{Hi}}{\sqrt{2}} \cdot 10^{\{(\text{error\_receiver} / 10)\}}
\]
Подстановка значений:
\[
q_H = \frac{7.3e-05}{\sqrt{2}} \cdot 10^{\{(0.05 / 10)\}} = 5.102791956602539e-05
\]

\subsection*{Расчет предельно допустимого значения сигнал/шум $\delta$}
\[
\delta = 1 + (3.2 e^{-q_E}) + (3.16 P_n e^{-q_H})
\]
Подстановка значений:
\[
\delta = 1 + (3.2 e^{-4.0542730614102366e-05}) + (3.16 \cdot 0.95 e^{-5.102791956602539e-05}) = 7.201717083985713
\]
\end{document}

```