

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

Дисциплина «Системы ввода-вывода»

Лабораторная работа №3

Авторы: Глушкова Лис Григорьевна

Гуменник Петр Олегович

Руотси Ян Евгеньевич

Факультет: ПИиКТ

Поток: 1.4

Преподаватель: Быковский Сергей Вячеславович

**ИТМО**

Санкт-Петербург, 2025

## Цель

Познакомится с принципами обмена данными между устройствами, алгоритмами обмена и форматами передачи данных на примере интерфейсов I2C, SPI, 1-Wire.

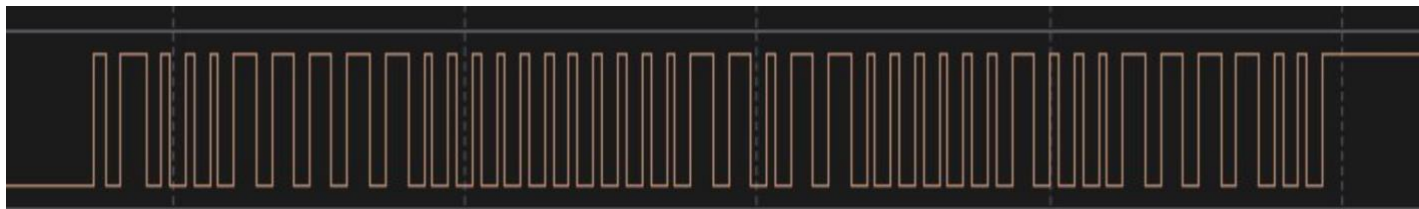
## Задачи работы

1. Подключить комплект с контроллер с датчиком и логическим анализатором к компьютеру
2. С помощью логического анализатора записать временную диаграмму обмена данными по сигнальным линиям в течении трех транзакций обмена.
3. Расшифровать протокол обмена данными.
4. Перевести значение физической величины, заданной в варианте задания, в человекочитаемый формат.
5. Нарисовать временную диаграмму передачи другого, отличного от полученных, значения физической величины.
6. Определить скорость интерфейса.

## Основная часть

### 1. Работа с датчиком DHT-11

Получаем данные с датчика:



Наша последовательность имеет вид:

0001 1111 0000 0000 0001 1011 0000 0010 0011 1100

Пример расшифровки:

<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1101</u>
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate :

$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$

Received data is correct:

Humidity:  $0011\ 0101 = 35H = 53\%RH$

Temperature:  $0001\ 1000 = 18H = 24^{\circ}C$

То есть мы получаем:

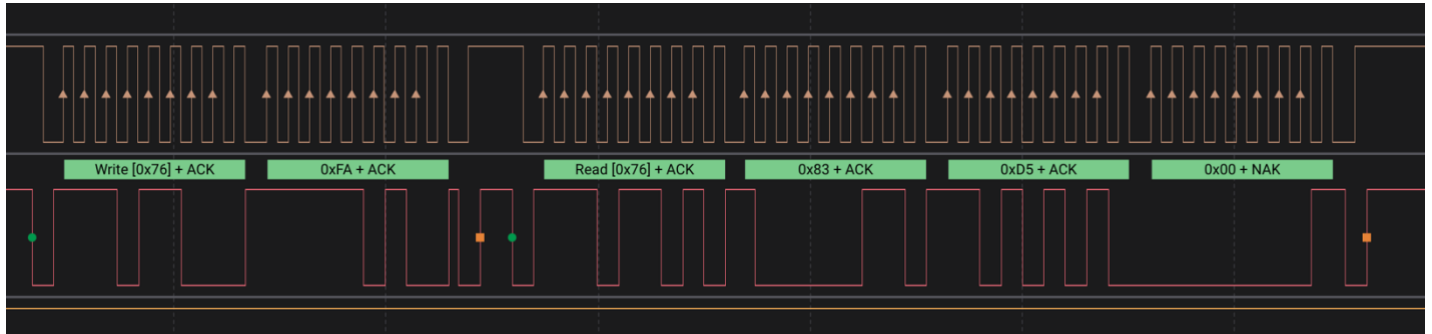
0001 1111 0000 0000 – влажность 31.0 %

0001 1011 0000 0010 – температура 27.008 °C

0001 1111 + 0000 0000 + 0001 1011 + 0000 0010 = 0011 1100 – четность ОК

## 2. Работа с датчиком BMP280

Получаем данные с датчика:



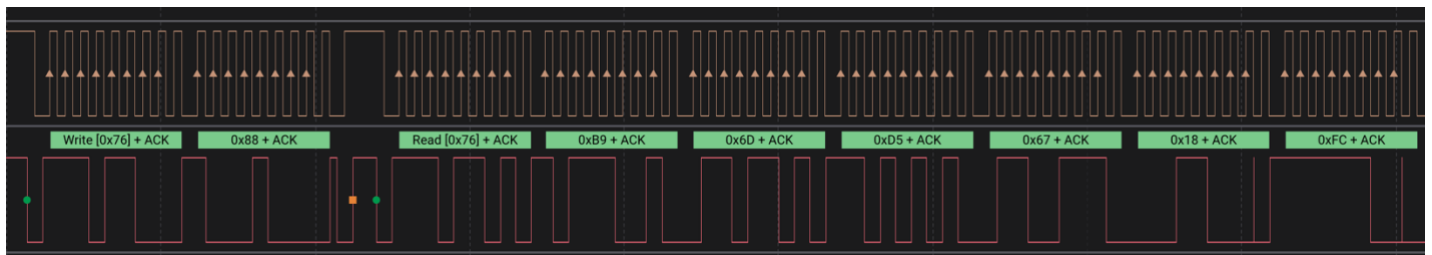
Данные температуры находятся по адресам 0xFA...0xFC. Получаем:

temp\_msb = 0x83

temp\_lsb = 0xD5

temp\_xlsb = 0x00

Далее с помощью калибровочных данных найдем фактическую температуру:



Калибровочные данные находятся по адресам 0x88...0xA1. Так как нам нужны только для температуры берем с 0x88 до 0x8D. Получаем:

dig\_T1 0xB9 0x6D

dig\_T2 0xD5 0x67

dig\_T3 0x18 0xFC

Теперь используя написанную программу, ищем температуру:

```
import struct

temp_msb = 0x83
temp_lsb = 0xD5
temp_xlsb = 0x00

# Калибровочные коэффициенты (в порядке LSB, MSB)
dig_T1 = 0x6DB9 # unsigned short
dig_T2 = 0x67D5 # signed short
dig_T3 = 0xFC18 # signed short

# Декодируем значения из 2-х байтов (little-endian)
def u16(val): # unsigned short
    return val & 0xFFFF

def s16(val): # signed short
    return struct.unpack('<h', struct.pack('<H', val & 0xFFFF))[0]

# Калибровка
dig_T1 = u16(0x6DB9)
dig_T2 = s16(0x67D5)
dig_T3 = s16(0xFC18)

# Сырые данные температуры (20 бит)
adc_T = (temp_msb << 12) | (temp_lsb << 4) | (temp_xlsb >> 4)

# Формула из изображения (плавающая точка)
var1 = ((adc_T / 16384.0) - (dig_T1 / 1024.0)) * dig_T2
var2 = ((adc_T / 131072.0) - (dig_T1 / 8192.0)) ** 2 * dig_T3
t_fine = var1 + var2
T = (var1 + var2) / 5120.0 # Градусы Цельсия

print(f"Температура: {T:.2f} °C")
```

Получаем фактическую температуру 28.6 °C

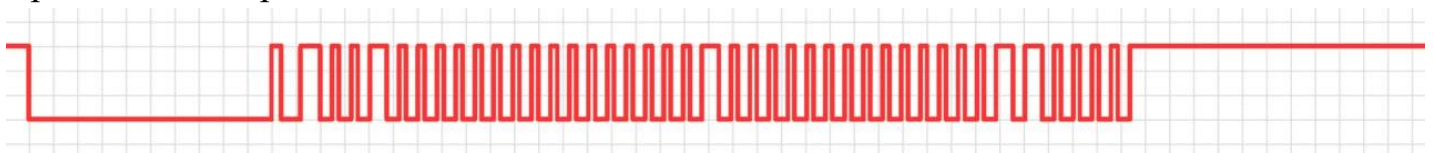
## Обратное преобразование

### 1. Для датчика DHT-11

Возьмем температуру 16 °C и влажность 32 %. Тогда наша последовательность бит будет такой:

0010 0000 0000 0000 0001 0000 0000 0000 0011 0000

Временная диаграмма:



## 2. Для датчика BMP280

Возьмем температуру 24 °C. Напишем программу для нахождения MSB, LSB, XLSB:

```
import struct

# Функции для интерпретации коэффициентов
def u16(val): return val & 0xFFFF
def s16(val): return struct.unpack('<h', struct.pack('<H', val & 0xFFFF))[0]

# Калибровочные коэффициенты (сохраняются те же)
dig_T1 = u16(0x6DB9)
dig_T2 = s16(0x67D5)
dig_T3 = s16(0xFC18)

# Целевая температура
target_temp = 24.0

# Подбор adc_T через перебор
for adc_T in range(0, 1 << 20): # 20 битное значение
    var1 = ((adc_T / 16384.0) - (dig_T1 / 1024.0)) * dig_T2
    var2 = ((adc_T / 131072.0) - (dig_T1 / 8192.0)) ** 2 * dig_T3
    T = (var1 + var2) / 5120.0

    if abs(T - target_temp) < 0.01: # допуск по температуре
        temp_msb = (adc_T >> 12) & 0xFF
        temp_lsb = (adc_T >> 4) & 0xFF
        temp_xlsb = (adc_T & 0xF) << 4 # только старшие 4 бита
        print(f"Температура ≈ {T:.2f} °C")
        print(f"temp_msb = 0x{temp_msb:02X}")
        print(f"temp_lsb = 0x{temp_lsb:02X}")
        print(f"temp_xlsb = 0x{temp_xlsb:02X}")
        break
```

Полученные значения для MSB, LSB, XLSB равны:

temp\_msb = 0x80

temp\_lsb = 0x41

temp\_xlsb = 0xD0

Тогда временная диаграмма будет иметь следующий вид:

