

*Министерство образования Республики Беларусь
Учреждение Образования
«Брестский Государственный Технический Университет»
Кафедра ИИТ*

Лабораторная работа №10

Выполнил:

Студент 3-го курса
Группы АС-59
Сахацкий А.С.

Проверила:

Давидюк Ю.И.

Брест 2023

Цель работы:

приобрести практические навыки проектирования инфраструктуры «умного дома», научиться основам программирования микроконтроллерных устройств

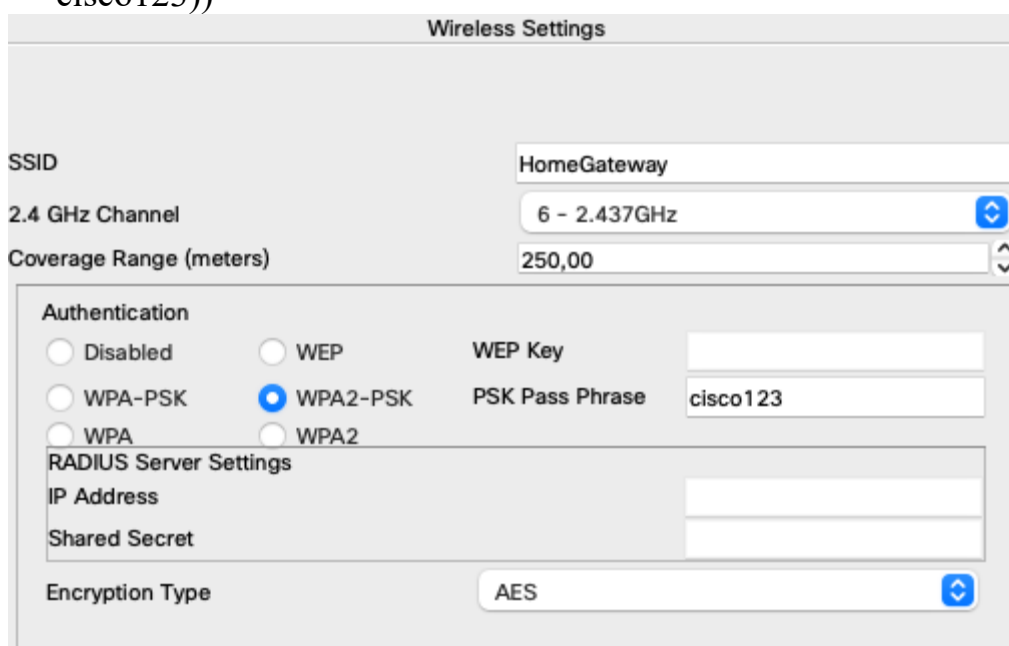
Вариант 8

Задание 1

- 1) Нам надо поставить сервер IoT



- 2) Необходимо настроить его (определить тип аутентификации как WPA-2-PSK и задать пароль из 8 символов (в моем случае cisco123))



The screenshot shows the 'Wireless Settings' page of a router. The SSID is 'HomeGateway'. The 2.4 GHz Channel is set to '6 - 2.437GHz'. The Coverage Range (meters) is '250,00'. Under the 'Authentication' section, 'WPA2-PSK' is selected. The 'PSK Pass Phrase' is 'cisco123'. The 'Encryption Type' is 'AES'. There are also fields for 'WEP Key' and 'RADIUS Server Settings' (IP Address and Shared Secret).

- 3) После настройки надо добавить устройство IoT (лампа, окно, кофеварка, вентилятор, дверь). Эти устройства по умолчанию не поддерживают беспроводную передачу данных. Поэтому мы в Config выберем беспроводной адаптер PI-IOT-NW-1W

Network Adapter 2

PT-IOT-NM-1W

Выполняем для всех устройств IoT.

- 4) После настройки должен появиться интерфейс Wireless3. Откройте его и настройте подключение к серверу, задав правильный тип аутентификации, пароль и выбрав вариант DHCP в IP Configuration (этот вариант чаще всего задан по умолчанию, убедитесь в этом случае, что узлом получен IP-адрес из того же диапазона, что и IP-адрес сервера – как правило, из 192.168.25.0). В данном случае сервер IoT Home Gateway является DHCP-сервером для подключаемых устройств (автоматически раздает IP-адреса).

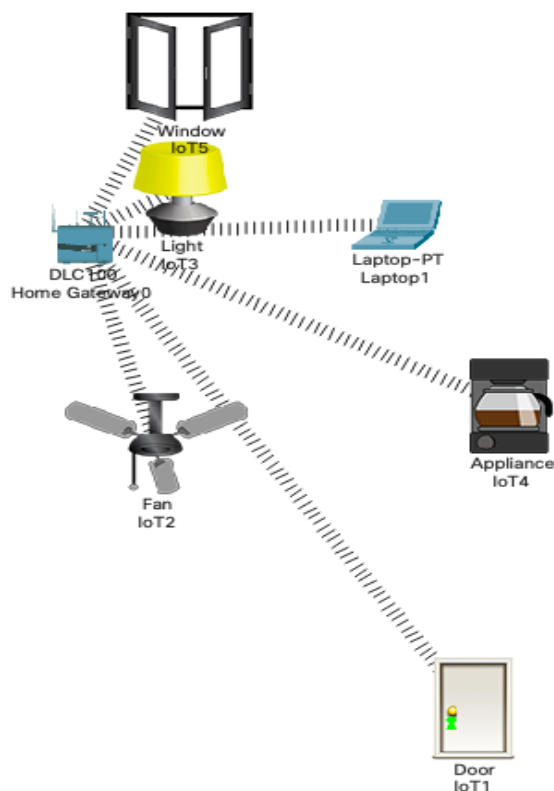
The screenshot shows the 'Wireless3' configuration page. On the left is a sidebar with a tree view containing 'GLOBAL' (with sub-items 'Settings', 'Algorithm Settings', 'Files') and 'INTERFACE' (with sub-items 'Ethernet0', 'Wireless3', 'Bluetooth'). The 'Wireless3' interface is selected. The main configuration area includes: 'Port Status' (checked 'On'), 'Bandwidth' (300 Mbps), 'MAC Address' (0040.0B47.269B), and 'SSID' (HomeGateway). The 'Authentication' section has radio buttons for 'Disabled', 'WPA-PSK', 'WPA', '802.1X', 'WEP', 'WPA2-PSK' (selected), and 'WPA2'. To the right of these are fields for 'WEP Key', 'PSK Pass Phrase' (cisco123), 'User ID', 'Password', 'Method' (MD5), 'User Name', and another 'Password'. The 'Encryption Type' is set to 'AES'. The 'IP Configuration' section has radio buttons for 'DHCP' (selected) and 'Static', with fields for 'IPv4 Address' (192.168.25.109) and 'Subnet Mask' (255.255.255.0).

- 5) В настройках Settings поставим в группе IoT Server переключатель в положение Home Gateway

The screenshot shows the 'IoT Server' settings. It features three radio buttons: 'None', 'Home Gateway' (which is selected), and 'Remote Server'.

- 6) После выполнения этих пунктов убеждаемся что появилась связь между сервером и настраиваемом узле
- 7) Выполняем все эти пункты (со 2 по 5) для всех IoT устройств

Убеждаемся что все устройства подключились (Лэптоп игнорируем, тк его настройки идут дальше)

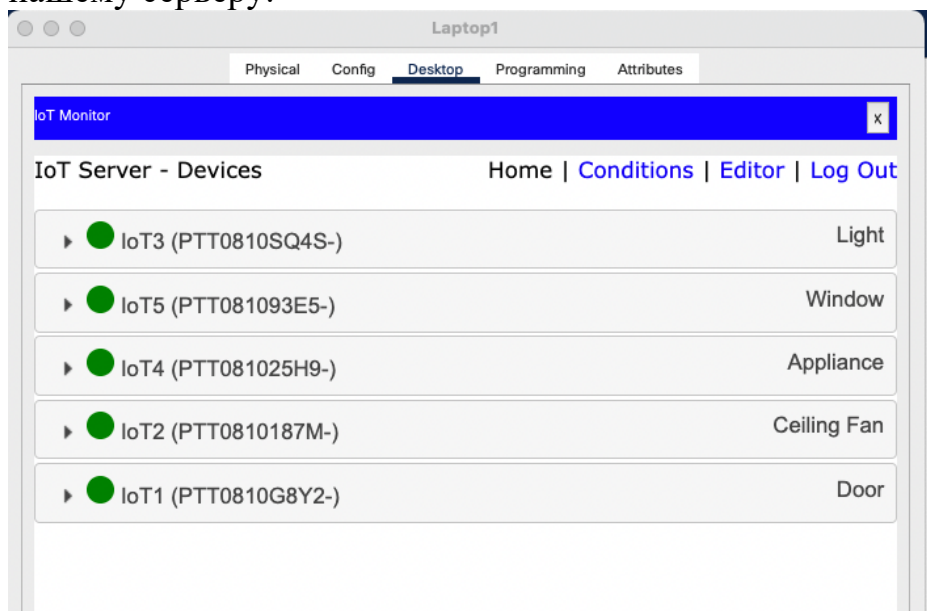


8) Теперь настроим лэптоп

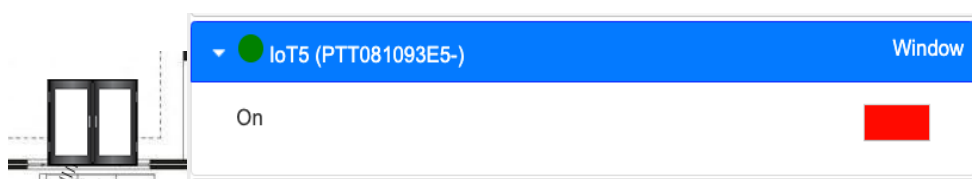
9) Откроем лэптоп и изучим его физическую конфигурацию. Можем заметить, что на нем также, как и на IoT-устройствах не установлен модуль беспроводной связи. Это можно исправить следующим образом: извлечем установленный Fast Ethernet-модуль (предварительно выключив лэптоп) и поместим в свободный слот модуль PT-LAPTOP-NM-1W. После этого включим устройство и произведем похожие настройки беспроводного интерфейса (укажем SSID, тип аутентификации и пароль). Между сервером и лэптопом должна появиться визуализация беспроводной связи.

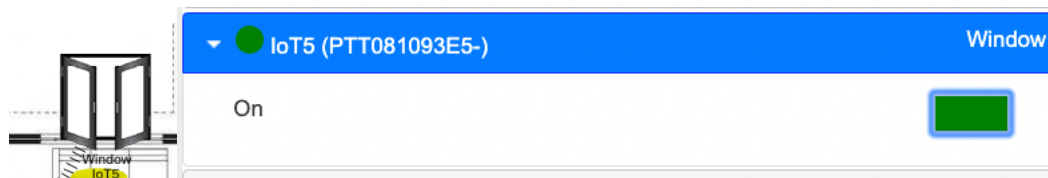


- 10) Откроем вкладку Desktop лэптопа и далее IoT Monitor. Нажмем Ok в окне авторизации на сервере, убедимся в правильности написанного IP-адреса сервера. После этого появиться список всех беспроводных устройств, подключенных к нашему серверу.

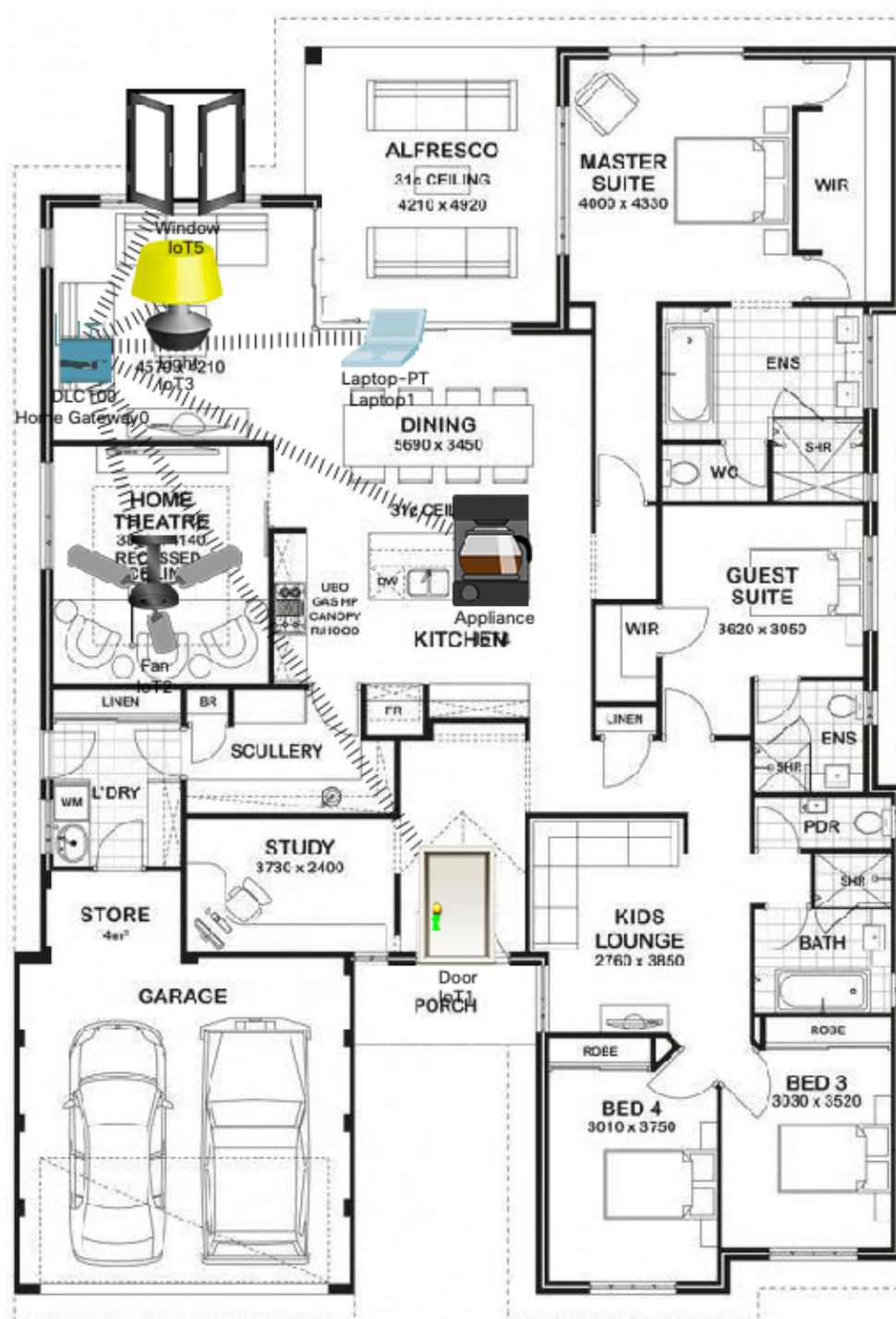


Например: я открою окно



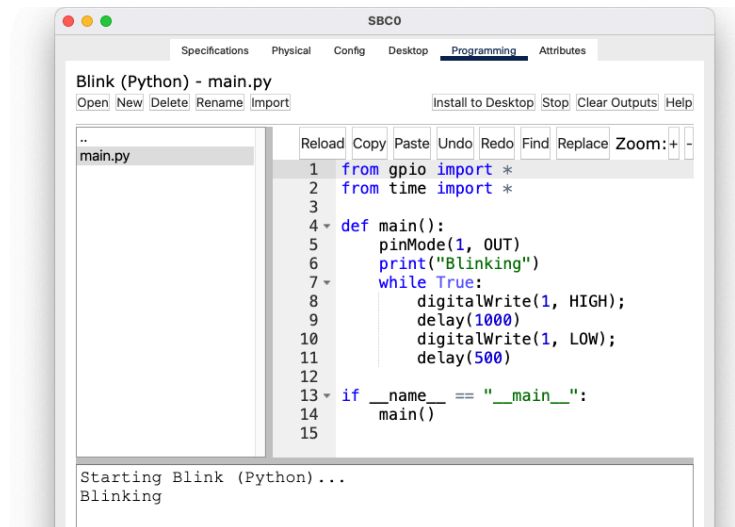
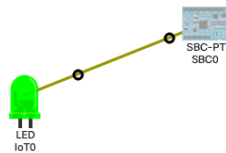


- 11) Добавим фон для построенной инфраструктуры, воспользовавшись предложенными (папка background)

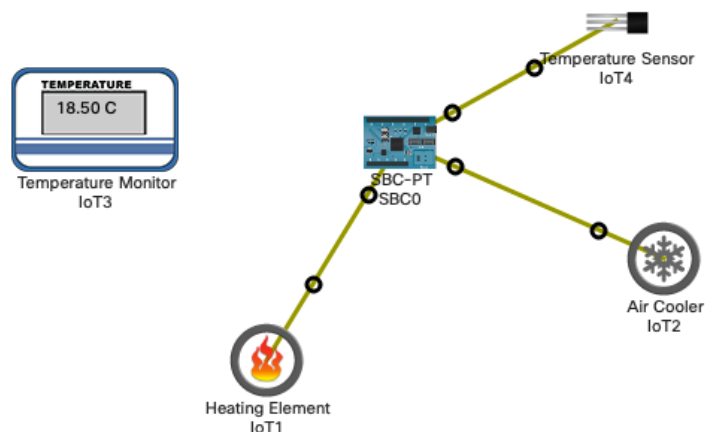


Задание 2

Пункт 1 – 3



Пункт 4-7



Микроконтроллерная схема

После сборки проекта надо написать программу, которая поддерживает температуру в заданном интервале (20 – 25) градусов.

ПРИМЕЧАНИЕ 1:

Чтобы не вкл/выкл постоянно все приборы и поддерживать температуру от 20 до 25 градусов, точкой выхода я поставил 23 градуса, тк это средняя температура на заданном интервале.

Те, если температура была меньше 20 градусов, то система поднимет ее до 23 градусов, а если больше 25 то опустит до 23.

Код программы:

```
from gpio import *
from time import *

# Функция получения информации температуры
def infoTemp():
    # Записываем в переменную значение датчика
    a = analogRead(3)
    # С помощью формулы преобразуем это значение
    b = (float(a)/1023)*200-100
    # Возвращаем значение для дальнейшей работы
    return b

# Функция повышения температуры (получает значения с датчика(преобразованное))
def plusTemp(temp):
    # Записываем сигнал (вкл) в пин 1
    digitalWrite(1, HIGH)
    # Бесконечный цикл
    while True:
        # Перезапись переменной для получения обновленных значений
        temp = infoTemp()
        # Ставим точку выхода 22 (СМ.ПРИМЕЧАНИЕ 1)
        if temp > 22:
            # Записываем сигнал (выкл) в пин 1
            digitalWrite(1, LOW)
            # Возвращаем значение температуры
            return temp
            break
        # Задержка перед считыванием данных
        delay(6000)

# Функция понижения температуры (получает значения с датчика(преобразованное))
def minusTemp(temp):
    # Записываем сигнал (вкл) в пин 2
    digitalWrite(2, HIGH)
    while True:
        # Перезапись переменной для получения обновленных значений
        temp = infoTemp()
        # Ставим точку выхода 22 (СМ.ПРИМЕЧАНИЕ 1)
        if temp < 22:
            # Записываем сигнал (выкл) в пин 2
            digitalWrite(2, LOW)
            # Возвращаем значение температуры
            return temp
            break
        # Задержка перед считыванием данных
        delay(6000)

def main():
    # Изначально все выключено, кроме сенсора
    digitalWrite(1, LOW)
    digitalWrite(2, LOW)

    while True:
        # Получение значения температуры
        t = infoTemp()
        # Если температура ниже 20, то поднимаем ее
        if t < 20:
            t = plusTemp(t)
        # Если температура выше 25, то понижаем ее
        if t > 25:
            t = minusTemp(t)

if __name__ == "__main__":
    main()
```
