

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«БРЕСТСКИЙ ГОСУДАСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
КАФЕДРА ИИТ

**Отчет**  
по лабораторной работе №10  
«Проектирование инфраструктуры «умного дома»»

Выполнил:  
Студент 3 курса  
Факультета ЭИС  
группы АС-50  
**Левкович Р.А.**  
Проверил:  
**Давидюк Ю.И.**

Брест 2020

## Цель работы:

*Приобрести практические навыки проектирования инфраструктуры «умного дома», научиться основам программирования микроконтроллерных устройств.*

## Задание 1

**Последовательно выполните задания, используя режим реального времени СРТ. В качестве результата должна быть получена работающая сетевая инфраструктура, изображенная на рис. 1.**

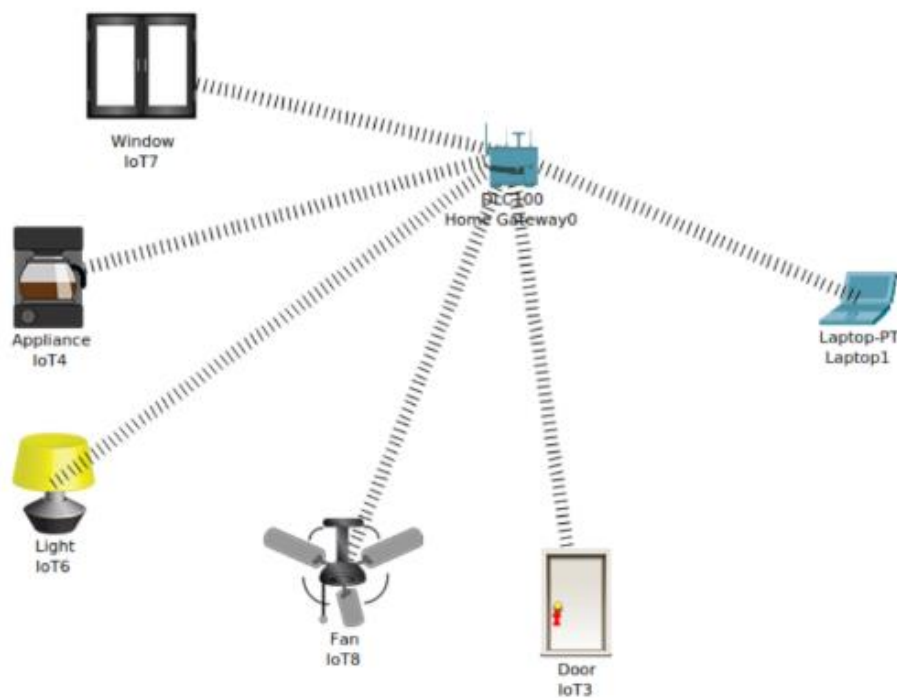
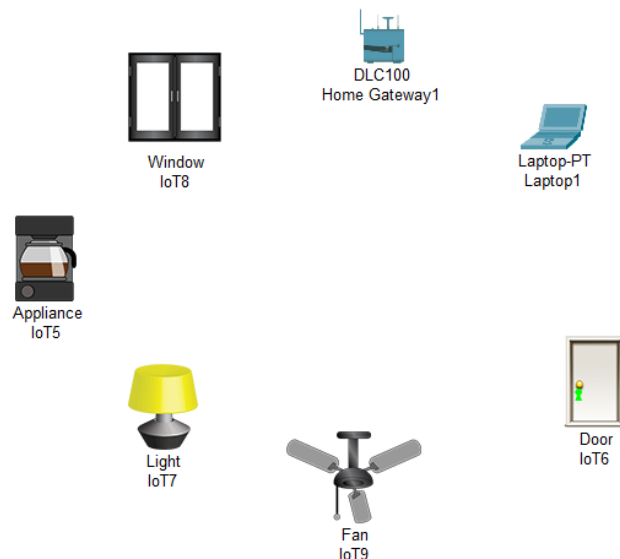


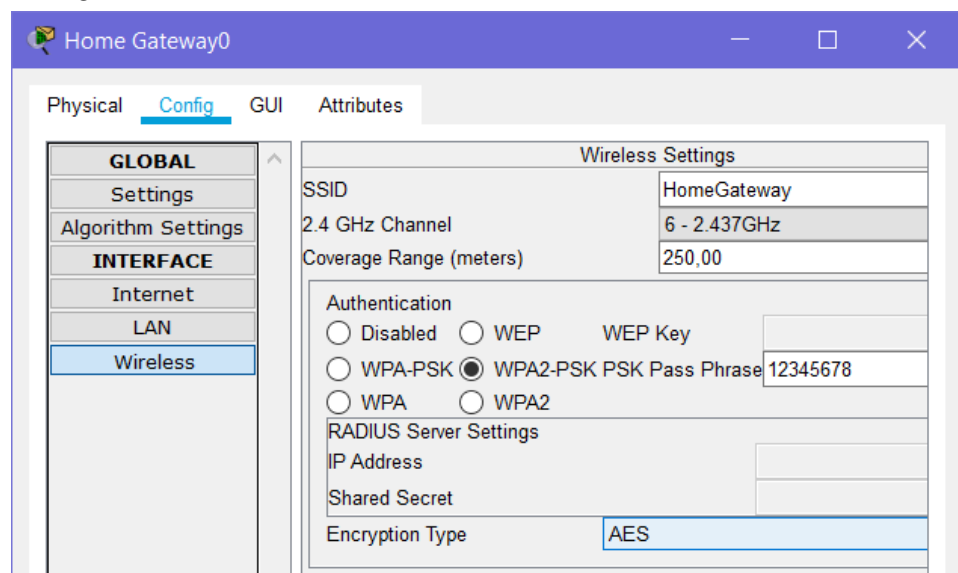
Рис. 1. Схема подключения устройств «умного дома»

- 1) Все необходимые устройства могут быть найдены во вкладках End Devices → End Devices, End Devices → Home и Network Devices → Wireless Devices. Ключевое устройство Home Gateway. Именно оно объединяет все устройства умного дома и клиентские терминалы (такие, как лэптоп) в общую беспроводную сеть. Это сервер IoT.

Нашел все необходимые устройства.

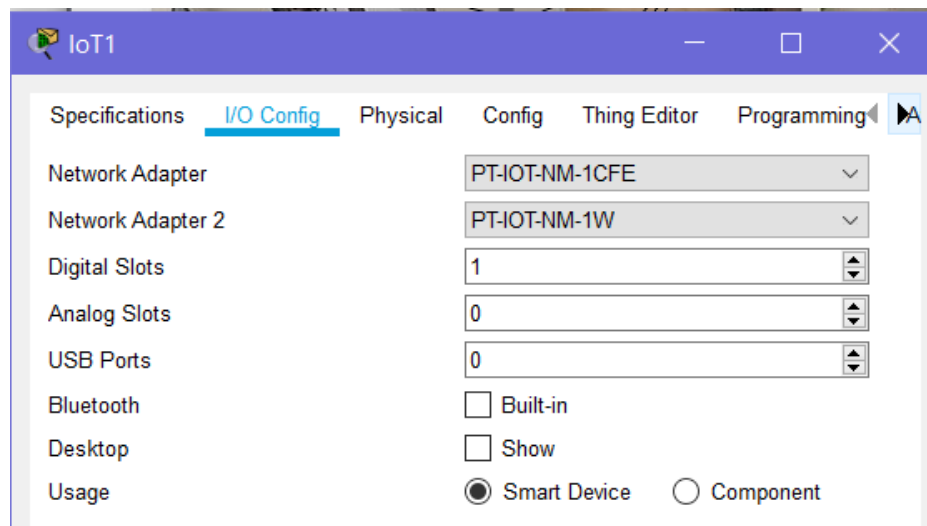


- 2) После размещения всех необходимых устройств в рабочей области откройте Home Gateway и во вкладке Config → Interface → Wireless определите тип аутентификации как WPA2-PSK и задайте любой пароль из 8 символом.

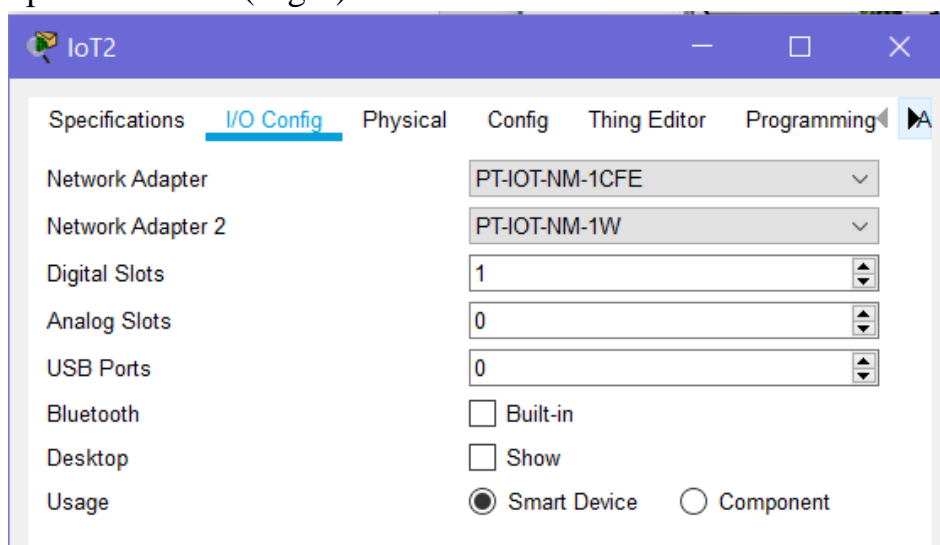


- 3) После настройки сервера, переходим на любое устройство IoT и открываем расширенные настройки (Advanced). Дело в том, что эти устройства по умолчанию не поддерживают беспроводную передачу данных. Откройте вкладку I/O Config. Далее в списке Network Adapter2 выберите беспроводной адаптер PT- IOT-NM-1W.

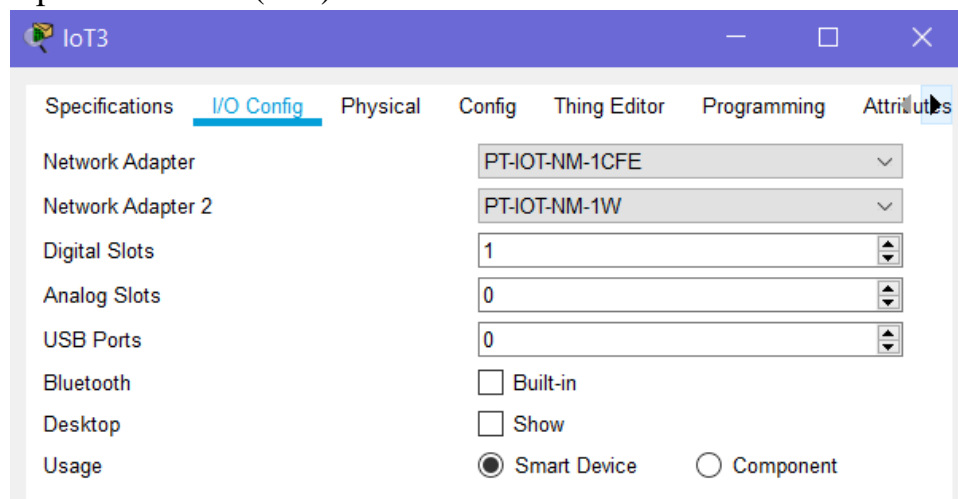
Возьмем устройство IoT1 (Windows)



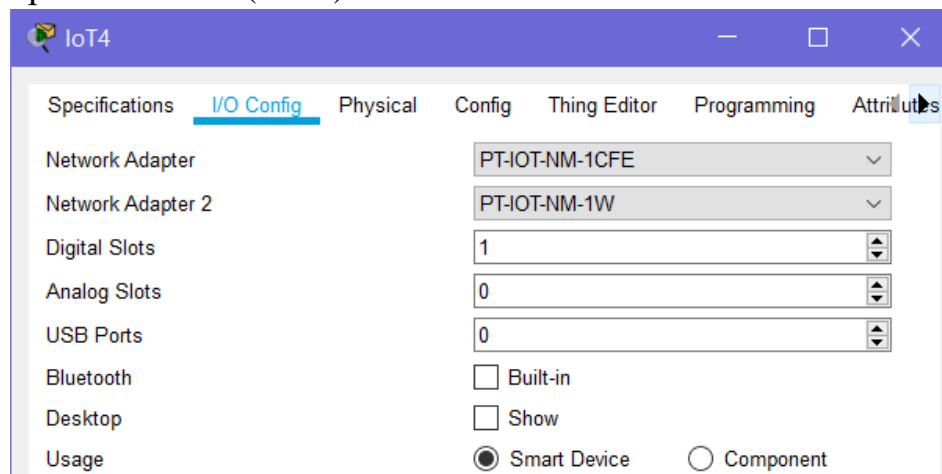
Для устройства IoT2 (Light)



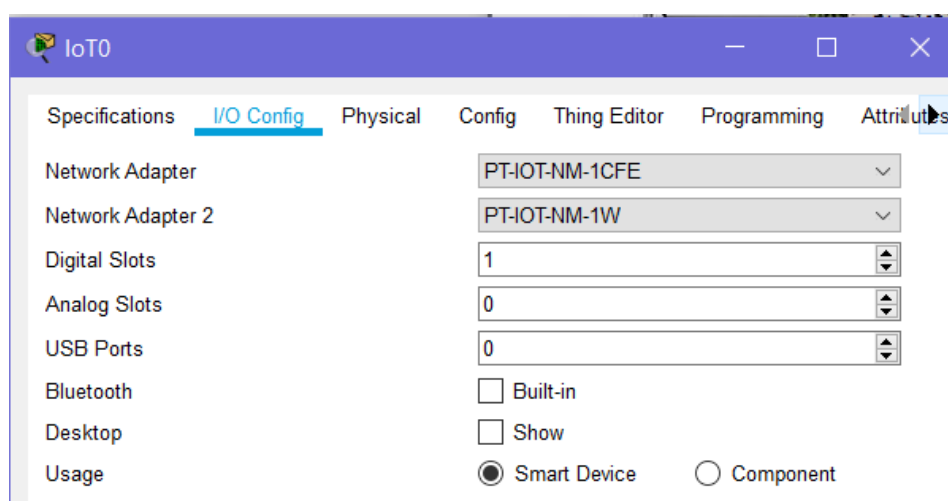
Для устройства IoT3 (Fan)



Для устройства IoT4 (Door)



Для устройства IoT0 (Appliance)



- 4) После выполнения предыдущего действия во вкладке Config появится беспроводной интерфейс Wireless3. Откройте его и настройте подключение к серверу, задав правильный тип аутентификации, пароль и выбрав вариант DHCP в IP Configuration. В данном случае сервер IoT Home Gateway является DHCP- сервером для подключаемых устройств (автоматически раздает IP-адреса).

## Возьмем устройство IoT1 (Windows)

The screenshot shows the configuration window for IoT1. The 'Config' tab is selected, and the 'Wireless3' interface is chosen from the left sidebar. The configuration details are as follows:

Wireless3	
Port Status	<input checked="" type="checkbox"/> On
Bandwidth	300 Mbps
MAC Address	00E0.8F03.9647
SSID	HomeGateway
<b>Authentication</b>	
<input type="radio"/> Disabled	<input type="radio"/> WEP
<input type="radio"/> WPA-PSK	<input checked="" type="radio"/> WPA2-PSK
<input type="radio"/> WPA	<input type="radio"/> WPA2
<input type="radio"/> 802.1X	Method: MD5
WEP Key	
PSK Pass Phrase	12345678
User ID	
Password	
User Name	
Password	
Encryption Type	AES
<b>IP Configuration</b>	
<input checked="" type="radio"/> DHCP	
<input type="radio"/> Static	
IP Address	192.168.25.101
Subnet Mask	255.255.255.0

## Для устройства IoT2 (Light)

The screenshot shows the configuration window for IoT2. The 'Config' tab is selected, and the 'Wireless3' interface is chosen from the left sidebar. The configuration details are as follows:

Wireless3	
Port Status	<input checked="" type="checkbox"/> On
Bandwidth	300 Mbps
MAC Address	0040.0B14.1991
SSID	HomeGateway
<b>Authentication</b>	
<input type="radio"/> Disabled	<input type="radio"/> WEP
<input type="radio"/> WPA-PSK	<input checked="" type="radio"/> WPA2-PSK
<input type="radio"/> WPA	<input type="radio"/> WPA2
<input type="radio"/> 802.1X	Method: MD5
WEP Key	
PSK Pass Phrase	12345678
User ID	
Password	
User Name	
Password	
Encryption Type	AES
<b>IP Configuration</b>	
<input checked="" type="radio"/> DHCP	
<input type="radio"/> Static	
IP Address	192.168.25.103
Subnet Mask	255.255.255.0

## Для устройства IoT3 (Fan)

The screenshot shows the configuration window for IoT3 (Fan). The 'Config' tab is selected, and the 'Wireless3' interface is chosen from the left sidebar. The configuration details are as follows:

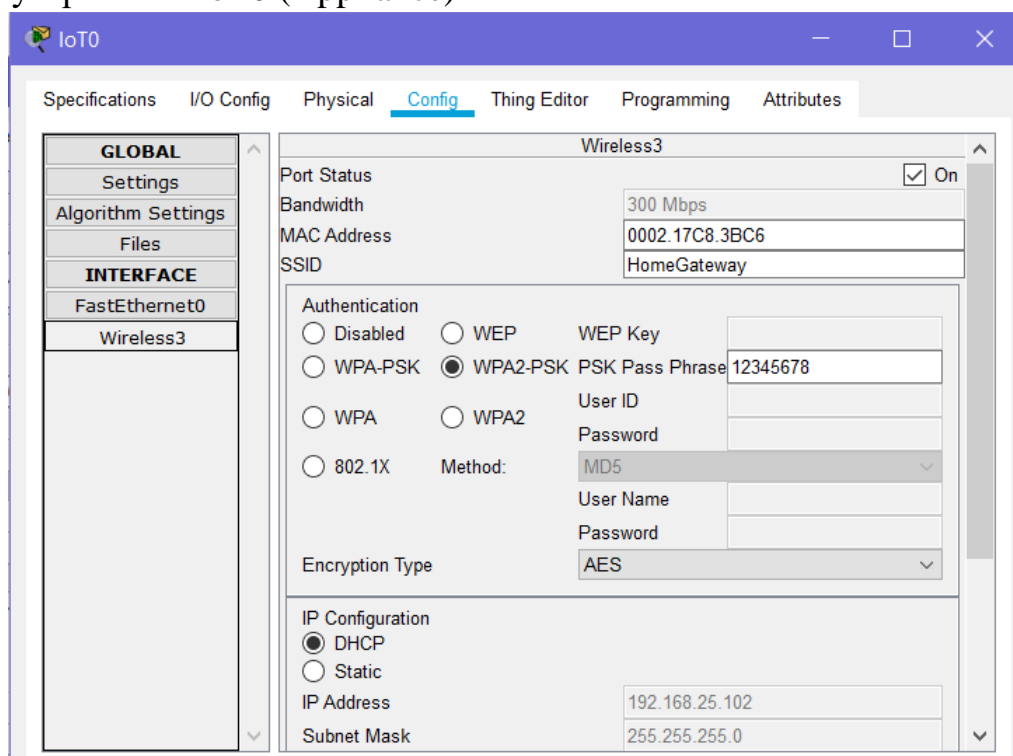
Wireless3	
Port Status	<input checked="" type="checkbox"/> On
Bandwidth	300 Mbps
MAC Address	00D0.BA82.43BB
SSID	HomeGateway
<b>Authentication</b>	
<input type="radio"/> Disabled	<input type="radio"/> WEP
<input type="radio"/> WPA-PSK	<input checked="" type="radio"/> WPA2-PSK
<input type="radio"/> WPA	<input type="radio"/> WPA2
<input type="radio"/> 802.1X	Method: MD5
WEP Key	
PSK Pass Phrase	12345678
User ID	
Password	
User Name	
Password	
Encryption Type	AES
<b>IP Configuration</b>	
<input checked="" type="radio"/> DHCP	
<input type="radio"/> Static	
IP Address	192.168.25.104
Subnet Mask	255.255.255.0

## Для устройства IoT4 (Door)

The screenshot shows the configuration window for IoT4 (Door). The 'Config' tab is selected, and the 'Wireless3' interface is chosen from the left sidebar. The configuration details are as follows:

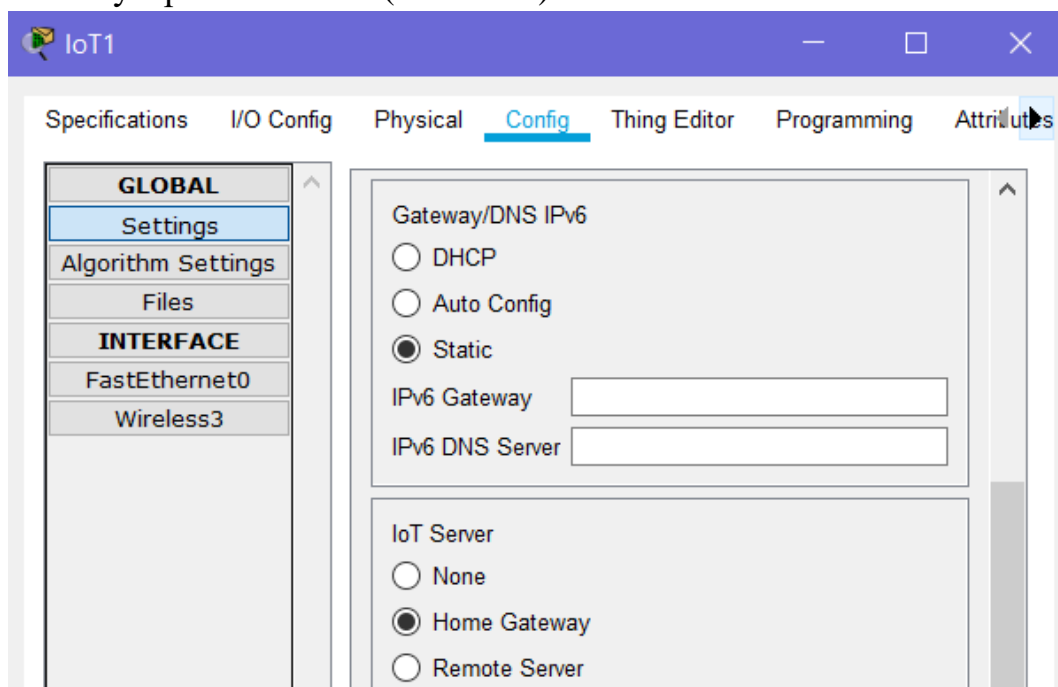
Wireless3	
Port Status	<input checked="" type="checkbox"/> On
Bandwidth	300 Mbps
MAC Address	00D0.BA55.579C
SSID	HomeGateway
<b>Authentication</b>	
<input type="radio"/> Disabled	<input type="radio"/> WEP
<input type="radio"/> WPA-PSK	<input checked="" type="radio"/> WPA2-PSK
<input type="radio"/> WPA	<input type="radio"/> WPA2
<input type="radio"/> 802.1X	Method: MD5
WEP Key	
PSK Pass Phrase	12345678
User ID	
Password	
User Name	
Password	
Encryption Type	AES
<b>IP Configuration</b>	
<input checked="" type="radio"/> DHCP	
<input type="radio"/> Static	
IP Address	192.168.25.105
Subnet Mask	255.255.255.0

Для устройства IoT0 (Appliance)



- 5) Далее откройте Settings (там же, во вкладке Config) и поставьте в группе IoT Server переключатель в положении Home Gateway.

Возьмем устройство IoT1 (Windows)





Для устройства IoT2 (Light)

The screenshot shows the IoT2 configuration window with the 'Config' tab selected. The left sidebar contains a tree view with 'GLOBAL' (containing 'Settings', 'Algorithm Settings', and 'Files') and 'INTERFACE' (containing 'FastEthernet0' and 'Wireless3'). The main area is divided into two sections. The top section, 'Gateway/DNS IPv6', has three radio buttons: 'DHCP', 'Auto Config', and 'Static' (which is selected). Below these are two text input fields for 'IPv6 Gateway' and 'IPv6 DNS Server'. The bottom section, 'IoT Server', has three radio buttons: 'None', 'Home Gateway' (which is selected), and 'Remote Server'.

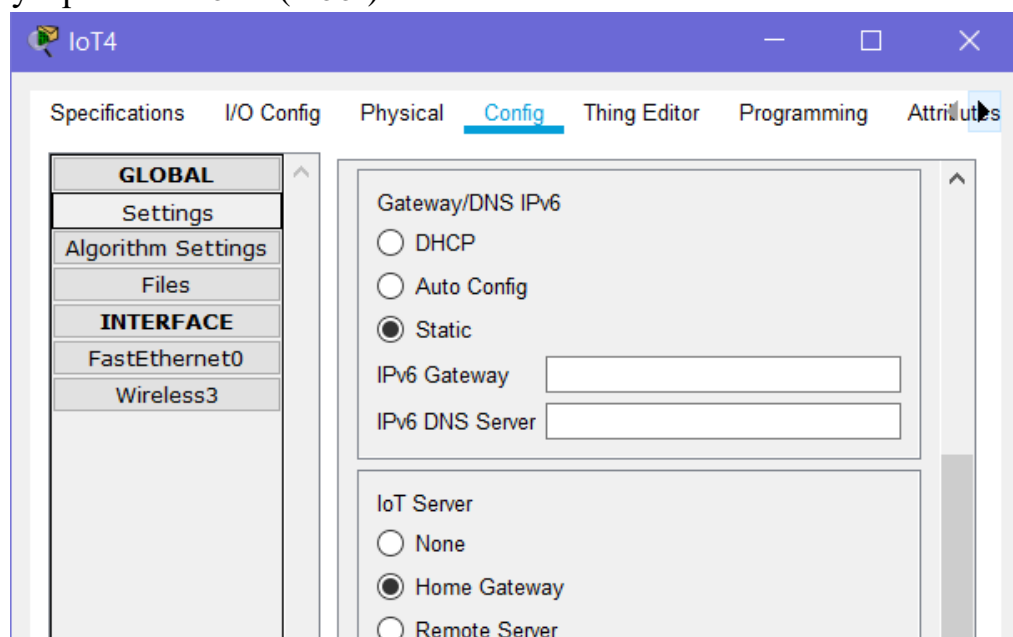
Tab	Section	Option	Value
Config	Gateway/DNS IPv6	DHCP	<input type="radio"/>
		Auto Config	<input type="radio"/>
		Static	<input checked="" type="radio"/>
	Gateway/DNS IPv6	IPv6 Gateway	<input type="text"/>
		IPv6 DNS Server	<input type="text"/>
	IoT Server	None	<input type="radio"/>
Home Gateway		<input checked="" type="radio"/>	
Remote Server		<input type="radio"/>	

Для устройства IoT3 (Fan)

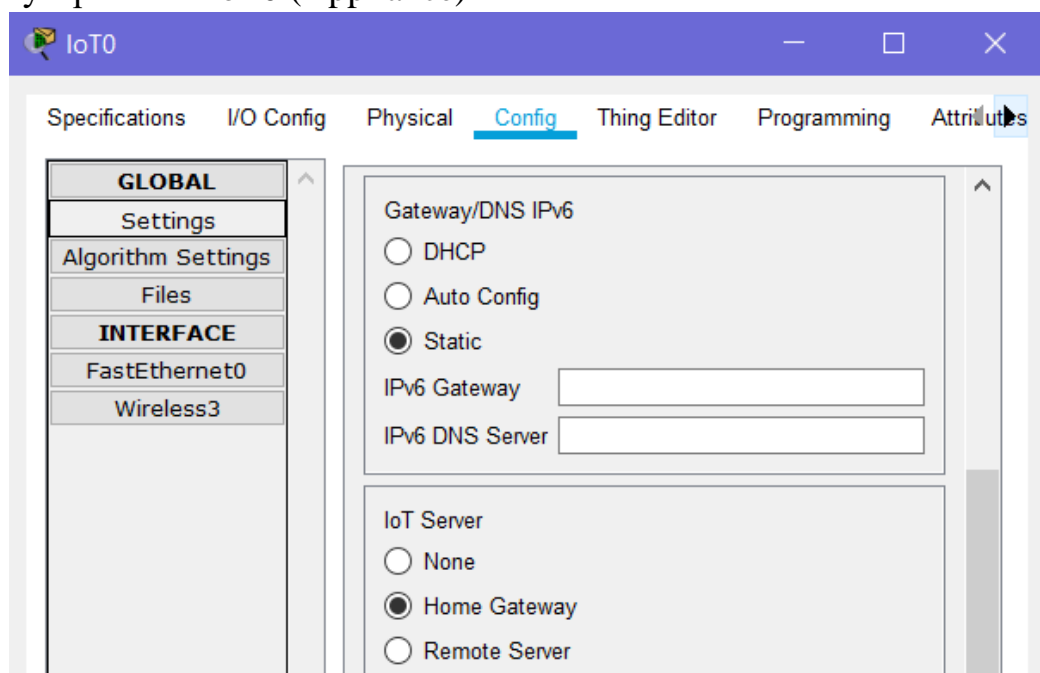
The screenshot shows the IoT3 configuration window with the 'Config' tab selected. The left sidebar contains a tree view with 'GLOBAL' (containing 'Settings', 'Algorithm Settings', and 'Files') and 'INTERFACE' (containing 'FastEthernet0' and 'Wireless3'). The main area is divided into two sections. The top section, 'Gateway/DNS IPv6', has three radio buttons: 'DHCP', 'Auto Config', and 'Static' (which is selected). Below these are two text input fields for 'IPv6 Gateway' and 'IPv6 DNS Server'. The bottom section, 'IoT Server', has three radio buttons: 'None', 'Home Gateway' (which is selected), and 'Remote Server'.

Tab	Section	Option	Value
Config	Gateway/DNS IPv6	DHCP	<input type="radio"/>
		Auto Config	<input type="radio"/>
		Static	<input checked="" type="radio"/>
	Gateway/DNS IPv6	IPv6 Gateway	<input type="text"/>
		IPv6 DNS Server	<input type="text"/>
	IoT Server	None	<input type="radio"/>
Home Gateway		<input checked="" type="radio"/>	
Remote Server		<input type="radio"/>	

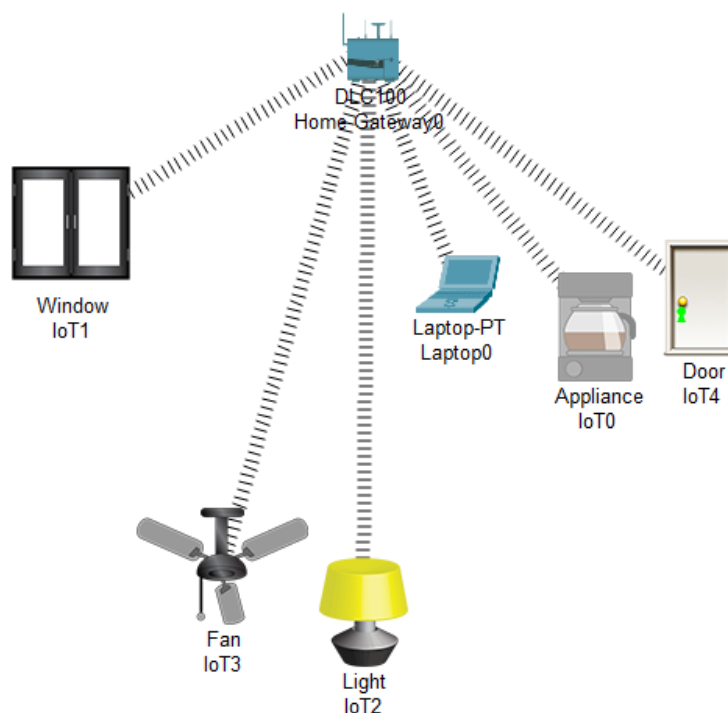
Для устройства IoT4 (Door)



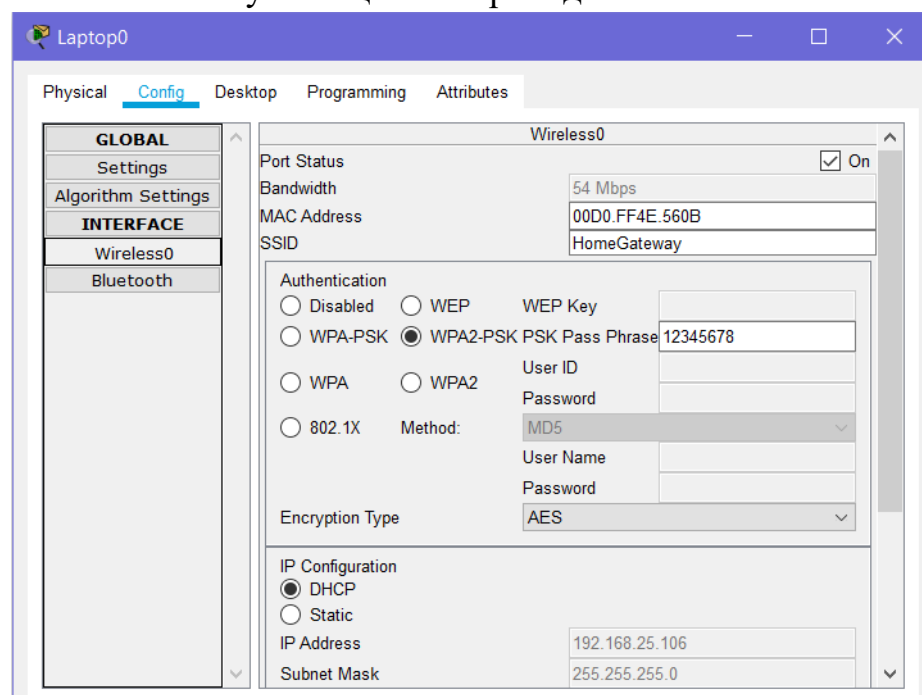
Для устройства IoT0 (Appliance)



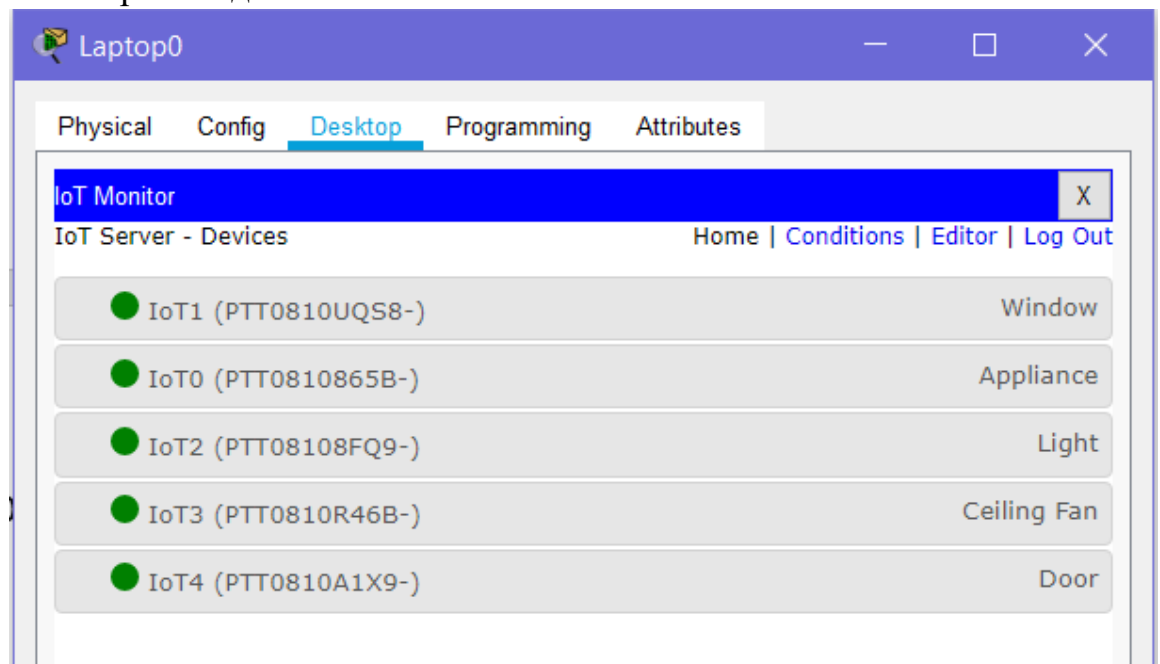
6) После выполнения всех этих действий, убедитесь, что между сервером и настраиваемым узлом появилось отображение беспроводной связи.



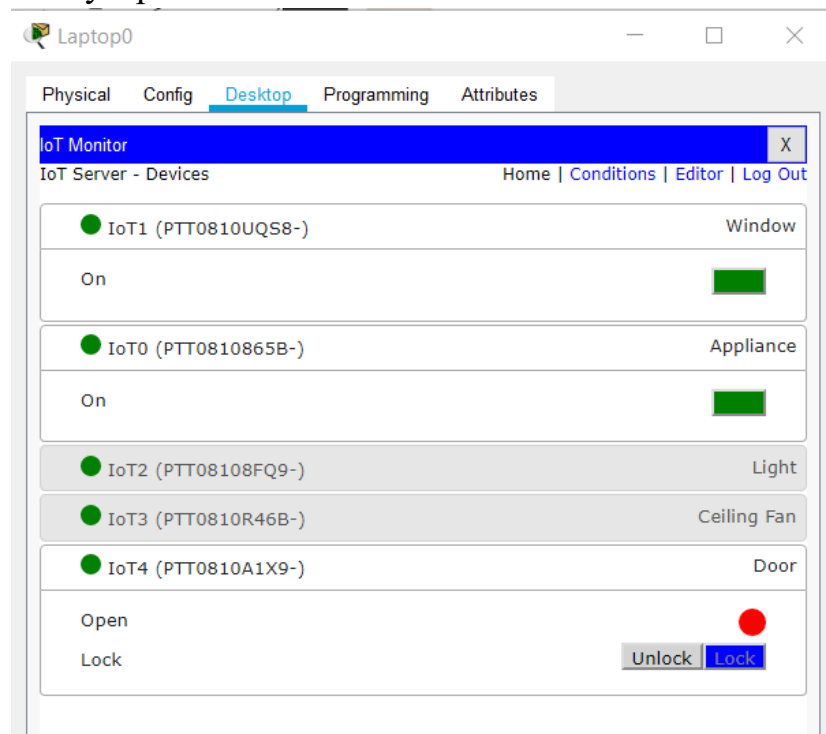
7) Откройте лэптоп и изучите его физическую конфигурацию. Вы можете заметить, что на нем также, как и на IoT-устройствах не установлен модуль беспроводной связи. Это можно исправить следующим образом: извлеките установленный Fast Ethernet-модуль (предварительно выключив лэптоп) и поместите в свободный слот модуль PT-LAPTOP-NM-1W. После этого включите устройство и произведите похожие настройки беспроводного интерфейса (укажите SSID, тип аутентификации и пароль). Между сервером и лэптопом должна появиться визуализация беспроводной связи.

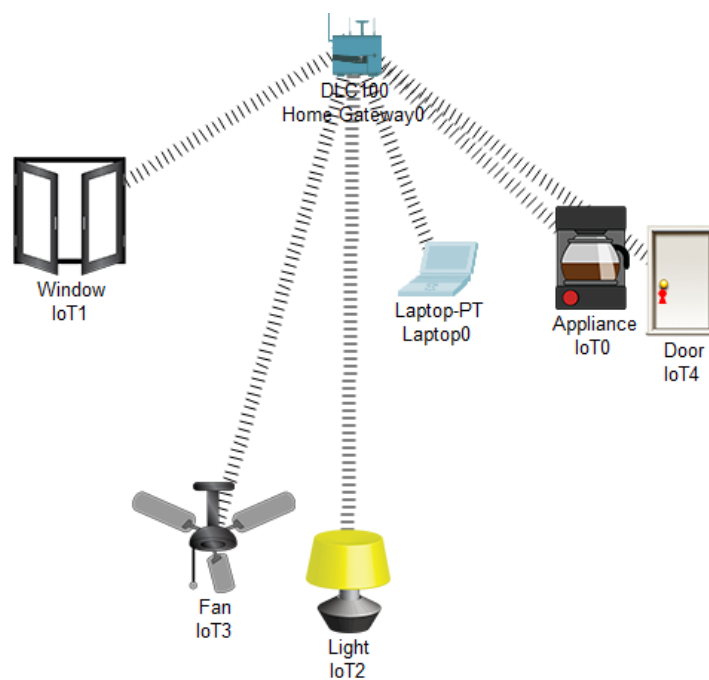


- 8) Откройте вкладку Desktop ноутбука и далее в IoT Monitor. Нажмите ОК в окне авторизации на сервере, убедившись в правильности написанного IP-адреса сервера. После этого перед вами должен появиться список всех беспроводных устройств, подключенных к нашему серверу. Поэкспериментируйте с кнопками включения/выключения устройств и изучите изменения, которые с ними происходят.



Изменения, которые произошли при включение/выключение кнопок определенных устройств.





Мы открыли окно, включили кофеварку и закрыли дверь.

- 9) Добавьте фон для построенной инфраструктуры, воспользовавшись предложенными background или своими.

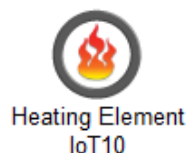
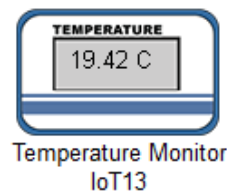


## Задание 2

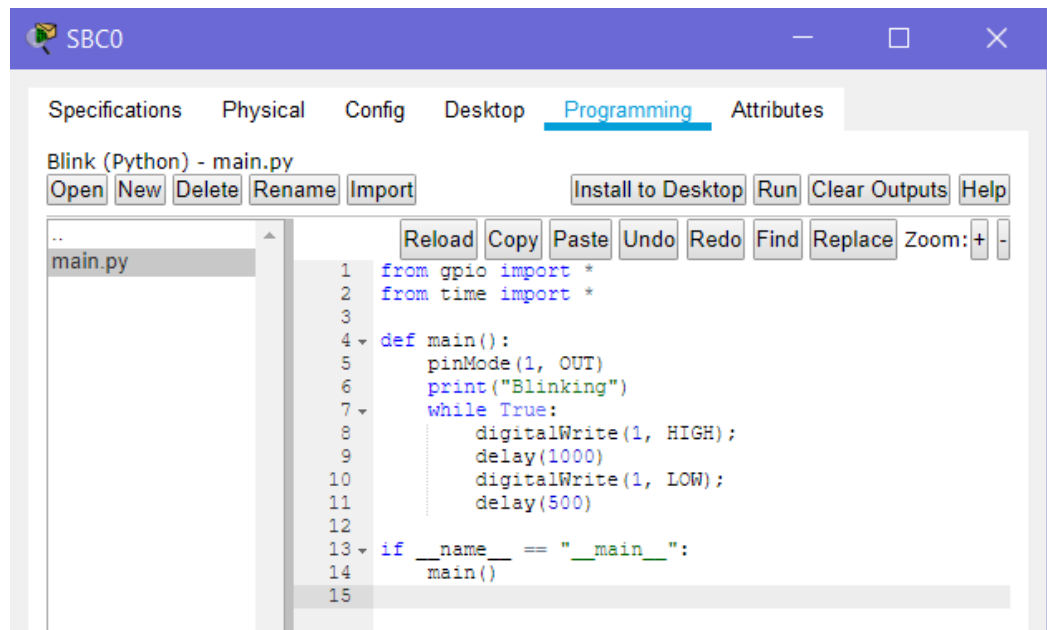
В первом задании, несмотря на наличие IoT-устройств, сформирована лишь сетевая инфраструктура, но не полноценное IoT-решение. Это так, поскольку все устройства контролируются (пусть и удаленно), но человеком. Т.е. человек принимает решение о включении/выключении устройств, а не сама система.

Попробуем создать решение, которое будет обладать определенной автономностью. Для этого воспользуемся микроконтроллерными устройствами, которые будут принимать решение о активации тех или иных узлов системы. Спроектируем систему для поддержания комфортной температуры внутри помещения, изображенную на рис.3

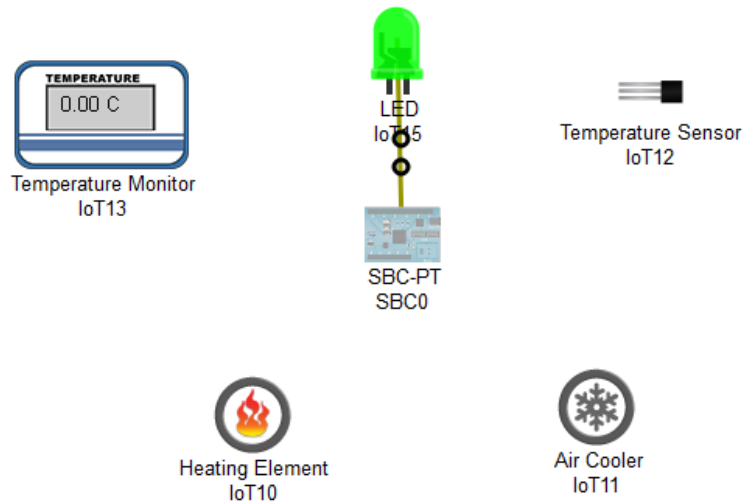
- 1) Для начала добавьте микроконтроллерную плату в рабочую область (вкладка Components → Boards). Выберите из предложенных плату SBC Board.



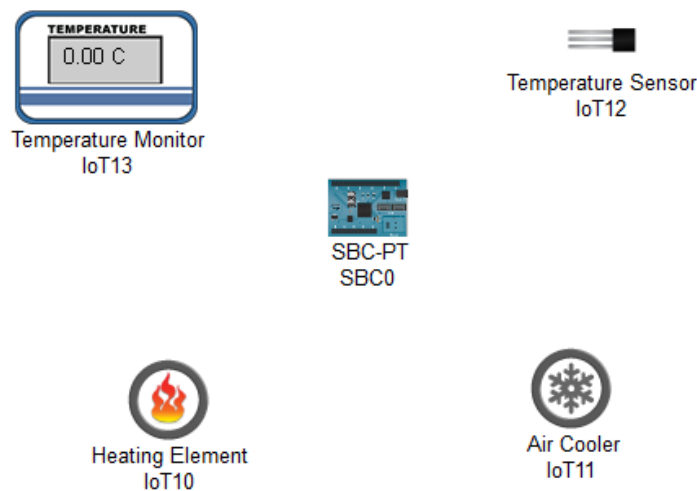
- 2) Откройте добавленную плату на вкладке Programming. Далее в списке слева выберите пункт Blink (Python) и далее скрипт main.py. Программирование для такой платы производится на языке Python. Он является достаточно простым скриптовым языком с большим количеством разработанных библиотек. Скрипт, который откроется, нужен для решения простой задачи – он включает и выключает пин (разъем) на нашей плате, активируя подключенную к нему нагрузку. В качестве такой нагрузки может выступать светодиоды, разные датчики, LCD-экраны и т.д.



- 3) Попробуйте добавить светодиод (LED) с вкладки Components → Actuators к рабочей области. Затем во вкладке Connections выберите тип соединения IoT Custom Cable и соедините пин D1 вашей платы с пином D0 светодиода. Запустите программу, нажав на кнопку Run. Вы должны увидеть мигающий светодиод. Откройте программу, попытайтесь изучить и понять ее содержимое. Команда `pinMode` нужна для определения режима, в котором будет работать наш пин платы (это может быть IN или OUT – для выходных и входных сигналов соответственно). Как следует из программы, мы делаем пин D1 (или просто пин с номером 1) выходным, для того, чтобы регулировать уровень напряжения и включать и выключать его. Пины бывают цифровыми (D) и аналоговыми (A). Цифровые пины оперируют 0 и 1 (или LOW и HIGH) и лучше всего описывают взаимодействие с устройствами, которые нужно включать и выключать. Аналоговые пины нужны для передачи какой-то многоуровневой информации (например, уровня температуры и влажности). Как вы видите, в программе мы записываем попеременно высокий и низкий сигнал в пин номер 1, что приводит к миганию светодиода (это делается с помощью функции `digitalWrite` с указанием номера пина и уровня сигнала). Функция `delay` вызывает задержку перед выполнением следующей команды на указанное количество миллисекунд.

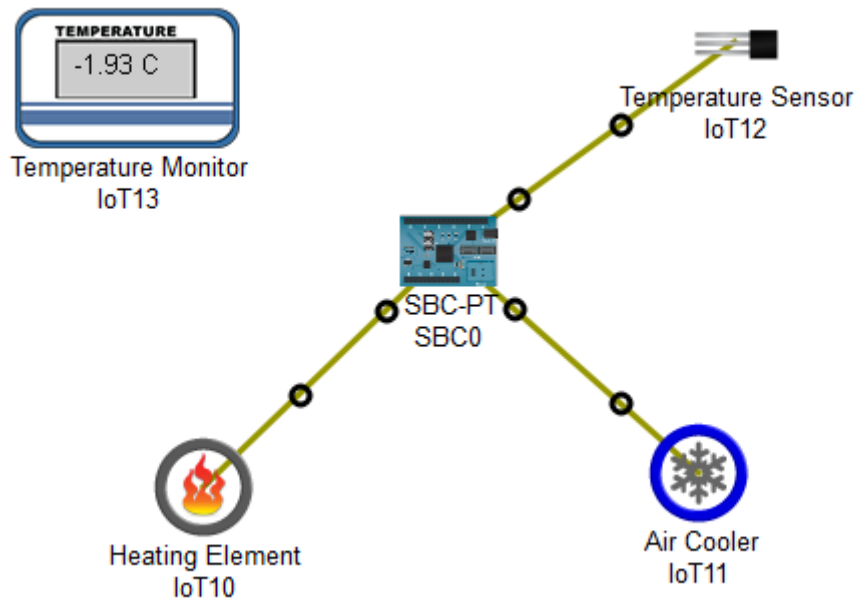


- 4) Удалите LED из рабочей области. Добавьте другие компоненты, необходимые для реализации проекта (вкладка Actuators), а также цифровой термометр для отслеживания температуры (End Devices → Home→ Temperature Monitor)). Температурный сенсор находится на вкладке (Components → Sensors→ Temperature Sensor).

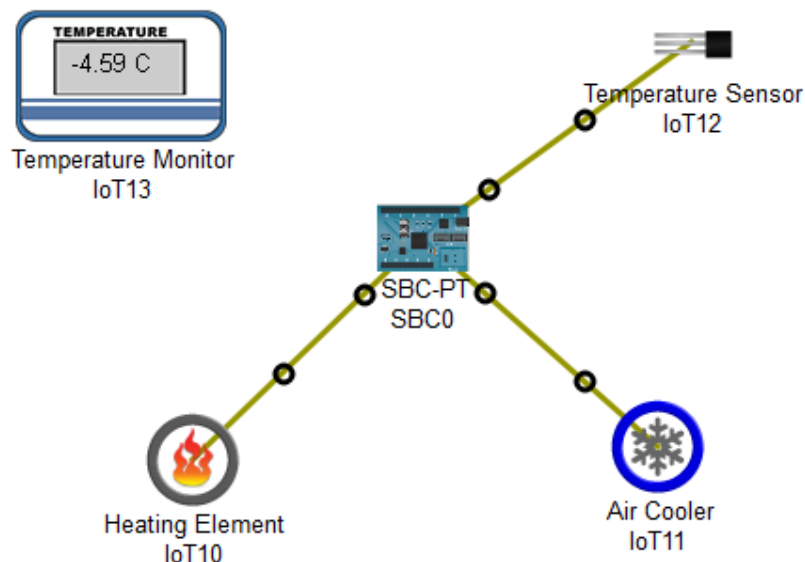


- 5) Heating Element нужен для повышения температуры, Air Cooler для понижения. О характеристиках этих устройств можно почитать, кликнув по ним. Для нас важно то, что они включаются и выключаются как цифровые устройства (т.е. вызовом команды digitalWrite). Temperature Monitor нужен для считывания данных о температуре. Это аналоговый датчик, поэтому для считывания данных применяется функция analogRead с указанием единственного параметра – номера пина. Подсоедините все указанные датчики к плате, выбрав произвольные пины (запомните свой выбор). Для Temperature Monitor выберите пин A0 на нем.





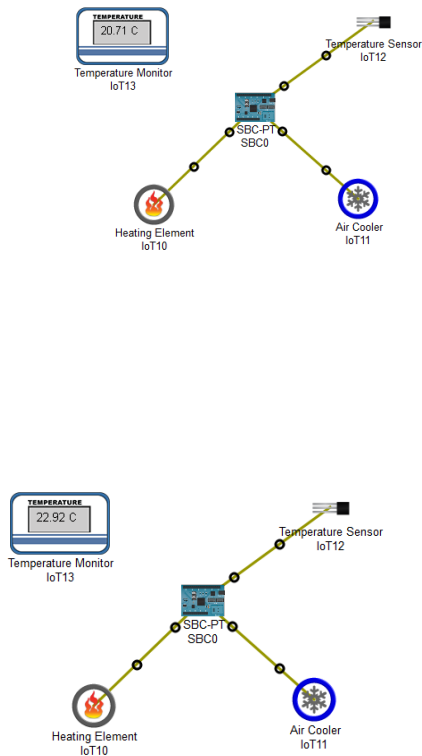
6) Далее изучите изменение температуры в течение суток с помощью показателей температурного монитора. В СРТ можно изменять текущее время суток (это делается нажатием на кнопку с «текущим» временем или Shift + E. Как вы заметите, температура изменяется. Хотелось бы, чтобы она оставалась в определенном заданном интервале (например, от 20 до 25 градусов).



7) Итак, мы подошли к самому главному. Теперь вам нужно написать программу, которая будет поддерживать текущую температуру в заданном интервале. Используйте пины, активируя устройства для обогрева и охлаждения на основании данных, считанных с температурного датчика. Имейте в виду, что датчик возвращает данные в интервале от 0 до 1023, соответствующие температуре -100 до 100 градусов. Используйте следующую формулу для получения значения температуры:

$$t_{celsius} = \frac{t_{sensor}}{1023} * 200 - 100$$

Функция float нужна для конвертации в вещественный тип.



```

1 from gpio import *
2 from time import *
3
4 def main():
5     pinMode(0, IN)
6     pinMode(1, OUT)
7     pinMode(2, OUT)
8     print("Temperature")
9     while True:
10        a = ((analogRead(0)*200/1023)-100)
11        if a<20:
12            digitalWrite(2, HIGH);
13            digitalWrite(1, LOW);
14        if a>25:
15            digitalWrite(1, HIGH);
16            digitalWrite(2, LOW);
17            print(a);
18            delay(1000)
19
20 if __name__ == "__main__":
21     main()

```

```

22
22
22
22

```

Код программы

```

from gpio import *
from time import *

```

```

def main():
    pinMode(0, IN)
    pinMode(1, OUT)
    pinMode(2, OUT)
    print("Temperature")
    while True:
        a = ((analogRead(0)*200/1023)-100)
        if a<20:
            digitalWrite(2, HIGH);
            digitalWrite(1, LOW);
        if a>25:

```

```
        digitalWrite(1, HIGH);  
        digitalWrite(2, LOW);  
    print(a);  
    delay(1000)  
  
if __name__ == "__main__":  
    main()
```

**Вывод: Приобрели необходимые навыки проектирования инфраструктуры «умного дома».**