«Умный дом 2» отличался от предыдущего примера «умного дома», основное отличие заключалось в интеллектуальном IoT, который теперь предоставляется удаленным поставщиком как услуга.

В этом случае владельцу дома нужно было только настроить домашнюю WLAN, а затем была предоставлена логика бэкенда путем удаленного подключения к облачному серверу, размещенному сторонним провайдером. В этом сценарии владелец дома также имел возможность удаленного подключения к главной странице управления IoT с помощью собственного мобильного телефона через сеть 3G.

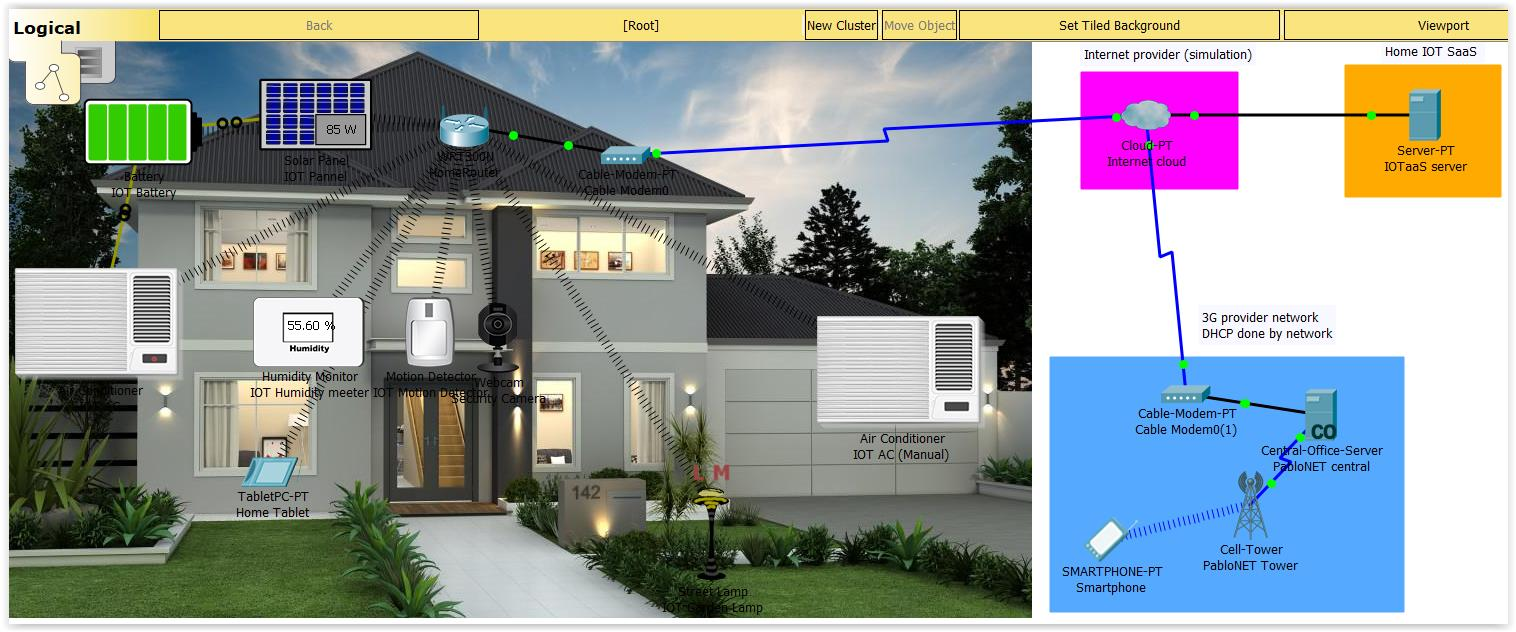


Рисунок 25. Схема Cisco Packet Tracer для модели Smart-Home 2 (SaaS)

Расположение сети

Smart-Home 2 имел более сложную сетевую настройку по сравнению с другим домашним примером. В этом случае схема сети была разделена на четыре области: домашняя сеть, поставщик Интернет-услуг, мобильная сеть 3G и внутренний IoT.

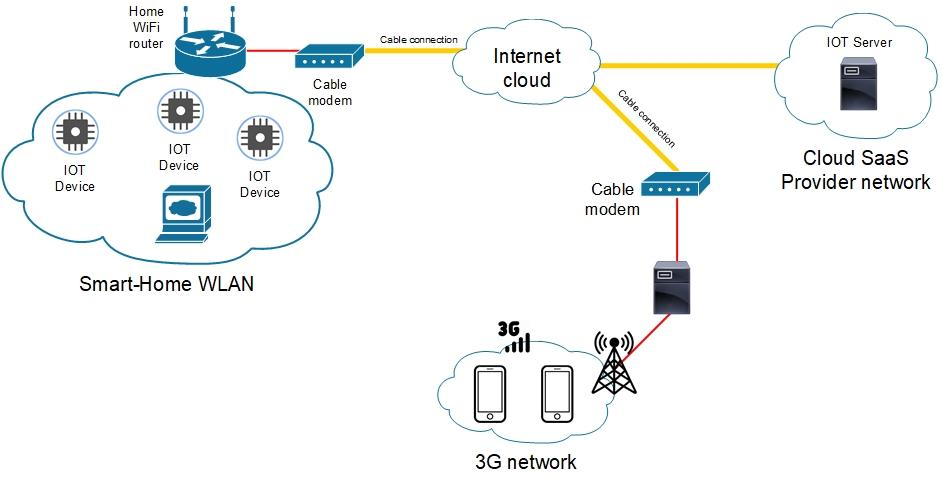


Рисунок 26 - Топология сети Smart-Home 2 (SaaS)

Несмотря на необходимость четырех сетей, настройка была проще по сравнению с предыдущим сценарием поскольку IoT, DNS и DHCP-сервер не были подключены к WLAN. Беспроводная сеть была предоставлена ​​беспроводным маршрутизатором, который в данном случае также предоставил функции DHCP.

Будучи простой домашней сетью, настройка устройства осуществлялась только с использованием определенного SSID и пароля. Домашний кабельный модем также использовался только для подключения маршрутизатора к Интернету, настройка этого компонента невозможна.

Сеть ISP, как и в другом примере с «умным домом», искусственно соединяла различные интерфейсы друг с другом для имитации интернет-соединения. Коаксиальные кабели, поступающие от домашнего модема и провайдера сети 3G, были подключены к кабелю Ethernet, к которому был подключен облачный сервер IoT.

Чтобы моделировать удаленную сеть, в этом примере использовалась инфраструктура мобильного провайдера 3G. Это позволило владельцу дома подключиться к странице мониторинга IoT, используя свой мобильный телефон вне домашней беспроводной локальной сети. Настройка сотовой сети была очень простой и требуется только вышка сотовой связи и сервер центрального офисного центра. Сервер консолидировал все

Сигналы, поступающие через коаксиальные кабели в магистральное соединение Ethernet, которое через кабельный модем затем подключалось к сети ISP.

И у вышки сотовой связи, и у сервера центрального офиса было ограниченное количество параметров, которые можно было настроить. К серверу центрального офиса можно подключить максимум шесть вышек сотовой связи.

Затем портативные устройства были подключены к сети сотовой вышки 3G, настроив правильное имя APN.

Чтобы упростить сеть облачного провайдера, удаленный сервер IoT был напрямую подключен к провайдеру без использования какого-либо маршрутизатора. Сервер использовал статический IP-адрес, чтобы убедиться, что устройство IoT всегда может подключиться к нему. Службы DNS также работали на этом сервере, преобразовывая URL-адрес IOTSmarthome.com в собственный IP-адрес.

Настройка облачных сервисов SaaS, использованных в этом упражнении, была искусственной и не была бы реалистичной в реальном SaaS-сценарии.

План IoT

Как и в другом примере с умным домом, все устройства IoT были подключены к одной сети, единственное отличие заключалось в том, что сервер IoT не был подключен к домашней сети. Для этого требовалось, чтобы локальный маршрутизатор WLAN действовал как DHCP-сервер, но для функций IoT устройство было подключено к удаленному IoT-серверу.

Поскольку ранее все устройства также использовали одно и то же имя пользователя и пароль IoT, владелец дома также должен был использовать одинаковые учетные данные для удаленного подключения через домашнюю страницу IoT через браузер.

Как видно из рисунка 27 ниже, в этом моделировании уже было восемь предварительно настроенных IoT Smart-устройств: солнечная панель, батарея, два блока переменного тока, датчик влажности, датчик движения, веб-камера и садовый светильник.

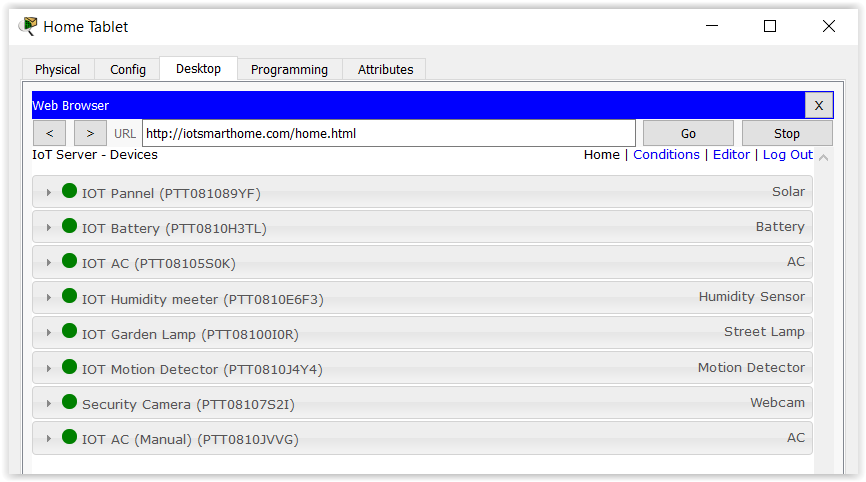


Рисунок 27 - Список подключенных устройств IoT

Также две симуляции IoT были предварительно настроены и видны с главной домашней страницы IoT, как показано на рисунке 28.

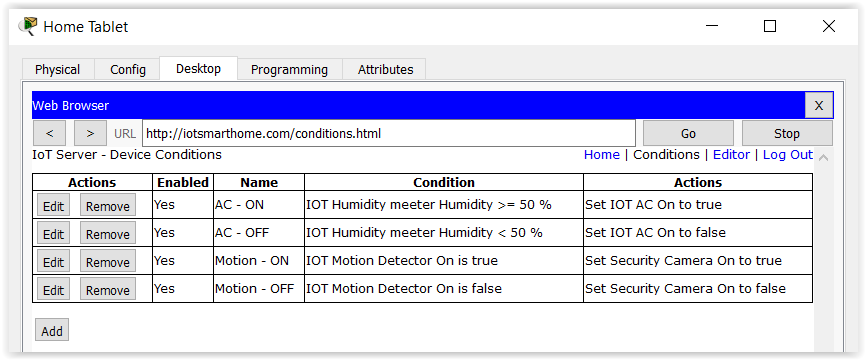


Рисунок 28 - Предварительно установленные симуляции IoT

Первая автоматизация повторила сценарий, в котором датчик влажности IoT был установлен в доме для контроля влажности окружающей среды. В случае, если обнаруженная влажность превышает предварительно установленное пороговое значение, блок переменного тока автоматически включался, понижая уровень влажности. Уровень влажности окружающей среды можно контролировать как с помощью датчика, так и путем выбора параметра на главной странице IOT.

Чтобы увеличить сложность моделирования, солнечная батарея и умная батарея были

также добавлен и подключен к блоку переменного тока.

Идея заключалась в том, что в дневное время, когда светило солнце, солнечная панель вырабатывала электричество, заряжая батарею. В то же время, когда в доме не повышалась влажность, включался блок переменного тока. Будучи блоком переменного тока, подключенным к батарее IoT, можно было наблюдать, что батарея накапливала электричество, когда блок переменного тока был выключен, но быстро разряжалась, когда был включен переменный ток.

Однако стоит упомянуть, что благодаря логике Cisco Packet Tracer переменный ток не остановился, если накопленная мощность батареи достигла нуля, а также что батарея естественным образом разрядила заряд, даже если переменный ток не был включен.

Оба количества электричества, производимого солнечной панелью и оставшейся мощностью в батарее, можно увидеть в режиме реального времени с главной домашней страницы IoT, как показано на рисунке 29.

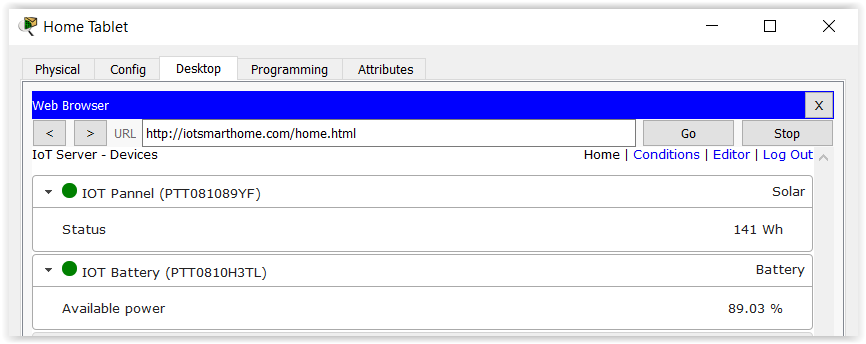


Рисунок 29 - Мониторинг электроэнергии, производимой и хранимой устройствами IoT

Второе моделирование IoT, включенное в упражнение, было похоже на предыдущую систему сигнализации «Умный дом». Основное отличие в этом случае заключалось в том, что детектор движения запускает не сирену, а в течение нескольких секунд веб-камеру, если что-то обнаружено.

Имитация может быть подтверждена нажатием клавиши ALT на клавиатуре и наведением на сенсор датчика движения вместе с ними, с домашней страницы IoT веб-камера затем передает некоторые изображения.

На рисунке 30 показан случай, когда никакие объекты не были обнаружены, а веб-камера была отключена, на рисунке 31 можно увидеть, что датчик сработал, а веб-камера показала изображения.

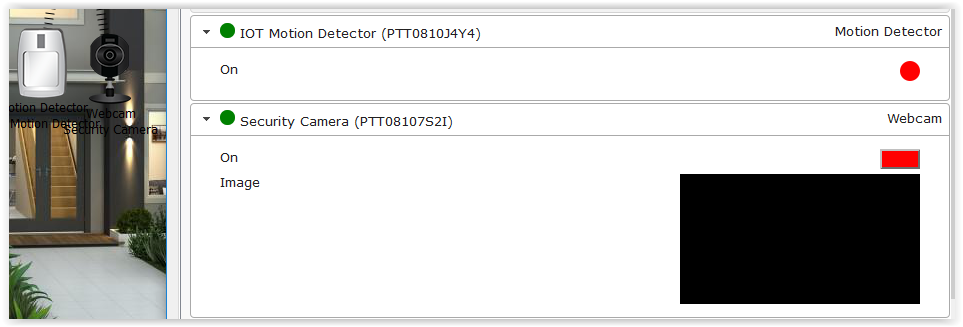


Рисунок 30 - Веб-камера выключена, так как датчик движения не обнаруживает движений

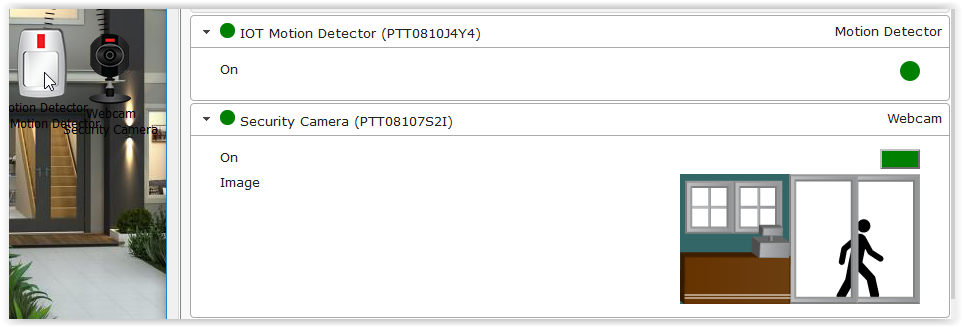


Рисунок 31 - Веб-камера включена, когда движения обнаруживаются датчиком движения (ALT + мышь)

Как упоминалось ранее в этой главе, умный садовый светильник также был включен в подключенные устройства. IoT light был примером интеллектуального устройства, которое включает в себя собственные датчики и из-за сложности его программного обеспечения могло функционировать даже без заданных в бэкэнде условий. Блок уличного фонаря фактически имел встроенный датчик движения и датчик освещения; Это позволило автоматически включить лампу, если был обнаружен находящийся поблизости объект или обнаружен низкий уровень видимого света.

Наряду с концепцией «SaaS» вторым большим отличием, по сравнению с предыдущим моделированием умного дома, было использование конкретной переменной среды.

Как кратко объяснено в главе 4.2, Cisco Packet Tracer предлагает переменные для создания более реалистичных симуляций, которые на самом деле могут помочь повлиять на поведение датчика, но также подтверждают точность логики симуляции устройства IoT.

В этом упражнении переменные были использованы для того, чтобы влиять на выработку электроэнергии, повышение влажности в доме и уровень солнечного света для включения садовой лампы.

На приведенном ниже рисунке 33 представлен дневной сценарий. Наглядно показано, что солнечные панели производили электричество и заряжали аккумулятор, блок переменного тока был включен, из-за высокой влажности, а садовый свет был выключен. Затем на рисунке 34 представлен ночной сценарий с выключенным блоком переменного тока, без электричества и включенным садовым освещением.

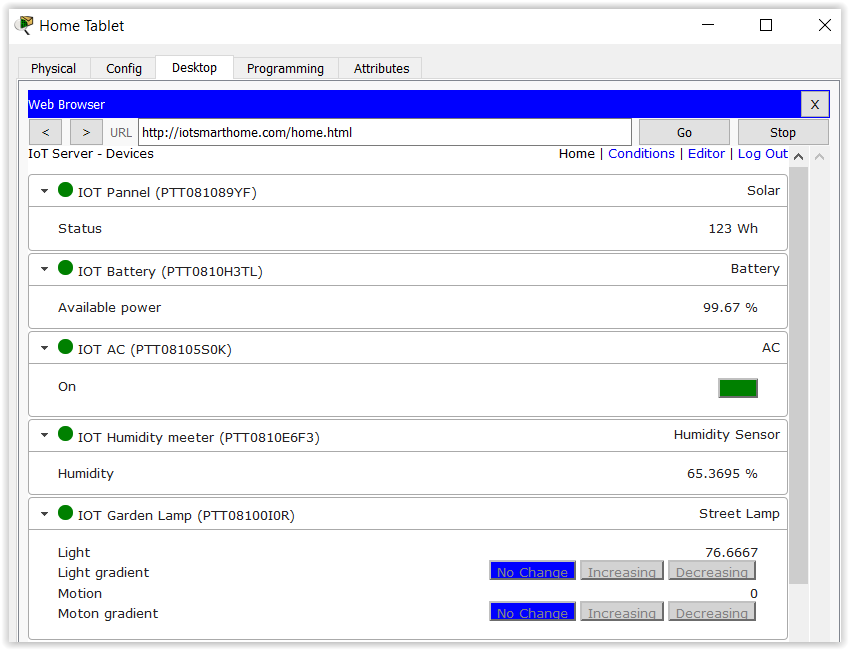


Рисунок 33 - Пример поведения оборудования в дневное время

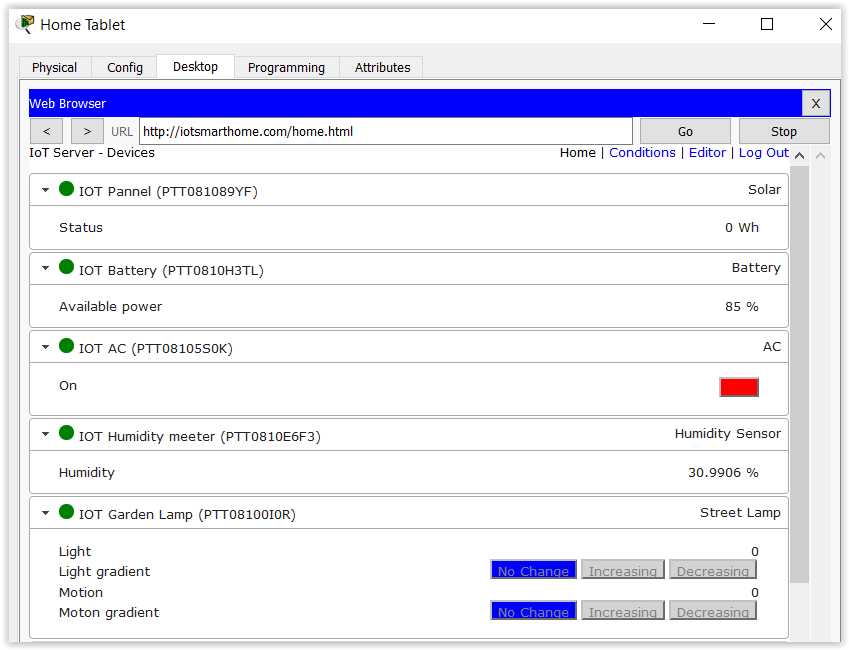


Рисунок 34 - Пример поведения оборудования в ночное время

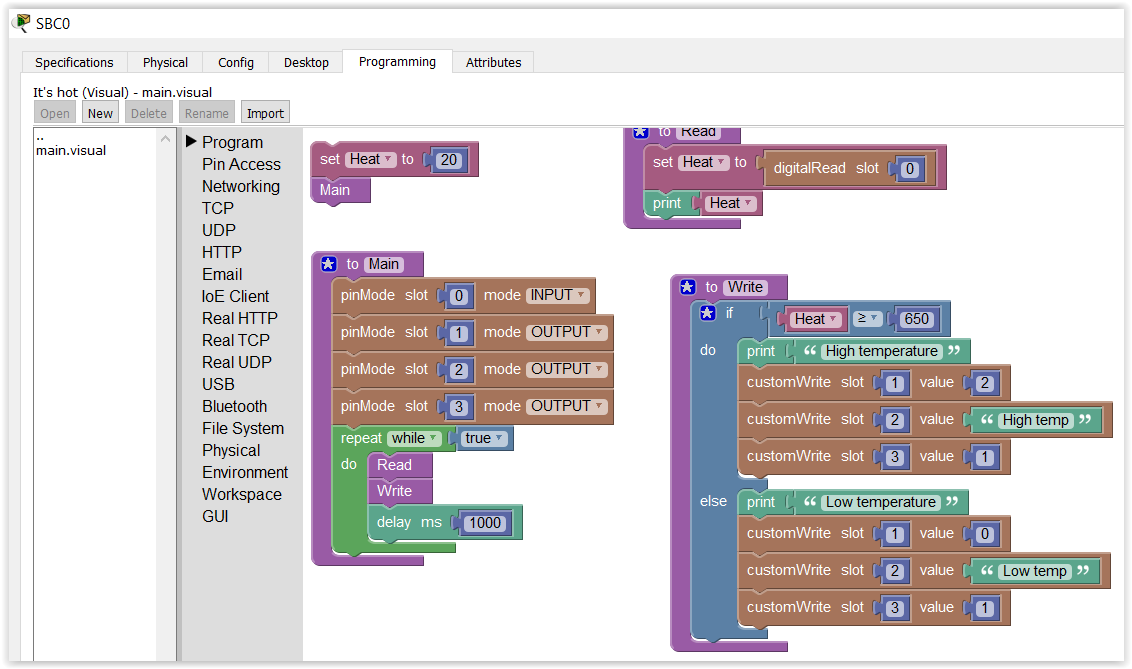
Пример IoT микроконтроллера

Один из примеров датчика IoT для привода также был включен в эту симуляцию Cisco Packet Tracer. Как и в предыдущем случае, плата SBC использовалась для того, чтобы подключить датчик к приводу без необходимости подключения к сети.

Как показано на рисунке 35, в этом сценарии датчик проверил температуру в физическом контейнере, включив вентилятор, если температура превышает заданное значение.

Для этого примера использовался неумный датчик IoT, который затем был подключен к входным контактам платы SBC. Абсолютное значение было отправлено датчиком из шкалы от 0 до 1023, затем пользовательская программа SBC обнаружила и сопоставила значение со стандартным показанием по Цельсию, сравнивая его в то же время с предварительно установленным условием. В случае выполнения условий SBC сигнализировал через выходные контакты, что вентилятор должен включиться. ЖК-экран также был подключен к выходным контактам SBC для отображения уровня температуры.

Блочную программу можно найти в Приложении 1.



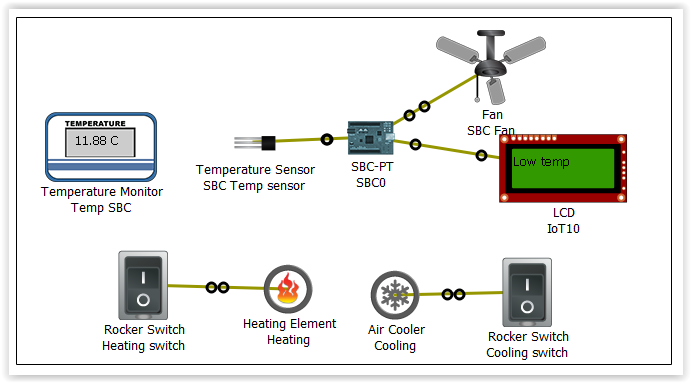


Рисунок 35 - Пример настройки микроконтроллера

Как объяснено в предыдущей главе 4.2, поведение датчика напрямую связано с переменными среды, для доказательства логики этого моделирования были добавлены нагревательные и охлаждающие элементы, чтобы быстро влиять на температуру окружающей среды контейнера. Эти элементы можно было бы быстро включить, нажав ALT на клавиатуре и нажав на подключенный тумблер.

Как показано на рисунке 26, когда нагревательный элемент был включен, температура быстро поднимается, включается вентилятор

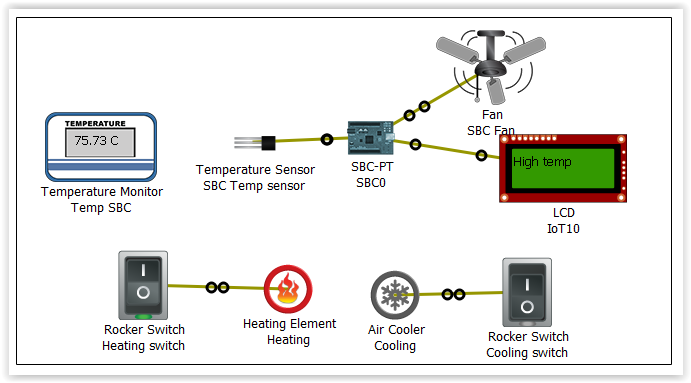


Рисунок 36 - Пример с включенным нагревательным элементом