Проблема

На всех предприятиях располагается огромное количество осветительных приборов. Но не всегда получается отследить работоспособность, так как охват осмотра огромен. В этом случае может помочь автоматизированный датчик тока. Но при поиске в интернете находятся либо целые комплексы датчик, что нагромождает работу, либо датчики-счетчики, которые также не подходят к решению этой проблемы.

Решение задачи

Решение найденной проблемы заключается в реализации проекта системы контроля осветительных приборов, который позволил бы задать рабочее и ночное время работы и отслеживать работоспособность элементов.

Функциональные требования

Проект должен обладать следующим функционалом:

– Функция отображения элементов и их работоспособности;

– Функция запроса данных;

– Функция задания ночного и дневного времени работы;

Эксплуатационные требования

Проект должен соответствовать следующим требованиям:

– Простота эксплуатирования и настройки устройства;

– Минимальное количество оборудования;

– Автономность;

Используемое оборудование и его стоимость

Для реализации прототипа поставленной задачи была использована следующая аппаратная составляющая:

– Базовая станция, реализованная на STM32L151CCU6 (1650 рублей, если в корпусе Unwired Range), UMDK-RF (1290 рублей), UMDK-GATE/E и на основе микрокомпьютера Black Swift/Unwired One (1800 рублей);

– Датчик тока ACS712 (180 рублей);

– Лампочка светодиодная (140 рублей);

– Элемент связи датчика со станцией, реализованный на Unwired Range (1650 рублей), UMDK-RF (1290 рублей);

Общая стоимость без учета UMDK-GATE/E составляет примерно 8000 рублей

Оценка энергопотребления прототипа

– Базовая станция ≈

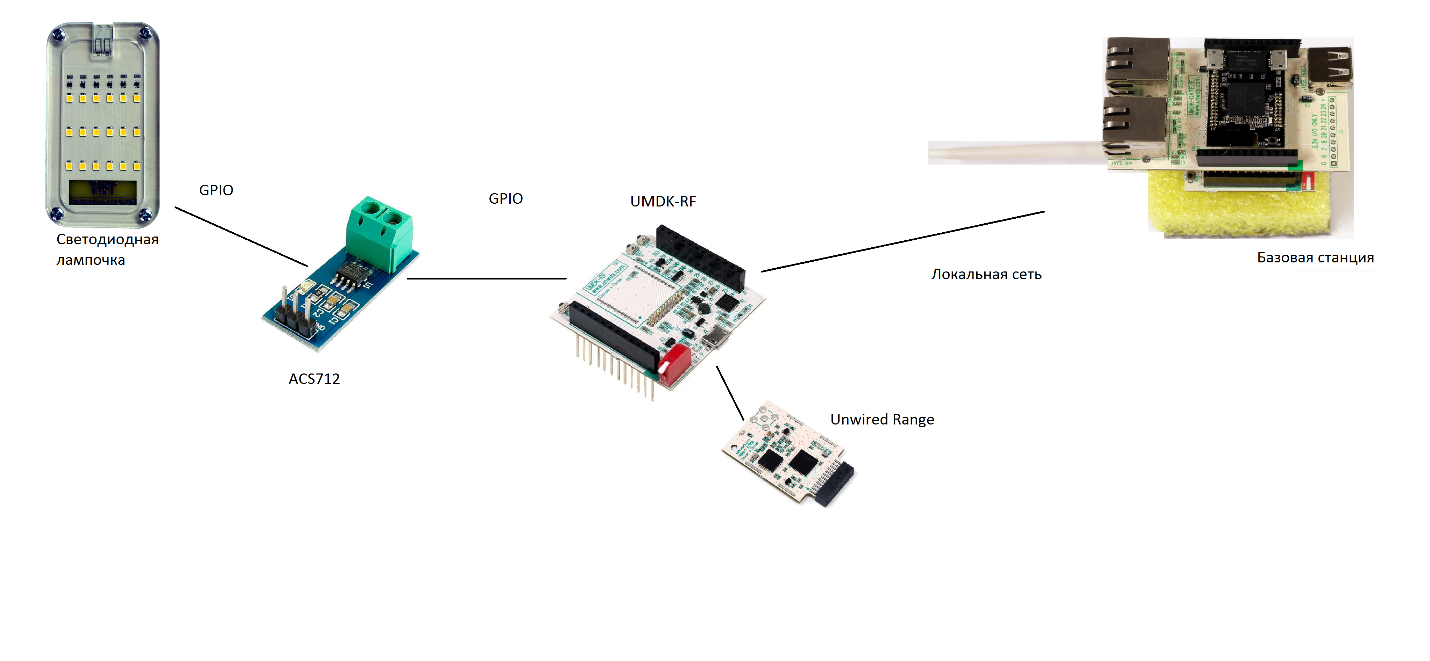
– Датчик тока ACS712 ≈ 11 мА

– Светодиодная лампа ≈ 6\*60мА ≈ 360 мА

Общая сумма ≈ 375 мА

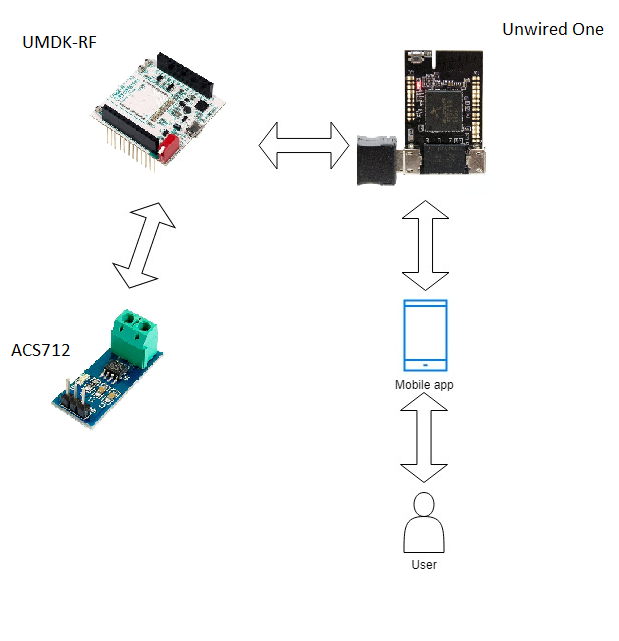
Взаимодействие оборудования и логика работы

К элементу связи подключен датчик тока, который размещен в цепи лампочки. Элемент связи берет данные датчика тока ACS712, подключенного по цифровым входам, преобразовывает их и отправляет на базовую станцию по локальной сети, которая является сервером, по MQTT протоколу.



Архитектура системы

Взаимодействие пользователя с устройством происходит следующим образом: датчик тока ACS712 передает данные UMDK-RF, которая отправляет по локальной сети данные на базовую станцию. Из нее отправляются данные на приложение с помощью MQTT протока, где они обрабатываются и выдаются пользователю. При отправке данных пользователем, происходит обратный путь до UMDK-RF

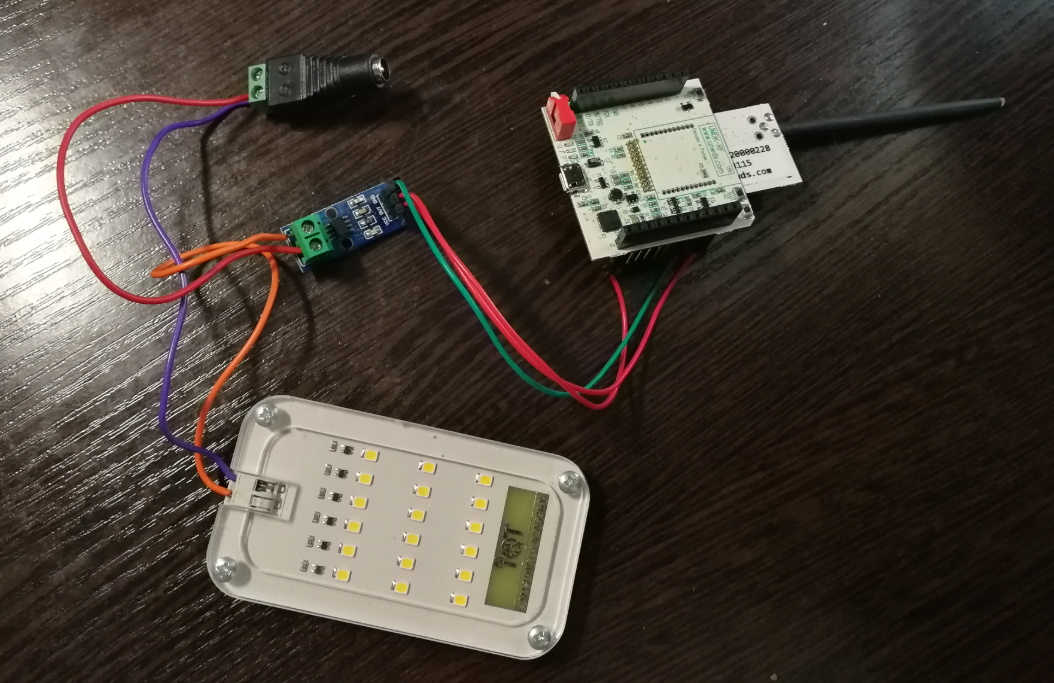


Программная реализация

Прошивки базовой станции и элемента связи были взяты из пройденных курсов IoT SAMSUNG ([**https://github.com/unwireddevices/stm32flash**](https://github.com/unwireddevices/stm32flash)**)**

Для написания мобильного приложения использовался язык Java, для отправки MQTT-запросов, использовалась библиотека eclipse/pah.mqtt.java (<https://github.com/eclipse/paho.mqtt.java>).

Внешний вид основного оборудования



Мобильное приложение

Мобильное приложение поделено на 2 основных фрагмента:

– Главная страница;

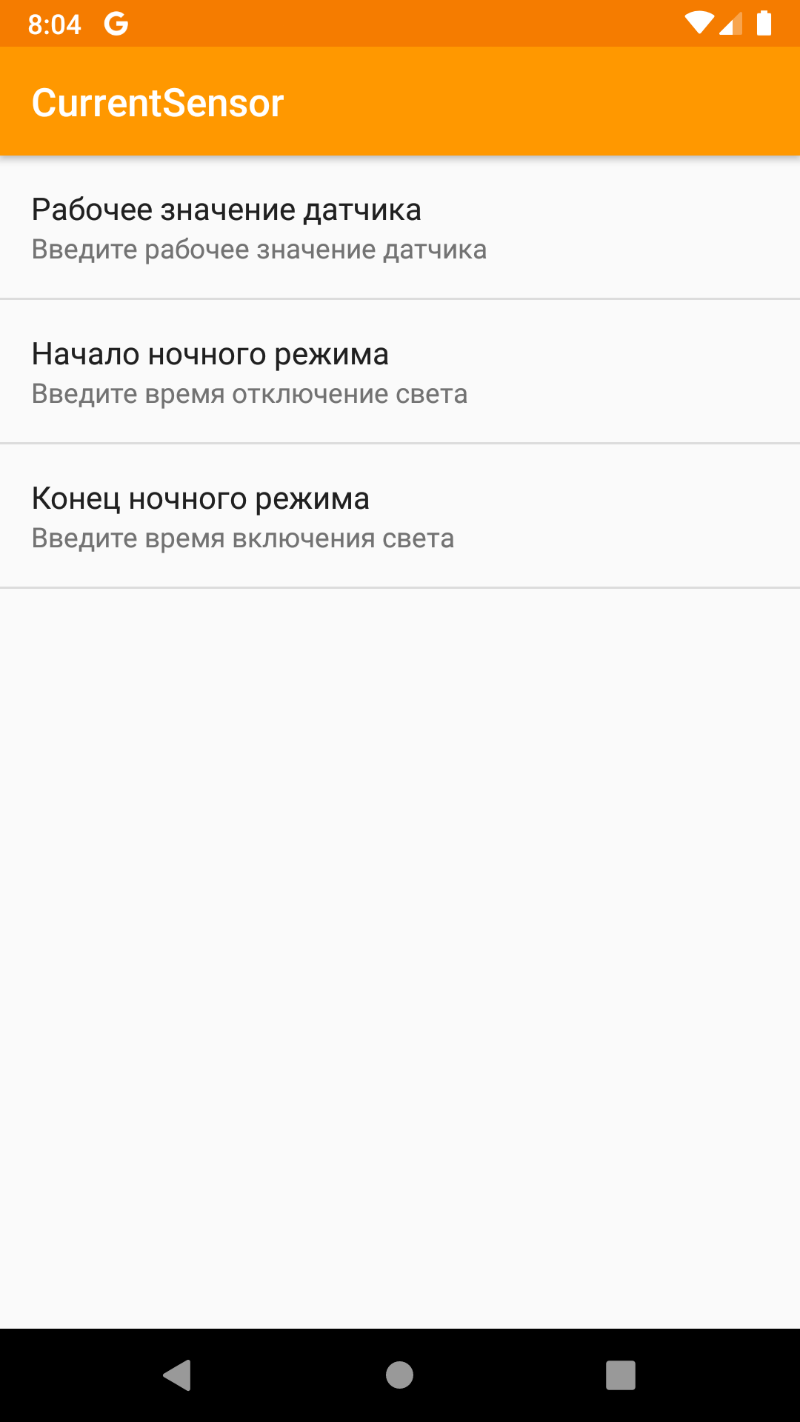
– Меню настроек;

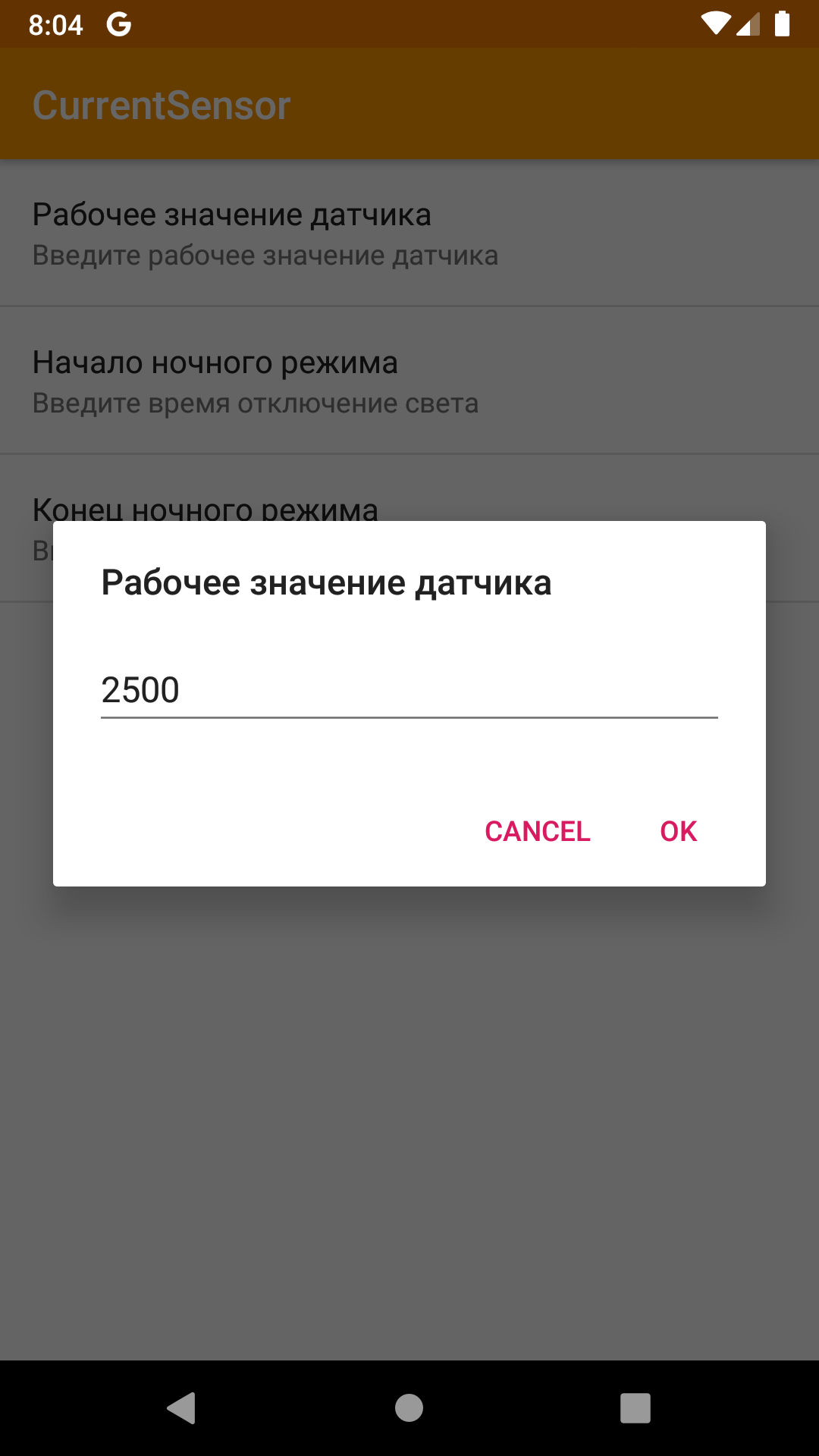
Приложение изначально подписано на базовую станцию, поэтому при запуске приложения, пользователь сразу получает данные со станции.





На главной странице пользователь может увидеть список всех элементов, за которыми ведется контроль, а также их работоспособность. В правом углу главной страницы есть меню, где пользователь может запросить данные со станции или же перейти во вкладку настроек.





В меню настроек пользователю предоставляется возможность установить среднее рабочее состояние датчика тока и настроить время ночного режима, т.е. время, когда датчик следит работает ли свет или он выключен.

Исходный код приложения находится на GitHub’е ([https://github.com/Gabslocks/CurrentSensor)](https://github.com/Gabslocks/CurrentSensor)/).

Заключение

Был разработан прототип контроля осветительных приборов, в которой есть возможность задать ночной режим работы, реализован контроль за элементами в цепи тока, реализовано общение через MQTT протокол и есть удобный доступ через мобильное приложение.