Swakeup

PROJECT PROPOSAL

within the lecture of
Programming Embedded Systems

at Uppsala University
in the Departement of Information Technology

Elmar van Rijnswou (Elmar.Vanrijnswou.9818@student.uu.se), Maximilian Stiefel (Maximilian.Stiefel.8233@student.uu.se) and Nikolay Georgiev (Nikolay.Georgiev.2061@student.uu.se)
Deadline: 2017-03-29 24:00

Processing Period:	March 2017 - May 2017
Supervisor:	Philipp Rümmer (philipp.ruemmer@it.uu.se)

Contents

1 General Description **1**
1.1 Current state 1

2 Objectives and Requirements **2**

3 Organization **4**

1 General Description

It is a well-known fact, that it is quite dark in Sweden in the winter. In a strong winter every source of light is a source of happiness. This wakeup light, which is based on a strong light source (10 W RGB LED), is able to give one the optimal start into a dark winter day. The *Swakeup* (from engl. "Swedish Wakeup Light") is communicating to the user through the light. It does not simply wake one up, but also gives one information about Facebook, latest mails, calendar and weather. The user interface consits besides of a big LED of an OLED screen. *Swakeup* is also part of the *IoT* as it has the ability to communicate via *IEEE 802.11*. This of course enables a lot of possibilities e.g. connecting your phone to the wakeup light. A lot of effort has been put into the designing maxim, that everything should be as small as possible. The whole electronics fit on an base area of 5 cm x 4 cm. So the *Swakeup* fits smoothly on the bedside table. And honestly: What is the last thing you are doing before you go to sleep? Right! You look on your phone. That is why *Swakeup* comes with a USB charger for your e.g. phone as well.

1.1 Current state

As this project is a continuation from a project of another course, certain steps have already been made. A first version of hardware is developed, containing hardware mistakes. The hardware consists out of 2 processors, an energy efficient Atmel xmega, taking care of the control loop and acts as interface for the screen. And an ESP8266 which will act as a gateway to the internet. No work has been done on the ESP8266 side yet, neither has communication been set up. The control loop that should take care of setting the LED to the right brightness is also not been implemented yet. For a more in-depth description of the current state see https://github.com/s3xm3x/SwakeUp.

2 Objectives and Requirements

As stated previously, some work still has to be done in order to get the basic functionality running. Therefor it's crucial that the hardware will be upgraded with new PCB's and components, the control loop implemented and visualization on the screen to be updated to reflect the social media interaction better. Furthermore, no communication exists yet between the two microcontrollers. A protocol has to be researched, and implemented. All these tasks are required for the xmega. Another part of the project will be the programming of the ESP8266, which will involve more time. First off, a programming environment and language will have to be selected for the ESP8266, as there are many different languages and frameworks to program it in. As soon a suitable framework has been found, implementation of the required functions can begin. The ESP8266 should be able to communicate with the xmega, be able to act as an access point when it can't connect to another access point, receive the current time from the Internet, download emails from the user and if time allows integrate with social media platforms. Below in fig. 2.1 a small diagram can be seen of what will have to be realized in order to get a working first version.

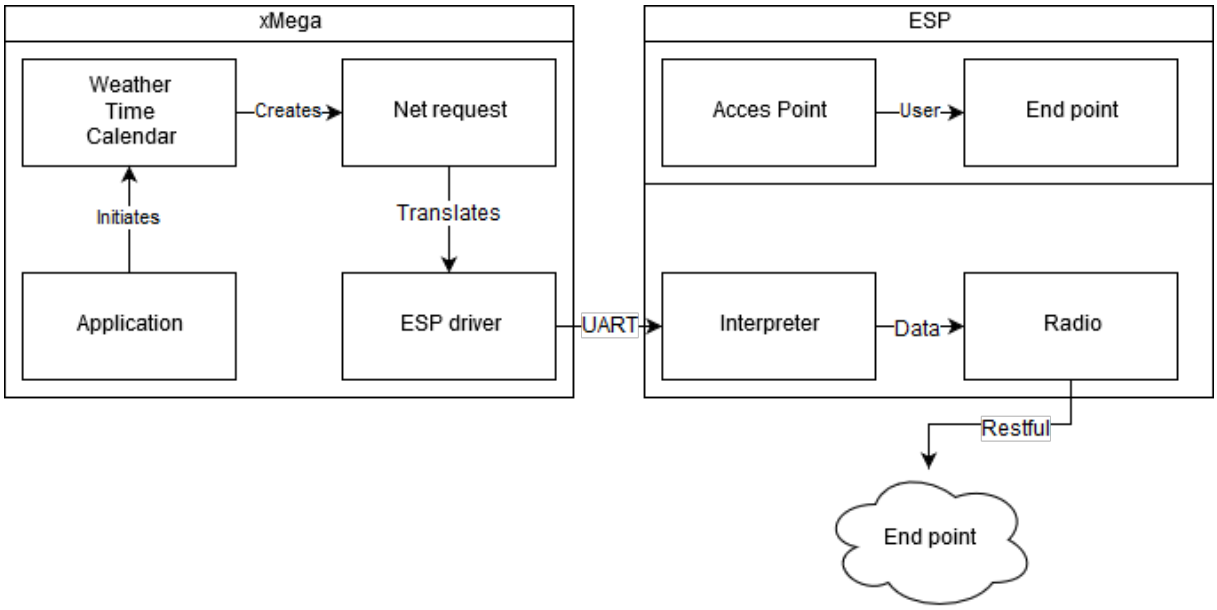

Figure 2.1: Rough sketch of the requirements.

The application part will activate one or multiple sub-systems such as the weather/calendar or time sub system. If either of those systems require an update, they will create a new net request. This request will be a data structure and will be passed to the ESP8266 driver. Here the data structure will be translated into the physical layer and sent via UART. Once this arrives at the ESP8266 it will have to be interpreted, and if needed data will be sent and or received from the radio. The radio will be in contact with microservices that are ran on a SaaS solution where it will be able to collect data from facebook/email/twitter. Other restful requests will be made to acquire the newest time and the latest weather.
Alongside this the ESP8266 will also act as an endpoint and access point. So that users will be able to set up a connection on the ESP8266 itself via a default route to a webpage hosted on the ESP8266.
Research has to be conducted on the requirements of getting information of one of theses social media

sites. As well as the encryption that goes with it. How the got the most accurate time from the internet. To detect a location from a local network so the user won't have to select that itself. As well as how to generate restful requests and how to handle HTTPS traffic. Furthermore a webserver will have to be constructed on the ESP8266, and a flexible communication protocol between the two chips will have to be selected and implemented. Lastly a well organized structure should be kept on the created operating system.

3 Organization

Multiple dedicated days have been allocated on which the team shall gather, and discuss progress and problems that have been run into. To communicate during other days, telegram will be used to talk. The tasks will be divided equally among all the members. To allow for seamless collaboration, version control will be used. The whole project is hosted on github where issues can be made, and progress can be committed.