Smart Green House

Alexander Platz
Aravinth, S. Panchadcharam
Florian David Roubal
Martin Kessel
Sven Erik Jeroschewski

Telecommunication Networks Group
Technische Universität Berlin

Agenda

- Project Progress
- Communication
- Architecture
  - Sensor Node & Sensors
  - Actuator Node & Actuators
  - Control Center
- Roadmap
- Live demo

Project Progress

Motivation

You live far away from home and no one is there to take care of your plants?

You are a busy student and you don’t have enough time to water your plants?

Our Smart Green House can help you!

Solution

The Smart green house …

- is able to water several plants automatically
- controls perfect illumination
- is scalable for a lot of plants
- user can interact with the system

Work Split Up

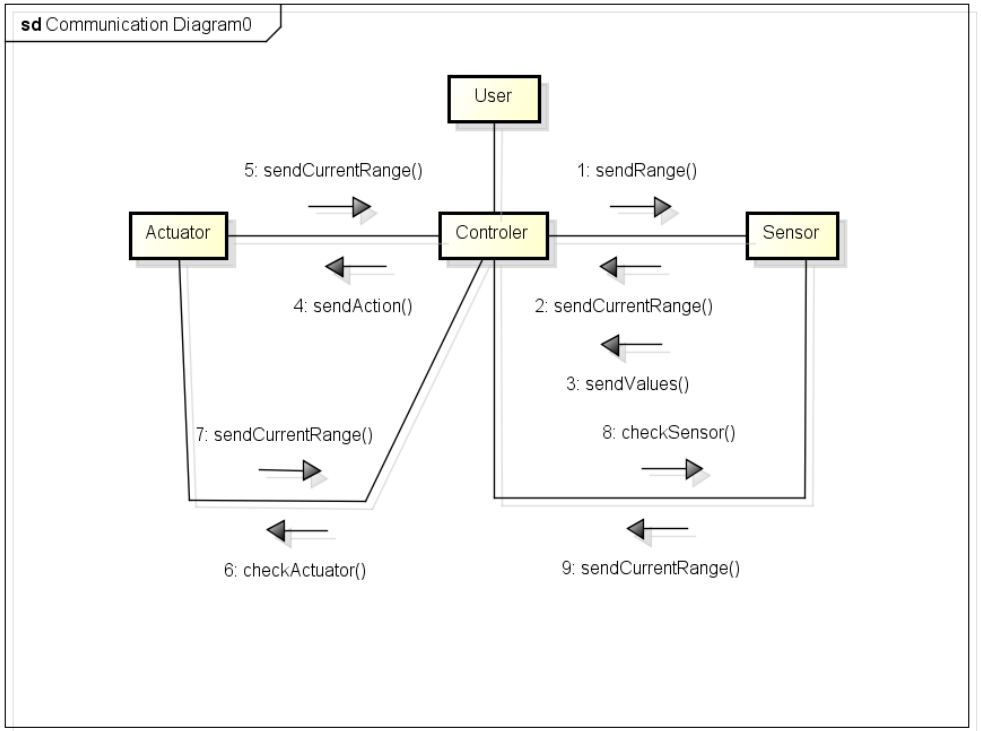
- Sensor Node & Sensors
  - Martin Kessel
- Communication
  - Sven Erik Jeroschewski
- Actuator Node & Actuators
  - Florian David Roubal
- Control Center & UI
  - Alexander Platz
  - Aravinth, S. Panchadcharam

Communication

Communication

- principle: main computing should be done in the controller
- message learned: 2,4 GHz is not 868 MHz ! (cost us hours)
- communication is only possible via strings!
- convert the strings into integers can be a lot of work

Communication



Architecture

Sensor Node & Sensors

Sensor Node & Sensors

Software principle:

- Check if message received:
  → yes: Update range values
- Read sensor values
- Decide if sending necessary to save energy
  → Only if values out of range
- Sleep for a in code definable time
- Wake up via RTC Interrupt and repeat cycle

Sensor Node & Sensors

Sensors and Values:

DHT11 Temperature and Humidity:

- Temperature in Celsius
- Humiditiy in %

Soil moisture sensor connected via ADC:

- Value between 0 and 1023
- Dry = High value, Wet = value decreases

Light sensor connected via I2C:

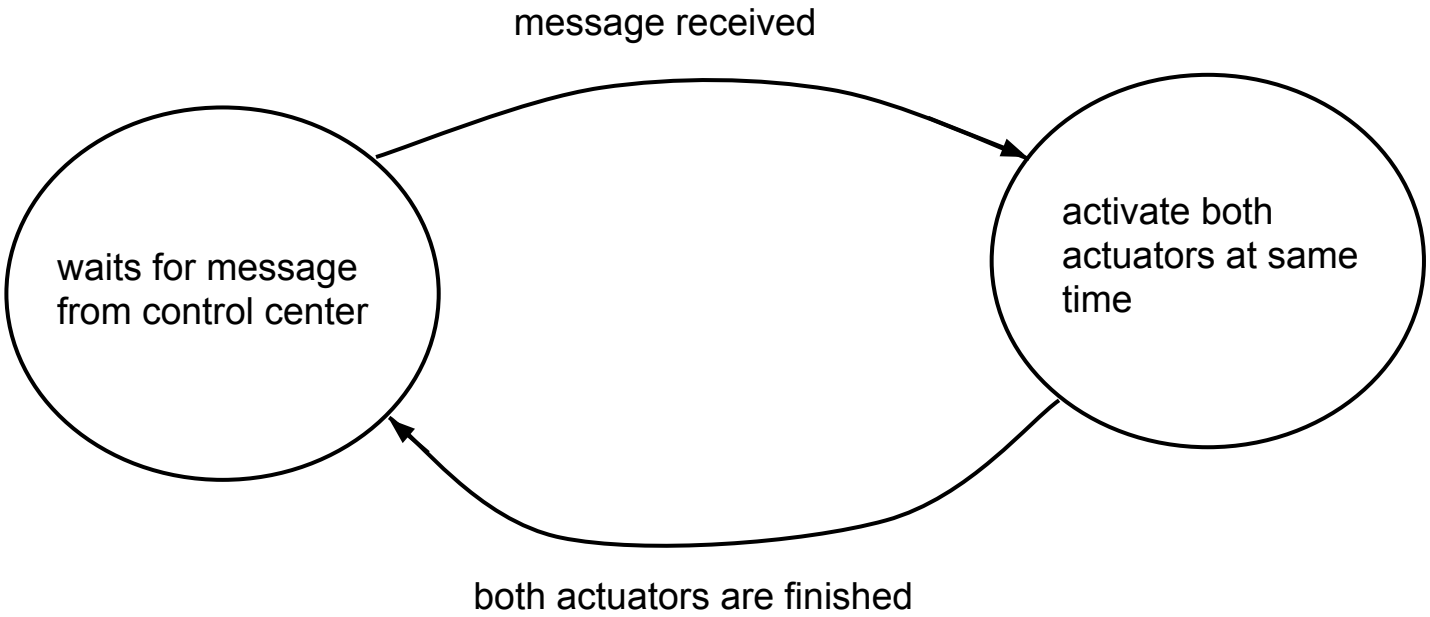
- Luminous flux per area = LUX

Architecture

Actuator Node & Actuators

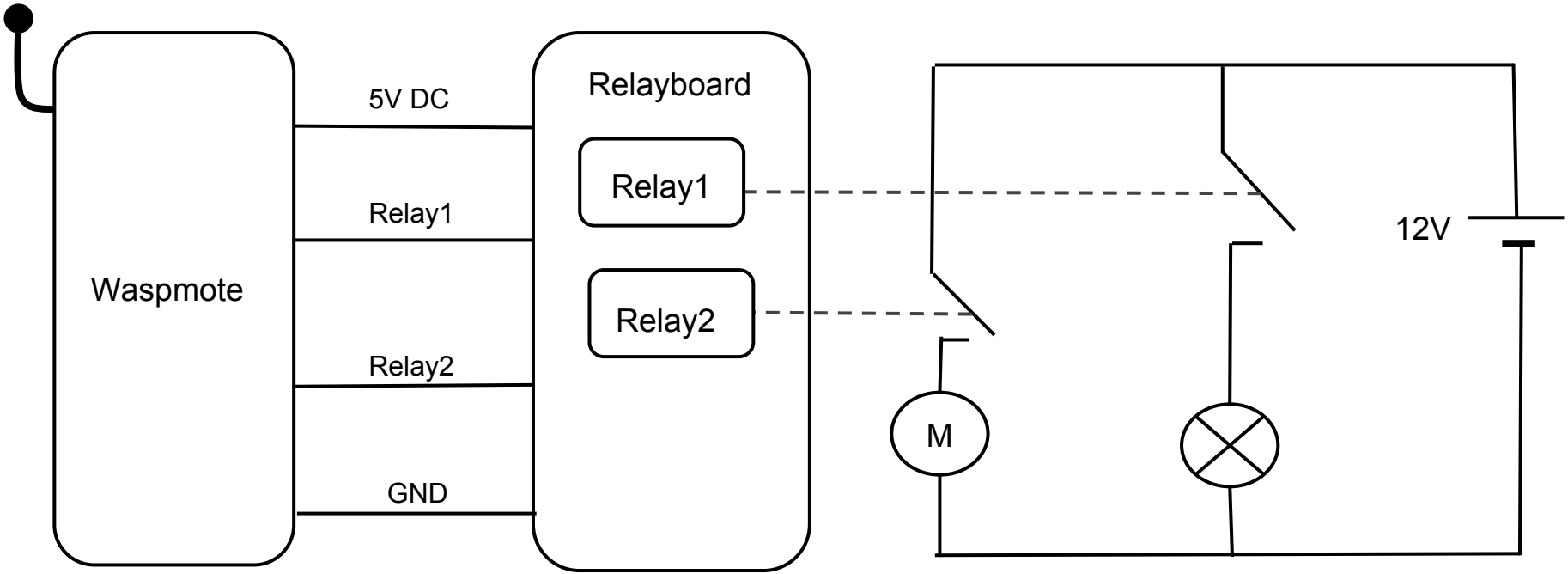
Actuator node & actuator

- Software Components



Actuator node & actuator

- Hardware Components



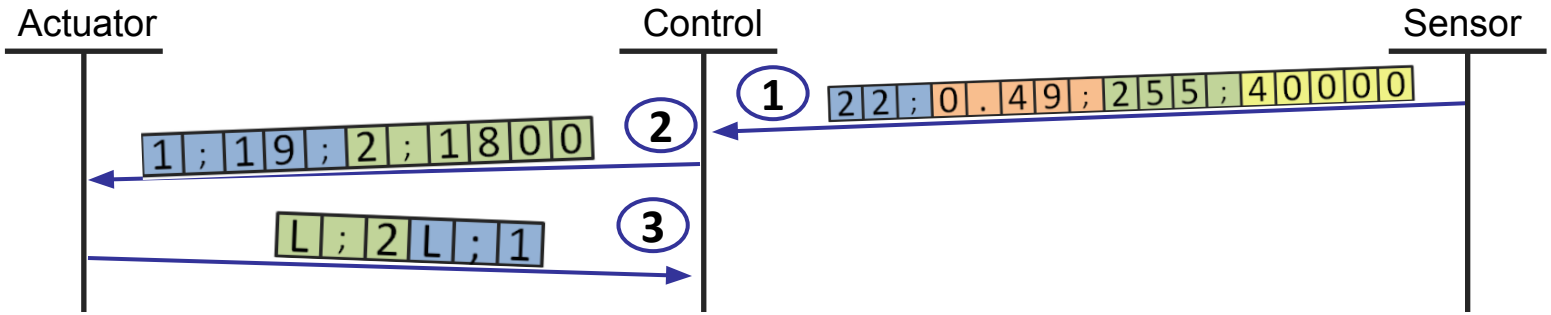
Presentation Mode

- Sender
  - sends periodically
  - message: {Pump on for 3 seconds ; lamp on for 1 second}
- Receiver
  - waits for message from sender
  - activates both actuators
  - sets timer: pump timer = 3 seconds, lamp timer = 1 second
  - waits until both timers expired
  - waits again for message

Architecture

Control Center

Processing



1. Processing incoming sensor data: **temperature, air humidity**, **soil moisture**, **light**
   Calculating uptime for actuators: **water pump**, **lighting**
2. Sending ON-Signal and uptime duration to: **Actuator-water pump** / **Actuator-lighting**
   Setting boolean variable TRUE (water pump / lighting is on)
3. Receiving OFF-Signal from actuator
   Setting boolean variable FALSE (water pump / lighting is off)

Robustness / Troubleshooting

- Control Center ⟷ Sensor Node
  - CC needs to check whether Sensor Node is still operating (f. e. node discovery or searching specific nodes every hour)

- Control Center ⟷ Actuator Node

  Double Check uptime of actuators (CC and actuator node know the uptime, if OFF-Signal not received within time, check what is wrong)

- Checking and evaluate errors
  - Error flags: `error_AT; error_RX; error_TX` [868Guide, sec 2.2]
  - Getting Errors:
    - `xbee868.getRFerrors();` [868Guide, sec 7.3.1]
    - `xbee868.getTransmisionErrors();` [868Guide, sec 7.3.4]

- Retry mechanism: `xbee868.setMacRetries(0x0A);` [868Guide, sec 7.3.10]

Roadmap

final sprint

Roadmap 1/2

- **Communication: Control Center → Actuator Node**
  - Control Center sends ON-Signal and uptime duration to actuator
  - Actuator node reacts and switches on it's two power outputs in the relay shield for the given time
- **Communication: Sensor Node → Control Center**
  - Sensor Node sends the values of the connected sensors to the Control Center
- **Implementing robustness**
  - Control Center checks whether nodes are still operating
  - Double check whether the actuator node has switched off it's actuators
  - API Error handling, retry mechanism

Roadmap 2/2

- **Energy saving**
  - Switch Waspmote to low-power mode if there is nothing to process
  - Switching off the XBee Antenna if communication is not needed
- **Interrupts / Timers**
  - Programme timer interrupts with Real Time Clock (RTC)
  - Interrupts via sensors - but perhaps direct connected sensors at some pins can’t interrupt
  - Interrupts via XBee antenna - investigate if that is possible

Live demo

Bibliography

[868Guide] http://www.libelium.com/v11-files/documentation/waspmote/waspmote-868-networking_guide.pdf

Thank you for your attention.