WASP

Wireless Arduino Sensor Protocol

Group SW513E15

Christian Lundtofte
Henrik Djernes Thomsen
Jonathan Hastrup

Bjørn Opstad
Morten Mandrup
Mathias Corlin

AALBORG UNIVERSITY
STUDENT REPORT

Department of computer science
Selma Lagerlöfs Vej 300
9220 Aalborg Ø

Title:
WASP - Wireless Arduino Sensor Protocol

Theme:
Embedded Systems

Project period:
02/08/2015
21/12/2015

Project group:
SW513E15

Members:
Christian Lundtofte Sørensen
Henrik Djernes Thomsen
Jonathan Hastrup
Bjørn Opstad
Morten Mandrup Hansen
Mathias Corlin

Supervisor:
Hua Lu

No. printed Copies: ?
No. of Pages: ?
No. of Appendix Pages: ?
Total no. of pages: ?
Completed: 21/12/2015

Synopsis:

Synopsis her!

Bjørn Opstad

Christian Lundtofte

Morten Mandrup

Henrik Thomsen

Matihas Corlin

Jonathan Hastrup

Contents

1. Project introduction

This is an introduction.

Communication technology can today be seen as a foundation for wide range of technologies. Devices today can communicate with each other in many different ways, and can take part in large and small networks alike. This report resolves around embedded systems and how communication can be established between these. This focus point is quite broad in terms where it can be applied, so this report will make use of a specific usecase to showcase the communication between embedded systems its applied usage.

The usecase that will be in focus throughout the report is the moisture of a golf course, where multiple devices will relay information about the earth's moisture levels to a central unit. If a device is placed out of range from the central unit, it will need to send the its data reading to the central unit by using other devices within reach.

To ensure that the data will reach central unit, and ensuring that the data is still correct/usable(?), a sufficient protocol must be found/established to verify this. The protocol must be able executable on the embedded devices.

1.1 Initializing problem statement

How can a sensor network and a protocol be designed, so that data can be relayed throughout the network, enabling an endpoint device to receive the information without being within range of all sensors in the network?

Part I

Analysis

The analysis will discuss and look into the different aspects of the initializing problem formulation and the topics therein. The sections in this chapter blahblablah..

fix this meta

1.2 Interview

Two interviews were performed with a greenkeeper at Aalborg Golfcourse, Kim Jensen. Kim provided insights on ideas for the system, requirements, potential problems, and additions for later iterations of the system. The interview questions and answers can be found in Appendix A.

Kim provided the initial idea of automating the gathering of data about the golf course. He suggested soil moisture, pH value in the soil, and pressure. These are currently done manually, and could be automated to save time for the greenkeepers.

Some holes on Aalborg Golfklub is located in a swampy area, which often makes watering them obsolete. This project could help stop some of this water waste. Some holes are in a moor, where watering is often required. When it is required, this project could inform the greenkeepers, who could act on this.

Making the devices hot pluggable would be useful if there is suddenly more or less water than usually, in which case a new device can be added to this location, and the water levels can be measured to determine if there is a problem, and in that case how to solve it. The same applies if an area is more dry than usual.

An important consideration is where to place the devices. The sensors needs access to the soil in order to measure the moisture levels, but they should not be placed above ground as golfers could hit and destroy the devices. The initial idea was to place it in the holes for the sprinklers, but this is bad since sprinklers often leak, and the sensor would therefore get inaccurate readings. This makes it necessary to bury the devices at known locations in the course. The depth the devices are buried at is important, due to the different types of grass. The greens use a layer of sand at the top, which allows water to quickly go through. This makes it necessary to place the sensor at around 10-15cm's. in the soil.

In future iterations of the project, the other sensors mentioned in the interview could be implemented to save even more work. The pH-meter would allow the greenkeepers to create specific mixes of fertilizer for different parts of the course, depending on the pH value of the soil. The pressure sensor could help determining where on the course the soil has to be pricked, in order to keep the grass growing and water flowing.

1.3 Golf course

In this section, the physical and technical aspects of a golf course will be described. These factors are analyzed to be able to define requirements for the solution and to formulate a problem statement.

Firstly, the physical and geographical aspect will be examined. Thereafter, special elements of the task are studied to give grounds for choice of possible hardware for the solution.

1.3.1 Geography/topography

This subsection will contain well formulated descriptions and analysis of the following: A golf course is typically big, and has up to 18 holes. It has people walking all over, and there's a chance for objects on it getting hit by golf balls. They are mainly located outside and are accordingly susceptible to weather. They are used frequently, but it is not critical for the whole course to always be operational (digging cables for sensors). Golf courses are static by nature and changes to the region are organizationally applied.

1.3.2 Soil properties

A part of the task of maintaining the soil, is to consider the soils moisture. This can be measured by several methods, for example by visible indications or direct assessment by a sensor. Currently, in accordance with the interview with Kim, this is monitored by using visual aids and indications. Soil mostly consists of dirt and sand.

1.4 Other issues

Some other potential issues not directly encountered through the analysis, but which could still be relevant, will be presented here.

1.4.1 Security

In most softwaresystems it is important to evaluate the security, to avoid unauthorized access or data loss/theft. This project is not excluded from such evaluation, and in this section the greatest security issues will be considered.

The main functions of this project is to exchange data throughout a network wirelessly. Since the data is sent wirelessly, it can easily be manipulated or interfered resulting in incorrect data.

Another view is the process of connecting a new device to the network. If a device is connected too easy, it could become a problem if a neighbouring network is using the same protocol which could result in the two networks exchanging data unintentionally. Another point is that a malicious device could easily inject data into the system.

Considering the hardware design of a measuring device, it would also be relevant to analyze if a device can be physically reconstructed to intentionally transmit malicious data to the network. Though being more a question of physical security it could be an aspect in the design phase.

There could potentially be a lot of minor security issues, but the major issue in this project is that data is exchanged wirelessly and it is therefore very adaptive to malicious input. This must be taken into account in the design of a final solution.

1.4.2 User interface

As the problem is to transfer data from sensors to some point, where a user can read and react appropiatly to it, it would be necessary to present this data for the user. As the typical human being is unable to read bytestreams, a userinterface with some more informal way of presenting this data is a practical thing.

The amount of interaction would not need to be excessive, as the system would only serve to feed the user with data, and as such a simple interface for oneway communicating the data to the user could be enough.

2. MoSCoW

This is a roundup of the analysis chapter, and will be the transition into the technology chapter. This chapter aims to map the requirements of the project and sort them based on different priorities. First a full listing with the requirements and their priorities, followed by a description for the contents of each category.

Requirements are an essential part of developing a system that fulfill its purpose. All requirements can be considered important, but some might be more beneficial for systems core purpose than others. By categorizing the requirements it becomes possible to identify and separate the important criteria from the lesser ones.

The MoSCoW (Must-have, Should-have, Cant-have and Wont-have) analysis method have been used to sort the requirements. The requirements have are the result of the this reports analysis chapter. The Wont-have category have not been used. All the requirements are summarized below:

Must have:

- Transfer data from all sensor nodes to a destination/main node wirelessly.
- Implement a network with a communication protocol.
- The system should respond appropriately to a disconnecting node.
- Nodes equipped with a sensor.

Should have:

- Be able to seamlessly add and remove sensors to an existing system (Hot-pluggging).
- Be modular build, so parts of a node can be replaced if destroyed.

Could have:

- Separate system installations, so nodes do not connect to another installation on a nearby golf course.
- Have a graphical user interface for presentation of data.

Each category have been expanded and further explained below:

Must have:

- Transfer data from all sensor nodes to a destination/main node wirelessly.

The core purpose of the system is to transfer data between nodes. Without this the system would not be able to fulfill its purpose and this requirement have therefore been categorized as a must have.

- Implement a network with a communication protocol.

The system will transfer data between nodes wirelessly and it is therefore required of each node that they can send and receive readable data. A communication protocol is needed to ensure that the data is treated correctly.

- The system should respond appropriately to a disconnecting node.

When multiple nodes have been spread around the golf course, they will communicate wirelessly, and since each node are dependent on each other to reach the main/end node, it becomes necessary to ensure that the entire system does not break down if one node were to disconnect from the rest. How would the system administrators find the disconnected node? This is why the system needs some sort of fail-safe or a mechanism to remain functional if a node were to disconnect.

- Nodes equipped with a sensor.

The nodes scattered around the golf course need to collect some data to transfer back to the main node/unit. To show a proof of concept, the only thing the system need is some arbitrary data about the golf course to transmit between the nodes. Based on the interview, there are three types sensors that collect relevant data for the system. The analysis of the sensors can be found in the technology chapter of this report.

Should have:

- Be able to seamlessly add and remove sensors to an existing system (Hot-pluggging).

The requirement aims to add flexibility both during the installation of the system, and for further configuration thereafter. This would enable the addition, removal, and re-positioning of nodes without requiring a restart or re-configuration of the entire system. The must-have requirement, concerning a fail-safe in case of a disconnecting note, would in addition be met if a hot-plugging system were to be implemented.

- Be modular built, so parts of a node can be replaced if destroyed.

This requirement is referring to the node/embedded system unit itself. The embedded system will consist of multiple parts, such as a micro controller, a sensor, and a transmitter. These parts should be put together in such a way that if a part breaks down, it would be an easy task to replace it. Since the devices will be placed outside on an actual golf course, the devices will exposed to weather and golf balls and is therefore prone to taking damage, it makes this requirement important an important one but not a must.

Could have:

- Separate system installations, so nodes do not connect to another installation on a nearby golf course.

If the final system includes the hot-pluggable functionality, the individual nodes should stay connected to a particular installation and not be confused by another nearby installation of the system. This requirement could potentially become very complex, and the problem with two installations that interfere with each other might not be relevant since a single installation should cover the entire golf course.

- Have a graphical user interface for presentation of data.

The data collected by the system should be processed by the central node/unit and presented for the system administrators through a graphical user interface. This could be done with a monitor connected to the central node/unit or through a smartphone application. The graphical user interface should also allow the system administrators to change the settings of the system, e.g. change the interval between sensor readings.

3. Problem Statement

Very good problem statement for you, my friend. Special prize.

Make a good sending data network for arduino.

3.1 Requirements

There are some requirements to the system and its software. These are split in two categories: functional and non-functional. This is based on some smart guys work [**keylist**].

3.1.1 Functional requirements

The list of functional requirements:

1. Actually run is an important part to passing the exam

3.1.2 Non-functional requirements

List of non-functional requirements:

1. Looking good is not a bad thing.

There are multiple technologies available to utilize in a solution. This chapter will contain information, descriptions and examinations of the considered technologies, as well as definitions used in the report.

Firstly, the controller units for the sensors will be described. These are necessary to process the information from the sensors and

Secondly, sensor devices are studied to be able to choose an adequate unit for the solution.

Thirdly, network theory and topology are explained to be able to make an informed decision for the solution structure.

Fourthly, communication devices are examined to select an appropriate unit to use in the solution.

Lastly, analysis of communication protocols.

4.1 Arduino

Arduino is an open source platform, which makes the software and hardware documentation available to the public. This is also why the Arduino platform is often used at school for learning about electronics. relevant?

The name "Arduino" covers both the software platform and the range of hardware platforms with boards of different sizes, from the smaller Arduino Nano up to the Arduino Mega. One of the more popular Arduino boards is the Arduino Uno which is a medium board, considering it's size and power relative to the currently available boards.

Arduino boards have a setup of input and output ports enabling it to read from a sensor, or a button and then maybeactivate a motor or an LED light. How the Arduino handles or reacts to input is up to the designer which can program the Arduino board using the Arduino IDE. what? maybe, my ass source

In the following sections, the Uno and Mega, which will be used in this project, will be described. spoiler?

4.1.1 Arduino Uno

One of the most common boards is the Arduino Uno 4.1, which is based on the ATmega328 microcontroller. It has 14 digital input/output pins, and 6 analog inputs for connecting the different components. Considering specifications, which is shown in table 4.1.1, the Uno is limited on its resources. Therefore it is needed to limit both program- and datasize, and also the amount of complex tasks. source

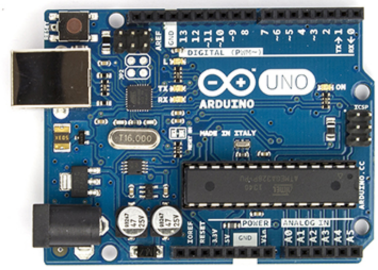

Figure 4.1: The Arduino Uno board[1].

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB is used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Table 4.1: Specifications for the Arduino Uno.

source

4.1.2 Arduino Mega

The Arduino Mega is a larger version of the Uno. The Mega has more memory and pins, which makes it better for handling larger programs or amounts of data. This also allows more components can be connected to the board. Since the clock speed is the same as the Uno, the Mega will not process data faster.

source

4.2 Raspberry Pi

Raspberry Pi is a series of single board computers. The Raspberry Pi is developed by the Raspberry Pi Foundation seated in the UK. The Raspberry Pi series contains some rather powerful controllers compared to their credit card sized

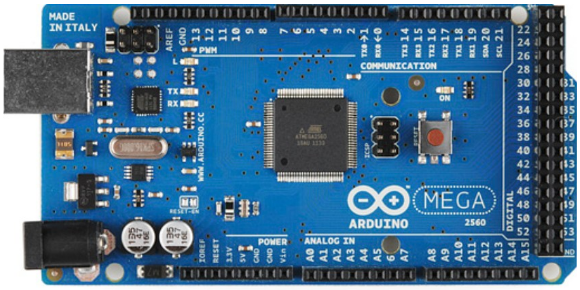

Figure 4.2: The Arduino Mega board[2].

Microcontroller	ATmega1280
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	128 KB of which 4 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Table 4.2: Specifications for the Arduino Mega.

boards, which makes them great as small processing units.

Something about those stupid pins....

Because of the power and memory of a Raspberry pi, a Linux OS is usually installed on a SD card and then plugged into the Pi for executing. A Raspberry Pi also contains a GPU and a video output, and finally a USB input. Because of this a Linux OS is easily supported.

In the following subsection a Raspberry Pi B+ will be described. The different models usually differs on cpu speed and memory, but a main difference from the early models is network connectivity.

4.2.1 Raspberry Pi B+

The Raspberry Pi B+ contains the specifications shown on 4.2.1 along with HDMI video output and Ethernet connectivity. As main storage a micro sd card must be installed, which also contains the OS.

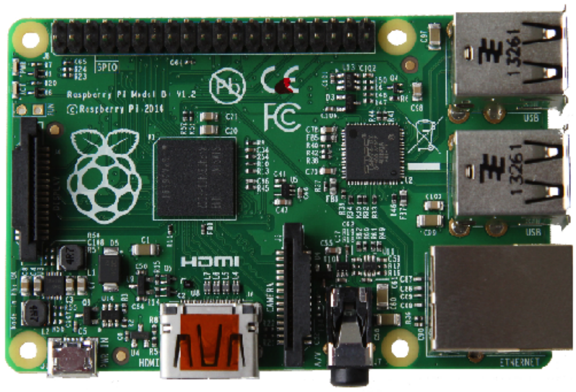

Figure 4.3: Raspberry Pi B+[**pibplus**].

Microcontroller	Broadcom BCM2835
RAM	512MB
Extended GPIO Pins	40
USB Ports	4
Clock Speed	700 MHz

Table 4.3: Specifications for Raspberry Pi B+

4.3 Sensors

To gather relevant data from the real world, and handling this data in a digital domain, there has to be some sort of sensor that is capable of reading this data.

4.3.1 Moisture Sensor

To Get information about the moisture of the mold, a moisture sensor is needed. A simple one, with digital and analog output is INSERTNAMEHERE.

find navnet -.-, og find documentation, er ebays beskrivelse gyldig?

beskriv hvad dens måleområde er,

4.4 Networks

There are several terms in the network domain, and this section will contain descriptions of networks and network theory, as well as definitions used in this report.

The implementation of a communication network is necessary, as the sensor system is supposed to reduce the amount of work for measuring the soil status over the golf course. The current method of measurement is time-consuming, because of the travel time used to cover the designated area of the golf course. A network implementation of the sensor system can transmit the data gathered from each sensor to an end point where the data can be analyzed or processed.

A computer network is a collection of computers and devices connected so that they can share information and services [3]. Devices and connections in computer networks can be modeled with graphs, and therefore the network terminology used will be similar to graph terminology. Further on in this report, a device connected in a network will be called a node and the connection from one node to another will be called an edge.

A topology is any arrangement of objects that can be connected with edges [4, p. 628]. A network topology is the arrangement of the nodes, using edges as the structure. There are different types of network topologies and here are some examples:

- Ring
- Line
- Bus
- Tree
- Star
- Mesh

A selection can be seen in figure 4.4. The bus topology will not be considered, because it is not suited for a wireless network implementation [**re-find-it**].

The star topology, figure 4.4 a, has one main node that the other nodes are directly connected to. An example of a star topology network is wifi, typically with a wireless router to which other devices connect to gain network access. The wireless router will handle the network communication and redirect the information to the correct device. A limitation of the star network is that all nodes must have a direct connection to the main node, and therefore be clustered within the reach of the main node coverage. kilde

A tree topology, figure 4.4 b, also utilizes a main node, called the root of the tree, but the devices in the network do not necessarily connect directly to the main node, but rather connect to another node that relays to the main node. This can

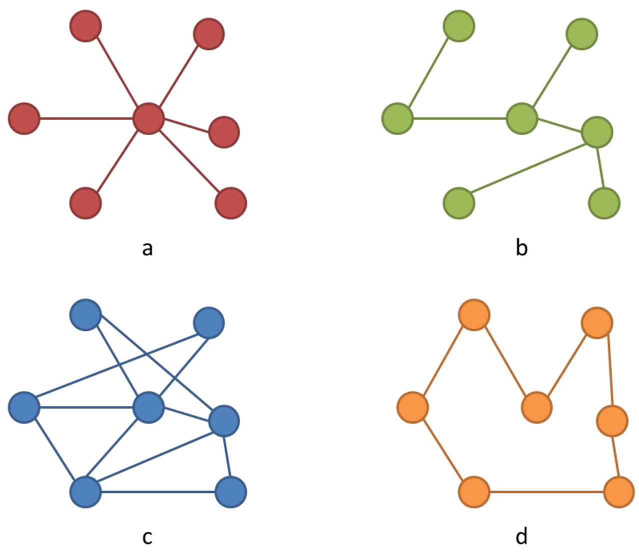

Figure 4.4: Network topologies.

repeat over multiple levels, so that information is relayed through several nodes, before reaching the main node. The tree network has a fixed node structure, and the relay nodes will route the information towards the destination. It has a reach kilde advantage over the star topology, as data of nodes not directly within reach of the main node still can be received at the destination node.

The ring topology, figure 4.4 d, connects all nodes in a Hamilton circuit, where the path passes through all nodes exactly once. This topology does not require a central node to control the data communication, and will like the tree topology relay the data being sent through a certain chain of nodes before it reaches the destination node. It is vulnerable, while there is only two directions to communicate, and if a connection is lost, there could be a stop in the data flow. If any constipation? one connection or edge is removed from the circuit, the remaining graph is a line, resulting in one chain of communication.

Another topology is the mesh, figure 4.4 c. There are two kinds of mesh networks: The full-mesh and the partial-mesh networks. A full-mesh describes a network where all the nodes are interconnected, similar to a fully connected graph. In this topology, there will be no redirecting or relaying, but just a direct connection to the destination node, regardless which destination. A partial mesh is also a mesh network, but does not require all nodes to be connected, so that it's similar to a tree topology, but with cycles. The partial-mesh must then support relaying of data, to transfer data from any node to the destination node.

There are multiple methods of communicating through a mesh network, therein Move this paragraph to protocols? routing and flooding are two alternatives. Routing will transfer the information to the destination node through a determined route, whereas the flooding method will notify all nodes within reach to distribute the information forward, and this will repeat until all nodes has transmitted the information, and hence the destination node also has received the information.

4.5 Communication devices

In the following section, different types of communication devices usable in the project is examined. The goal is to determine which communication type and device is best suited for the solution.

4.5.1 Bluetooth

Bluetooth is a wireless technology standard designed to transfer data over short distances. Bluetooth is often used in phones and computers, and can be used for connecting devices as the ones to be designed as the solution.

fix den her linje, er clue-less.

The number of devices that can be on a Bluetooth network is almost unlimited, and the the built-in interference reduction makes the technology usable for the product[5]. Bluetooth devices have to be paired in order to exchange data, which makes it hard to create a hot pluggable network of devices. This, and the short range of (at most) 100m[5]. makes it less viable for this project, as golf courses are large and these devices will have to cover great distances.

4.5.2 XBee

XBee is a radio module by Digi. Multiple versions of the XBee modules exists, with differences in power usage, range, built-in/external antenna, and some with processors[6]. XBees often use the ZigBee protocol. The ZigBee protocol is made for low-power devices, such as the product described in this report[7].

XBees are fast and predictable, making them a good choice for a project such as this. They are easy to work with on the Arduino platform, using the Serial pins. There even exists XBee shields made for Arduino.

XBees unfortunately are also quite expensive, which makes them unfit for this project. In later iterations of the product/unsurethey could be used if longer ranges became a requirement or it had to interface with other devices using the ZigBee protocol or XBee modules.

vi bør diskuterre om det er produkt eller løsning?

4.5.3 RF24 Transceiver

The RF24 transceiver module is a radio module used for exchanging data between modules of the same type. As with most of the XBee modules, the RF24 uses the 2.4GHz band, which is license-free in the whole world. It is well documented, and multiple libraries exists for using the device with Arduino.

The RF24 is cheaper than the XBees, although it has shorter range and, depending on the chosen protocol, is potentially slower.

The device chosen for this project is the RF24 because of the price. The shorter range is not a problem, as the project is more of a proof-of-concept. The speed is also good enough for the requirements, which makes this module good for the product described in the report.

The full RF24 specifications can be seen in appendix add this.

4.6 Communication protocols

A mesh network can use a wide variety of protocols, to manage the route data is transferred. In networking, a protocol is a special set of rules and standards for how nodes would interacts with each other. A well known protocols could be TCP/IP(Transmission Control Protocol/Internet Protocol), which today are used to communicate between almost anything with a internet connection. The mesh network we are looking at is a radio based network, and therefore some more relevant protocols will be examined. A few excising protocols will be presented in this section.

4.6.1 Time division multiple access

Time division multiple access(TDMA) is protocol that divides a single channel into smaller time slots. Each time slot transmits one byte or a segment of a signal, in a sequential serial data format. Each slot is active of a small amount of time, before the next slot in the queue get time to transmit[8].

TDMA is as an example used in the T1 telecommunication transmission system. Each T1 channels carry up to 24 voice telephone connections. Where each connection covers 300 Hz to 3000Hz and is digitized at an 8-kHz rate, which is two times the highest frequency component needed to retain all the analog content.

On Figure 4.5 it is seen how the channel, for T1, is split up into 24 smaller pieces. Each time slot is accountant for a user using a voice channel, to talk to some other user. Each time slot is of the size 8-bit, the user is unaware of this small data size, and that other are using the same channel, because the shift between

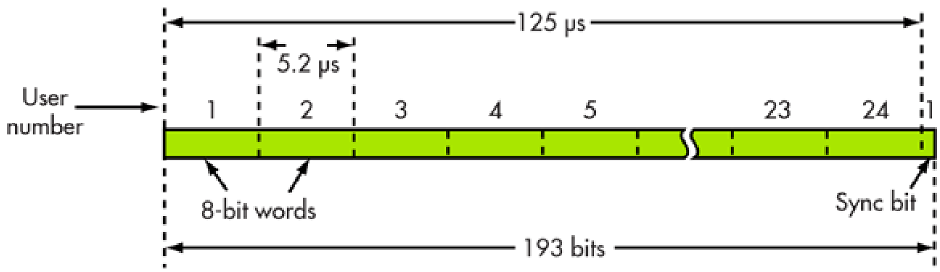

Figure 4.5: Illustration of the TDMA protocol

each time slot happens so fast. This gives the illusion of each user talking with no interruptions, even though each user only is assigned 1/24 of the total bandwidth. A single bit is used to synchronization. The TDMA system can maximum achieved a data rate of 1.544 Mbit/second[8].

TDMA can be used for any system that require several device to use the same channel, without interfering with each other.

4.6.2 Dynamic Source Routing protocol

The Dynamic Source Routing protocol (DSR) is a simple routing protocol designed specifically for use in multi-hop wireless ad hoc networks. With DSR the network is completely self-organizing and self-configuring, requiring no administration or existing network structure. The nodes in the network work together to forward packets to desternations that can be outside of the senders transmission range. As nodes can be added and removed ad hoc, the protocol automatically determine and maintain the routing of the packets though out the network. Since the number or sequence of nodes needed to reach any destination may change at any time, the resulting network topology may be quite rapidly changing. DSR is created to create a very low overhead yet being able to react quickly to change in the network environment. The protocol provides a highly reactive service to ensure successful delivery of data packets where node may be moving around, or other changes in the network occur[9].

The DSR protocol is composed of two main mechanisms that allow the discovery and maintenance of the routes in the network.

- Route Discovery is the mechanism by which a node S wishing to send a packet to a destination node D obtains a source route to D. Route Discovery is used only when S attempts to send a packet to D and does not already know a route to D.

- Route Maintenance is the mechanism by which node S is able to detect, while using a source route to D, if the network topology has changed such that it can no longer use its route to D because a link along the route no longer works. When

Route Maintenance indicates a source route is broken, S can attempt to use any other route it happens to know to D, or it can invoke Route Discovery again to find a new route for subsequent packets to D. Route Maintenance for this route is used only when S is actually sending packets to D.

DSR is a source routing protocol where packets carries a header, a ordered list of nodes to determine the route. This explicit use of routing allows the sender to select and control the route for the packets it sends. Source routing allow for load balancing, since the sender can create different routes out though the network, avoiding "heavy traffic". It is also guarantee that the routes used are loop-free, since a generated route never use the same node twice. By including this source route in the header of each packet, other nodes forwarding or overhearing any of these packets can easily cache this information for future use[9].

4.6.3 Ad hoc On-Demand Distance Vector Routing

Ad hoc On Demand Distance Vector(AODV) routing algorithm is a routing protocol designed for ad hoc networks. It is an on demand algorithm, meaning that it builds routes between nodes only as desired by source nodes[10].

AODV is a network where only the local nodes around a newly added node, is affected by the addition. If a link between nodes is broken, and it does not affect an ongoing transmission, no global notification occurs. This leads to a network where new nodes easily can be added and removed without overhead on the entire network. The transmission route in AODV is managed so, that only nodes in the direct route is active, which reduce the need for route maintenance and reduce idle nodes. The protocol can determine multiple routes between a source and a destination, but only a single one is implemented. The route is not necessarily the shortest[10].

A downside to AODV is that if a single route breaks, due to a defect node, it is not possible to know whether other routes exists. But if a route breaks, the protocol discovers a new route, if possible, as and when necessary.

When a node is ordered to send a packet to a specified destination, it checks its routing table to determine if it has a current route to the destination. If there already exist a node, the packet will be delivered to the next node in the route, repeating until it arrives at the correct location. If there does not exist a route, the node will initiates a route discovery process[10].

A route discovery process begins with the source creating a Route Request (RREQ) packet. The packet contains the sources node's unique ID and IP, and the destination node's unique ID and IP. The packet is then transmitted out from the source to it's neighbours. The unique ID of each node is then stored in the packet, to ensure the same node is not transmitting more than once. The destination ID is to ensure it knows when the destination node is reached. A simple

AODV network can be seen on Figure 4.6, where S and D represent the source and destination. It is visualized how the route discovery is invoked using the RREQ packet.[11]

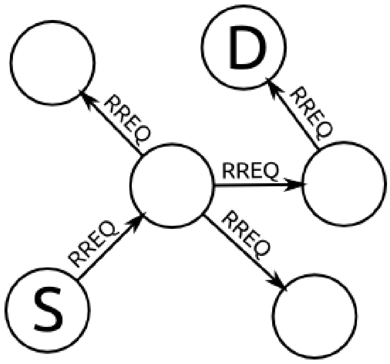

Figure 4.6: Illustration of the AODV route discovery

4.6.4 Better approach to mobile ad-hoc networking

Better approach to mobile ad-hoc networking (B.A.T.M.A.N) is a routing protocol for multi-hop ad-hoc mesh networks. B.A.T.M.A.N is said to solve some of the more typical problems with the classical routing protocols, which is not well suited for wireless ad-hoc networks. Some of the problems is, such networks are unstructured, are based on an inherently unreliable medium and dynamically change their topology.

When using the B.A.T.M.A.N algorithm, the approach is to divide the knowledge about the most reasonable path between nodes in a network to all of the participating nodes. Each node in the network only account for the best path to all other nodes in the system. This results in, the need for a global knowledge about local topology changes becomes unnecessary. In addition, does B.A.T.M.A.N have an event-based, but timeless, flooding mechanism that prevents the occurrence of loops and limits the amount of topology messages flooding the network. The protocol is designed to handle network where links between nodes are not always reliable[12].

4.6.5 Protocol comparison

Below is a comparison table with the protocol discuses in this chapter. The four different protocols is compared in what rute metric they are using, if they are loop free, support of load balancing, how reliable they are, and the estimated throughput. This table is to later help choose which protocol we decide to implement.

Er usikker på hvorvidt der skal skrives om vores valg her

Table 4.4: Protocol comparison

	Ad hoc	Rute metrics	Loop Free	Load balancing	Reliability	Thoughput
TDMA	Yes	Routes ensuring guaranteed bandwidth	Yes	Yes	High	Decreases as more nodes are added
DSR	Yes	Source routing	Yes	No	High	Decreases as more nodes are added
AOVD	Yes	Fastest & shortest path	Yes	No	High	Poor for more than 20 nodes
B.A.T.M.A.N	Yes	Fastest & shortest path	Yes	No	High	Good - scales well with more nodes

[13] [14] [9] [12]

Part II

Implementation

6. Design

7. Implementation

8.1 RF24 transreciever analysis

To confirm the usability of the RF24 transreciever, labeled as RF24 from this point, a short analysis with tests will be performed.

8.1.1 Range

A basic test to test the range is useful to ensure that RF24 is usable in a practical application with a wide distance between units.

Testing is done by having two units set as transmitter and reciever units respectively, having the transmitter transmit a single signal repeatedly while increasing the distance between the units. When the reciever no longer recieves the signal, the distance can then be measured.

CODE GOES HERE!

Transmitter / Reciever	Unit 1	Unit 2	Unit 3
Unit 1	-	?	?
Unit 2	?	-	?
Unit 3	?	?	-

8.1.2 Bit error rate:

what is?

how? code, setup etc.

Datasheet	Unit 1	Unit 2	Unit 3
0,0%	?	?	?

Part III

Conclusion

9. Reflection

oh..

9.1 What have we done!?

10. Summary

ok..

10.1 It ended like this

11. Future Work

Here's what's missing..

11.1 To be done

Bibliography

[1] Arduino. *Arduino - Introduction*. Seen 16/09/2015. URL: http://arduino.cc/en/Guide/Introduction.

[2] Seen 17/09/2015. URL: https://www.robotics.org.za/image/data/Arduino/Arduino%20Boards/arduino_mega_r3_002_hd.jpg.

[3] K. Mansfield and J. Antonakos. *Computer Networking for LANS to WANS: Hardware, Software and Security*. Networking (Course Technology, Inc.) Cengage Learning, 2009. ISBN: 9781423903161. URL: https://books.google.no/books?id=VQvhAN9iBuMC.

[4] K. Rosen. *Discrete Mathematics and Its Applications 7th edition:* McGraw-Hill Education, 2012. ISBN: 9780073383095. URL: https://books.google.no/books?id=XcrKi-kzwLIC.

[5] Inc. Bluetooth SIG. *A Look at the Basics of Bluetooth Technology*. Seen 24/09/2015. URL: http://www.bluetooth.com/Pages/Basics.aspx.

[6] SparkFun. *XBee Buying Guide*. Seen 24/09/2015. URL: https://www.sparkfun.com/pages/xbee_guide.

[7] Digi. *ZigBee Wireless Standard*. Seen 24/09/2015. URL: http://www.digi.com/technology/rf-articles/wireless-zigbee.

[8] Lou Frenzel. *Fundamentals of Communications Access Technologies: FDMA, TDMA, CDMA, OFDMA, AND SDMA*. Seen 17/09/2015. URL: http://electronicdesign.com/communications/fundamentals-communications-access-technologies-fdma-tdma-cdma-ofdma-and-sdma.

[9] UIUC D. Maltz Microsoft Research D. Johnson Y. HU. *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*. February 2007. URL: https://www.ietf.org/rfc/rfc4728.txt.

[10] Samir Das Ian Chakeres Charles Perkins Elizabeth Belding-Royer. *AODV*. Seen 17/09/2015. URL: http://moment.cs.ucsb.edu/AODV/.

[11] Johns Hopkins Dr. Baruch Awerbuch Dr. Amitabh Mishra. *Ad hoc On Demand Distance Vector (AODV) Routing Protocol*. Seen 17/09/2015. URL: http://www.cs.jhu.edu/~cs647/aodv.pdf.

[12] Jean-Philippe Lang. *B.A.T.M.A.N. protocol concept*. Seen 27/09/2015. URL: http://www.open-mesh.org/projects/open-mesh/wiki/BATMANConcept.

[13] Stephane Rousseau Vania Conan. *Route Selection for Capacity Maximization in Multi-Rate TDMA-based Wireless Ad Hoc Networks*. Seen 16/09/2015. URL: http://cnd.iit.cnr.it/rbruno/files/meshtech09.pdf.

[14] Piotr ZWIERZYKOWSKI Piotr OWCZAREK. *ROUTING PROTOCOLS IN WIRELESS MESH NETWORKS - A COMPARISON AND CLASSIFICATION*. 2009. URL: http://www.academia.edu/5756527/ROUTING_PROTOCOLS_IN_WIRELESS_MESH_NETWORKS_-A_COMPARISON_AND_CLASSIFICATION.

Part IV

Appendix

Hvilke ting holder i øje med, i henhold til vedligeholdelse af golfbanen?
Hvordan beplantningen groer, træer og græs.

Er det nogle arbejdsopgaver i kunne se automatiseret?
Set fra golf banens ledelse så ville automatiserede maskiner være gode da de så ville kunne spare greenkeeperen. Ville dog ikke være til vores fordel som greenkeepere

Med et middel der kunne gøre vi kunne se jordens tilstand med surhed og vandmængde, ville vi lettere kunne tilføre hvad der mangler af, gødning og vand

Ville en automatisering resultere i et besparelse eller anden fordel for jer?
Automatiske klippere, hvilket så er lavet

Andet interview nedenfor

Hvor vigtigt er det for jer at undersøge jordfugtigheden?
Vigtigt på greens, i forhold til om de tørrer ud. Sker ikke så ofte, men er relevant. Kan også være relevant på teested.

Hvordan gør i det nu?
Når hullerne flyttes undersøges dette.

Hvor ofte undersøger i det nu?
Mest om sommeren, da der kan udtørres. Planlægger det efter vejret, her kan produktet være til hjælp for at spare det arbejde.

Ville det være attraktivt med et system der automatisk undersøger det?
Ja det vil det. Der kan være forskel på forskellige steder på banen, da det kan regne nogle steder uden at det gør andre.

Kender i allerede til systemer til dette formål?
RainBird har et lignende system. Disse har også leveret styreprogram til vandingsanlæg til Aalborg Golfklub.

Vil det give mening at sende informationerne trådløst i stedet for gennem kabler?
Ja, så slipper vi for at grave dem over hver gang vi skal reparere noget på banen. Kan give problemer med ledninger i forvejen og vandingsanlæg. Især problemer med rødder på træer.

Hvor på banen skulle sensorer placeres? (Og hvordan? dybde? jord/sand?)
Mellem 10-15 cm. nede, da der skal være rødder på greenen. Der er ca. 30cm. sand på greenen, som vandet hurtigt ryger igennem. Rødder er ca. 10-15cm. nede, der hvor man gerne vil have det er fugtigt. En green kan være ret stor, så det kan være nødvendigt at placere flere på en green. En green kan være 500m^2.

Teesteder er ikke så vigtigt.

Nogle potentielle problemstillinger ved et sådant system?
Når jorden prikkes kan de rammes og evt. gå i stykker. Kan undgås ved at grave lidt ned, Kim mener at de prikker 10cm. ned. At finde dem efter de er gravet ned. Kan hurtigt blive et problem. Batteritid er vigtigt, kan blive et problem.

Nogle idéer til anvendelse eller forbedringer/udvidelser til systemet?
pH-meter indbygget vil være smart i forhold til at udregne brugen af gødning og kalk for at få den ønskede pH værdi (Græs gror bedst ved pH 5.6, varierer efter græssort, men under 7) Hvorvidt jorden er porøs eller ikke porøs, da dette skal bruges til at finde ud af om jorden skal prikkes. Jord har godt af at blive prikket, da for tæt jord holder på vandet og dette kan stoppe alt vækst og materiale.