Die approbierte Originalversion dieser Diplom-/Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Energy harvesting wireless sensor nodes in an AAL environment

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medieninformatik und Visual Computing

eingereicht von

BSc. Thomas Novotny
Matrikelnummer 0626340

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung: Ao.Univ.Prof.i.R. Dipl.-Ing. Dr.techn. Wolfgang Zagler
Mitwirkung: Projektass. Dipl.-Ing. Peter Mayer

Wien, 1. März 2017

Thomas Novotny	Wolfgang Zagler

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Energy harvesting wireless sensor nodes in an AAL environment

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Media Informatics and Visual Computing

by

BSc. Thomas Novotny
Registration Number 0626340

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ao.Univ.Prof.i.R. Dipl.-Ing. Dr.techn. Wolfgang Zagler
Assistance: Projektass. Dipl.-Ing. Peter Mayer

Vienna, 1st March, 2017

Thomas Novotny Wolfgang Zagler

Erklärung zur Verfassung der Arbeit

BSc. Thomas Novotny
Krebsengartengasse 3-5/1/9
1150 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbstständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. März 2017

Thomas Novotny

Danksagung

Computer, sowohl stationär als auch portabel, waren schon immer faszinierend für mich. Daher war es immer ein Fokus für mich, Computer zu verstehen und sie als Werkzeug zu verwenden. Als ich 12 Jahre alt war, hat mir einer meiner Brüder gezeigt, wie es möglich ist, auf einem Computer zu programmieren. Das war zu einer Zeit, als ein PC den Benutzer nicht mit einer BASIC-Prompt begrüßt hat, welche direkt zum programmieren eingeladen hat, sondern zu einer Zeit, in der die Zielgruppe für Computer schon die breite Masse war. Trotz dieser kleinen Hürde und dank des Internets und seinen endlos scheinenden Ressourcen über Programmieren habe ich mir im Laufe der Jahre Wissen darüber angeeignet, wie Computer funktionieren und wie ich sie programmiere.

Die glückliche Lage, einen Mastertitel verleiht zu bekommen wäre ohne die großartige Unterstützung meiner Eltern, die mir immer Freiraum gegeben haben, um meine Interessen zu verfolgen, wohl kaum zustande gekommen. Als jüngster von vier Brüdern und als dritter, der an dieser Universität ein Studium abschließt, hatte ich immer die Möglichkeit, meine Brüder als Beispiel dafür zu nehmen, wie meine Zukunft aussehen könnte. Weiters möchte ich meiner wundervollen Lebensgefährtin Henrieke danken, die meine Motivation und Inspiration für so ziemlich alles ist.

Spezifisch im Rahmen dieses Projekts möchte ich Dany Quiroga - den qualifiziertesten Ingenieur den ich kenne - danken, welcher sein Wissen und fabrizierte Prototypen für dieses Projekt zur Verfügung gestellt hat. Letztendlich danke ich auch dem gesamten Lehrpersonal auf der TU Wien, welches mir diese Ausbildung ermöglicht hat, insbesondere Peter Mayer, der dieses Projekt sowohl professionell als auch geduldig betreut hat.

Acknowledgements

Computers, stationary or portable, have always been fascinating to me. Therefore it was always a focus of mine to both understand computers as well as using them as tools of creation. I have to thank my brother Stephan, who introduced me to programming when I was about 12 years old. It was at a time where computers would not greet you with a Basic prompt upon turning and invite you to start programming, but at a time where computers started to be aimed at the casual home users. Despite that, thanks to access to the internet and its seemingly endless repository of information about programming, I acquired knowledge not only about how computers worked, but also about how to create programs.

Being in the fortunate situation of receiving a Masters degree in Computer Science is by no means something I could have done without the continuing support of my parents, who always gave me the freedom to pursue the fields that were most interesting to me. Being the youngest of four brothers and the third to get a degree at this University, I always had the opportunity to look at them for guidance of how my future could look like. Furthermore, I could not have done this without my wonderful partner Henrieke Goorhuis, who is a motivation and inspiration for everything I pursue.

Specific to this project I would like to express gratitude to Dany Quiroga, the most skilled engineer I know, for providing me with assistance and the manufacturing of prototypes. Lastly, my thanks go to all the teaching personnel at the Vienna University of Technology who enabled me to enjoy this education, especially Peter Mayer, who both, professionally and patiently, oversaw this project.

Kurzfassung

Nachdem der Altersdurchschnitt der Gesellschaft stetig zunimmt, gewinnt es immer mehr an Bedeutung, Menschen, die auf Hilfe angewiesen sind, die Möglichkeit zu geben, länger in ihrem gewohnten Umfeld zu leben. Durch eine Vielzahl von Active and Assisted Living (AAL) Technologien kann bis zu einem gewissen Grad ein unabhängigeres und sichereres Leben ermöglicht werden. Kabelgebundene Systeme sind für diesen Einsatz unpraktikabel, da sie einen hohen Installationsaufwand und zumeist permamente Veränderungen im Wohnraum mit sich bringen. Kabellose Systeme ermöglichen eine höhere Flexibilität, hängen allerdings immer noch von einer externen Stromquelle, welche Installationsaufwand verursacht, oder einer Batterie, welche regelmäßig gewechselt werden muss, ab. Die Firma EnOcean stellt einen Standard inklusive Hardware zur Verfügung, um kabellose *energy-harvesting* Module zu entwickeln, welche die nötige Energie aus dem Umgebungslicht oder anderen Quellen beziehen. Das macht es möglich, kabellose Sensormodule zu entwickeln, welche nahezu keinen Wartungsaufwand verursachen, sehr flexibel zu platzieren sind und keine zusätzliche Verkabelung notwendig machen. Ziel dieser Arbeit ist es, kabellose Sensormodule mit einem möglichst geringen Energieverbrauch zu entwickeln, welche die Möglichkeiten bieten, Präsenz, Stromfluß und Wasserfluß zu detektieren. Diese Module und die Erkenntnisse dieser Arbeit sollen als eine Grundlage für weitere Projekte im AAL-Bereich dienen.

Abstract

As society ages, having the option for persons to stay in their own home as long as possible becomes more important. This can be made possible through a variety of Active and Assisted Living (AAL) technologies that can assist with living a more independent life and ensure the safety of the person using it. Using cable-bound systems for this is neither practical nor economical as it requires making permanent changes to a home during installation. Wireless systems provide more flexibility, but still typically rely either on an external power supply or on batteries which have to be changed frequently. The company EnOcean provides a wireless standard and hardware for creating energy harvesting wireless sensor networks that rely on ambient light and other sources of energy that can be harvested from the environment, making it possible to create wireless nodes that are easy to install in a wide variety of locations without the need for additional wiring and maintenance. In this paper, the aim is to develop modules that provide an array of sensing options such as water flow detection, presence detection and non-invasive current detection while trying to keep the power consumption as low as possible, laying the groundwork for further projects that are based on wireless energy harvesting sensor nodes.

Contents

Kurzfassung **xi**

Abstract **xiii**

Contents **xv**

List of Figures **xv**

List of Tables **xv**

1 Introduction **1**

2 Background **5**
- 2.1 The case for AAL 5
- 2.2 Related Work 5
- 2.3 Available wireless technologies 6
- 2.4 Energy harvesting in wireless sensor nodes 7
- 2.5 EnOcean as basis for a sensor network 8
- 2.6 EnOcean Firmware development 14
- 2.7 Sensor options 17
- 2.8 Presence detection with infrared sensors 18
- 2.9 Non-invasive current detection 20
- 2.10 Water flow detection 21

3 Experimental Design and Prototypes **23**
- 3.1 Infrared sensor array module 25
- 3.2 Non-invasive current detection module 36
- 3.3 Water flow detection module 47

4 Results and Analysis **51**
- 4.1 Presence detection 51
- 4.2 Current detection 53
- 4.3 Water flow detection 55

5 Conclusion **57**

A Appendices **59**
A.1 Setting up Eclipse for EO3000I firmware development 59

Bibliography **65**

List of Figures

1.1 Population structure by major age groups, EU-28 2
1.2 Example setup showing different modules in a home setting 3
1.3 Flow of information for a suggested system using the developed nodes 4

2.1 An EnOcean STM330 module 8
2.2 Block diagram of an STM3xx module 9
2.3 Power usage during a measurement and transmit cycle of an STM330 module 11
2.4 The 5 fields of an ERP subtelegram 12
2.5 Pin outputs on the EVA330-1 development kit board 13
2.6 Pin descriptions of the 20-pin connector provided on STM3xx modules 14
2.7 Flowchart of the compilation of STM3xx firmware 15
2.8 Block diagram of the AMG88xx module 19
2.9 Graph of the different power states of an AMG88XX sensor 20
2.10 Data collected from a piezoelectric vibration sensor and a capacitor microphone attached to a metal water tap 22

3.1 Diagram of a full system setup, showing the USB300 transceiver and the 3 different sensor modules 24
3.2 Suggested minimal circuit for the AMG8831 25
3.3 Breakout board with AMG8831 module soldered on 26
3.4 Different mounting options for the grid infrared sensor 27
3.5 Test setup for the AMG8831 27
3.6 Heatmap used for visualization 27
3.7 Graph of the noise of 20 sensor pixels over the duration of 1000 seconds, pointed at a surface with consistent temperature, with one measurement every 30 seconds. 28
3.8 Initial test setup showing the location of the sensor, the field of view and the covered area. 28
3.9 Sensor readings 29

3.10 Graph of data gathered during the first test 30
3.11 Revised test setup with the sensor positioned higher and at a 60°angle 31
3.12 Graph of data gathered with the revised test setup 31
3.13 Graph showing: (a) time when the person leaves the bed; (b) time when the system stops detecting presence 32
3.14 Visualization of the sensor readings in 30-second intervals after the bed is unoccupied (from the same data set as Figure 3.13) 33
3.15 Prototype of the infrared sensor array module with 3D-printed case and a fixed 60° angle 34
3.16 Diagram of the infrared sensor array module prototype 34
3.17 Current detection circuit 36
3.18 The inductor is positioned orthogonally on the power cable. 37
3.19 Results of the first test with a 1000W load. (A_V=470, L=100μH) 38
3.20 Results of the second test with a 1000W load. (A_V=1424.24, L=100μH) . . . 39
3.21 Measurements with a load of 1000W: (1) A_V=470, L=100μH, (2) A_V=1424.24, L=100μH, (3) A_V=1424.24, L=10mH 39
3.22 Different loads with the corresponding sensor output on a logarithmic graph . 40
3.23 Output of the current detection circuit with the inductor placed in 45°steps around a 3-wire power cable with a 1000W load. 41
3.24 Output of the current detection circuit with the inductor placed in 5°steps around a 3-wire power cable with a 1000W load. 42
3.25 Output of the current detection circuit with the inductor placed in 5°steps around a 2-wire power cable with a 20W load. 43
3.26 Sensor output when connected to the power cable of an LCD TV. The TV was turned on and off two times. 44
3.27 Sensor output when connected to the power cable of a 700W rice cooker. Both cooking and warming mode can be clearly seen in the graph 45
3.28 Sensor output when connected to a washing machine doing a 30°C cycle including tumble-drying. Includes temperature sensor data 45
3.29 Current sensor prototype, attached to a 3-wire power cable 46
3.30 A temperature sensor connected to a flexible water tube 47
3.31 Temperature readings of the sensor connected to the flexible tube of a toilet water tank 48
3.32 Temperature readings on hot and cold water tubes of a bathroom sink 49
3.33 Prototype of the water flow detector module 50

4.1 Concept for an infrared presence detection module 53
4.2 Concept for a non-invasive current detection module 55
4.3 Concept for a water flow detection module 56

A.1 C Project wizard 60
A.2 Adding custom build configurations 61
A.3 Including C51 and EO3000I libraries 62
A.4 Adding keywords and typedefs 63

A.5 Creating a custom Error parser 64
A.6 Excluding uVision project files in the Project Explorer 64

List of Tables

2.1 Pin descriptions of the 20-pin connector of an STM3xx module 10
2.2 EO3000I power usage in different states 11
2.3 Sensor types and their possible uses in our specific use-case 18
2.4 Comparison of Infrared Grid Array modules 18

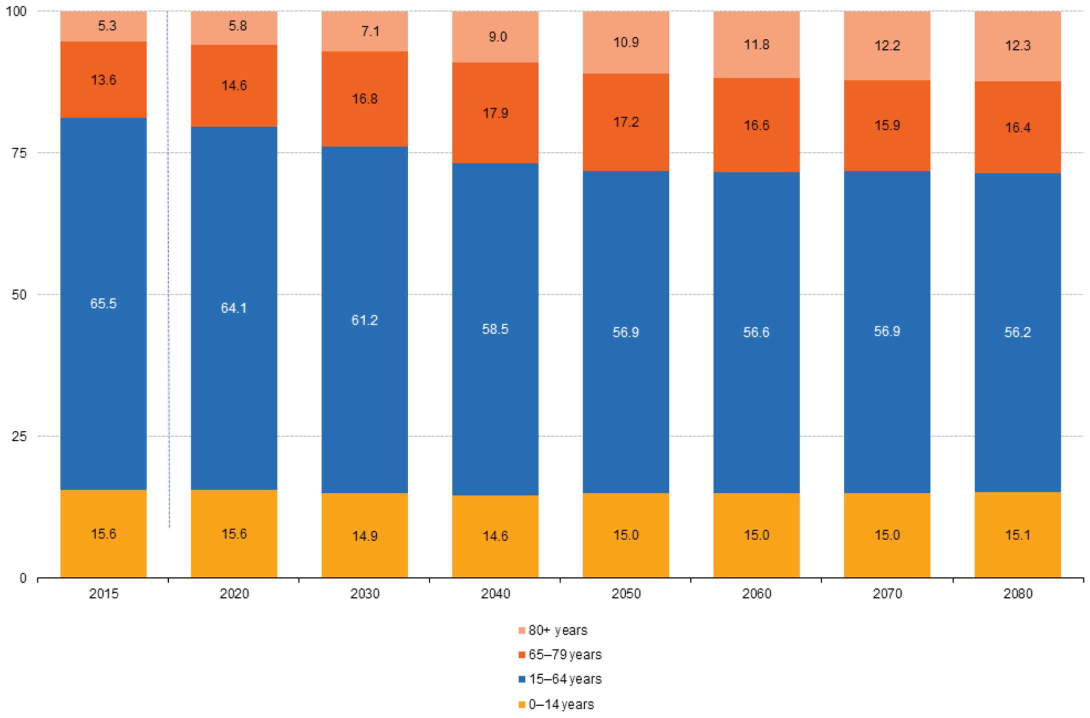
CHAPTER 1

Introduction

With the average age of citizens in the EU increasing significantly within the last decades [Eur], providing care and a safe environment for older people increasingly becomes a challenge. In all EU nations, the ratio of citizens over 65 versus citizens 64 and younger is increasing, creating an urgent need for solutions to provide proper care for the older population. Care homes are both expensive and often not desired by persons wishing to live mostly independently in their own homes. An alternative to care homes is support by Active and Assisted Living (AAL), which is an umbrella term for technology that aims to provide older persons and persons with disabilities with assistance in order to enable an independent life in their home. In recent years, Active and Assisted Living received attention from the European Union through the AAL Joint Programme [Ass16], which actively supports projects with the goal of finding practical solutions for these issues. Careful analyzing of data collected that give information about activities of daily living, collected at strategic points in a home can both improve the safety and lessen the need for additional care of the person in need of assistance.

In this thesis, wireless energy-harvesting sensor nodes will be introduced which are able to measure and detect various aspects of a persons' life which can provide information about their safety and well-being. This paper aims to provide a good resource for anyone who wishes to develop a passive AAL system that requires no user interactions. The following requirements are expected to be met:

- At no point the data collected should pose any significant invasion to privacy (such as recording of video would be)
- A system using these sensor modules should require no interaction with the user
- Modules should be easy to install and remove
- Modules should be non-obtrusive in their appearance



(¹) 2015: provisional; estimate. 2020–80: projections (EUROPOP2013).
Source: Eurostat (online data codes: demo_pjangroup and proj_13npms)

Figure 1.1: Population structure by major age groups, EU-28 [Eur]

- The need for periodic maintenance (e.g. battery replacement) should be kept at a minimum

The sensor nodes provide data about location and activity of the user within the home. These data can be used to make an educated decision whether assistance is necessary, in which case a person can be contacted and informed about a possible emergency.

To gather data, sensor modules will be developed to monitor the following aspects:

- Presence in bed
- Usage of toilet and water taps
- Usage of electric appliances, in particular ones that could pose a hazard

While dedicated systems for similar use-cases exist, downsides such as requiring additional cabling for power supply and data connection or maintenance effort for exchanging batteries in completely wireless systems lower the incentive to install such a system for only a limited amount of time. The proposed sensor nodes try to minimize installation effort and maintenance needs while also not being very noticeable and requiring no cables

for power or data transmission.
To achieve this, *EnOcean* wireless energy-harvesting modules form a base upon which sensor nodes are developed. Using these ultra-low-power modules imposes certain limitations onto the modules and the communication between the modules and the access point such as very little available energy and very small data throughput.

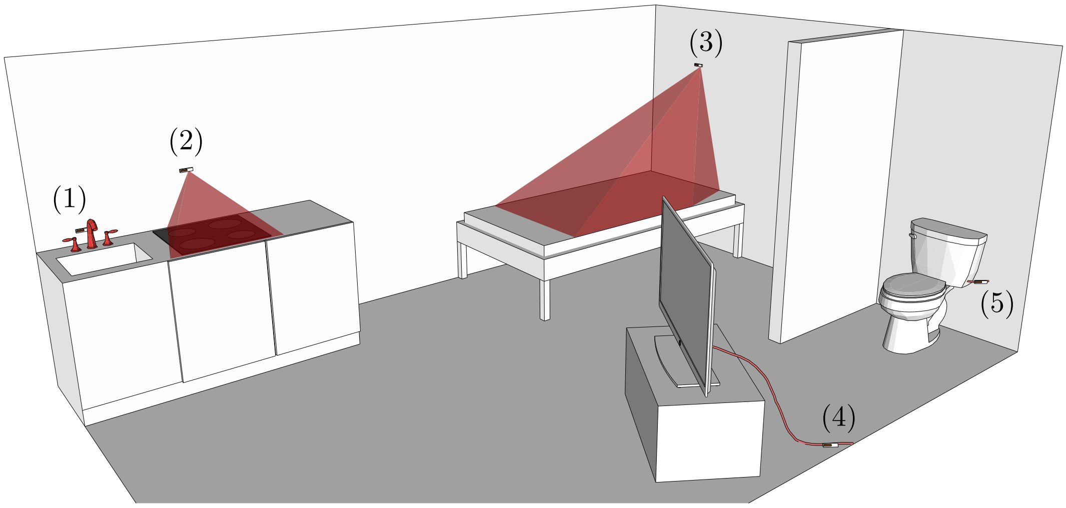


Figure 1.2: Example setup showing (1) water flow measurement at water tap, (2) infrared heat measurement of cooking plates, (3) infrared presence detection in bed, (4) current sensing of appliance, (5) water flow detection of toilet.

Figure 1.2 shows an example setup with all the modules which are introduced in this paper being used in the context of a small apartment.
While this thesis is primarily focused on developing sensor modules, a suggested system with flow of information from the sensors to a person who can give assistance is shown in figure 1.3.

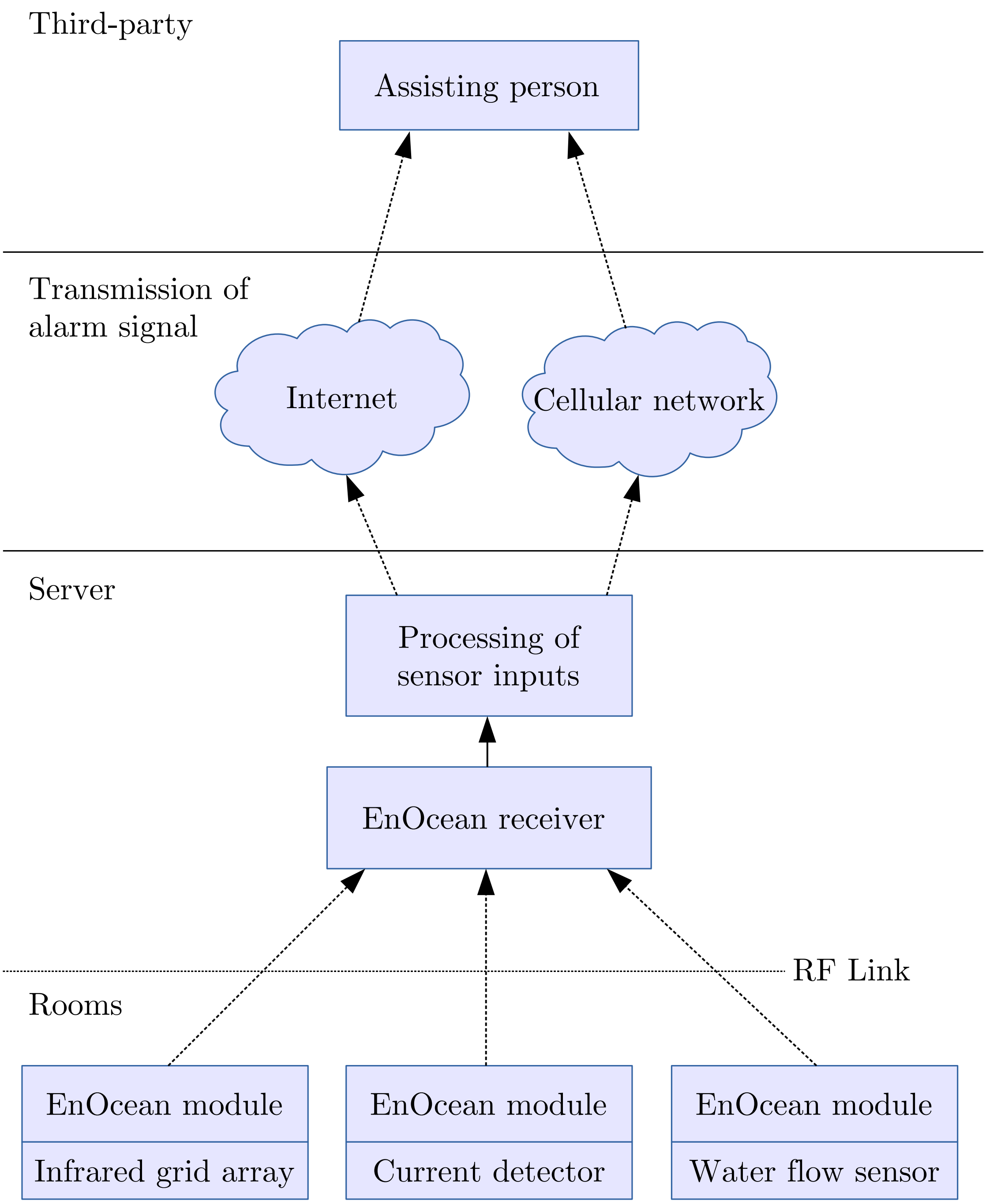


Figure 1.3: Flow of information for a suggested system using the developed nodes

CHAPTER 2

Background

2.1 The case for AAL

There are numerous arguments that support the use of technology to assist older and disabled people in their homes. First and foremost, the amount of people in need of assistance is steadily increasing, which also means that the amount of care workers will have to increase as well. While technology will — at least in the foreseeable future — never fully replace care workers, many tasks can be replaced or at least supported with technology, especially tasks that involve monitoring the safety and well-being of a person. At this time, Active and Assisted Living technologies are not yet widely employed and have not reached their true potential.

Giving people the option to stay in their own home for a longer period of time is desirable as it lessens the need for care homes and enables people to stay in their familiar environment. Another upside of reducing the need for care workers is the overall effect on the economy as it reduces overall care costs and enables people who are caring for their relatives to spend at least part of their time differently. With new cost-effective wireless technologies and sensors it is now possible to economically implement systems that monitor many different kinds of values that relate to the well-being of a person in their home. Considering nowadays almost everyone carries a computer with a multitude of sensors in their pocket, it would make little sense to not use similar technology to extend the time that people can stay in their own home.

2.2 Related Work

Due to AAL being a big research field and due to the fact that no AAL technology is widely adopted yet, there are many projects seeking to address the issues of extending the time that people can stay in their own homes.

Better health and ambient assisted living (AAL) from a global, regional and local economic *perspective* [EFH09] offers an overview and good arguments for AAL systems, raising the question why other places such as — for example — a car offer more assistance as technology progresses while homes generally stay the same.

A popular approach to ensuring safety of the system-user is wearable technology, which offers the advantage of not having to cover a whole living space with sensors. *Wearable wireless accelerometer with embedded fall-detection logic for multi-sensor ambient assisted living applications* [LFR$^+$09] introduces a body-worn device based on ZigBee modules and a MEMS accelerometer to detect falls. While devices like these are were shown to work well, the disadvantage is that requiring a user to wear a device at any time is invasive into the user's life. *Context-aware wireless sensor networks for assisted living and residential monitoring* introduces a more complex system of body-worn sensors which provide biometric data as well as location information to a back-end which analyzes data and recognizes unusual situations which might require action. A similar back-end could possibly be employed to analyze the data generated by the sensor modules introduced in this thesis.

A Survey on Ambient-Assisted Living Tools for Older Adults [RM13] evaluates different AAL systems and offers insight on the many different aspects that can be measured in a living environment. Different types of sensors (e.g. PIR, Camera, Pressure, etc.) are evaluated and examples for their use in AAL environments are given.

An Energy Efficient Middleware for an Ad-hoc AAL Wireless Sensor Network [FLPM$^+$13] introduces methods to reduce data traffic by processing output data of a PIR sensor locally before transmitting it to a remote server. This is relevant especially for energy-harvesting systems where both size and frequency of transmissions should be kept low.

2.3 Available wireless technologies

While this paper focuses on proprietary EnOcean wireless technology, it is relevant to look at other wireless technologies that are being used in AAL and home automation contexts. These wireless technologies fall under the term *WPAN*[1], which describes networks that typically have a range of 10-100m.[2] A selection of relevant wireless technologies are low-power Wi-Fi, Bluetooth LE, ZigBee and 6LowPAN [MSM16]. Wireless technologies designed for WPANs typically exhibit the following characteristics:

- Low overhead in data transmission
- Sleep and standby modes
- Low output power

While Wi-Fi is generally not optimized for ultra-low power applications, it is possible to reduce the power consumption to a point where it becomes a viable option for devices which run on battery power for an extended amount of time. Fully integrated solutions

---

[1] Wireless personal area network

[2] Through mesh networking, the total span of the network can be substantially bigger

such as the Espressif ESP8266 [Esp15] offer options to achieve this, but require heavy optimization of the firmware to reduce power usage to an absolute minimum [Sch16].
ZigBee [Org08] and 6LoWPAN[3] are both protocols based on IEEE 802.15.4 Standard for Low-Rate Wireless Networks [Soc15]. Numerous modules such as the Digi XBee, which offers firmware to support either ZigBee or 6LowPan, are available on the market. ZigBee remains the most popular 802.15.4-based standard in the home automation field as of today. The concept behind 6LoWPAN is connecting devices directly to an IPv6 network and giving every module a unique address, making it a prime technology for IoT[4] applications [M^+07] [ZS09].
Bluetooth LE is an extension of the Bluetooth standard and operates on the 2.4 GHz radio frequency band. A state-of-the-art Bluetooth LE 4.0 system is Samsung Artik [Sam15], a family of modules which are aimed at devices where processing of collected data is done on-module before sending data out.
While all of these technologies can be adapted for energy-harvesting applications, they still vastly differ in terms of power consumption compared to EnOcean modules, due to the fact that EnOcean modules are designed with the main objective of keeping power usage to a minimum.

2.4 Energy harvesting in wireless sensor nodes

There are multiple ways to harvest energy from the environment for an ultra-low-power system such as an EnOcean STM3xx module. This section gives a brief overview of the various sources of energy that can be practically or theoretically used.
Ambient light is a common source for energy harvesting. This energy is usually harvested using solar cells. When using solar indoor environments, it is important to consider that only about 0.1% of the sunlight level is available [EnO11c]. Furthermore, in many cases the harvester will likely generate little or no output current for an extended amount of time every day, which means extra considerations need to be made to make sure that this lack of harvested energy does not pose a problem to the system.
Temperature difference can be used for energy harvesting using a Peltier element. Peltier elements rely on the Seebeck effect to generate a current proportional to the temperature difference between the hot and the cold junctions of the element [MM05]. Using these elements is often not practical however as a constant and reliable source of heat is required (such as a machine or a person). Peltier elements can be used with STM3xx modules using the ECT310 DC/DC converter [EnO13c, p. 4].
Kinetic energy offers an option for energy harvesting in specific cases. As an example, linear motion from a user pressing a switch can be converted to create the small amounts of energy that an STM3xx module needs to transfer a single telegram [EnO15].
Piezoelectric materials can be used as energy harvesters as they produce a voltage when mechanical stress is applied [CCTS08] [KKK11]. Other options for energy harvesting include RF harvesting [PML13] and Electrostatic energy harvesting [TRM09].

---

[3] IPv6 over Low power Wireless Personal Area Networks

[4] Internet of Things

2.5 EnOcean as basis for a sensor network

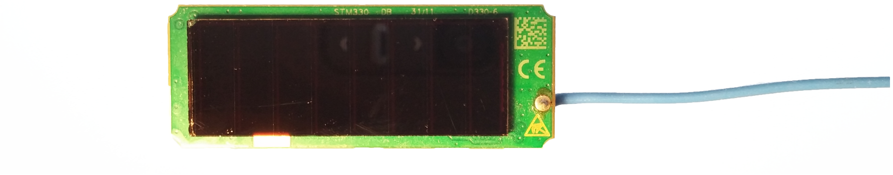


Figure 2.1: An EnOcean STM330 module

EnOcean provides a family of energy-harvesting wireless modules and has been chosen for this project as an underlying technology to develop sensor modules upon [EnO13c]. Specifically, STM3xx family modules are used as base for the wireless sensor nodes along with a USB-300 transceiver which houses a TCM310 transceiver module [EnO14c]. STM3xx modules are only able to transmit data periodically and are severely limited in doing so by the amount of ambient light, with the default time between 2 successive packet transmissions being 15 minutes [EnO13c]. The transmission range of a STM3xx module to a TCM300 (as used in the USB300 transceiver) is roughly 10-30 meters in indoor environments [EnO12b, p. 28]. Advantages of using EnOcean over other low-power wireless technologies such as ZigBee and Bluetooth LE include:

- The EnOcean family of products is designed from ground up for applications that use energy harvesting and is specifically targeted at indoor environments.
- The chance of collisions is extremely low due to rare and short transmissions
- Modules include everything that is essential for stand-alone operation, needing only a minimal external circuitry to connect periphery

Due to the ultra-low-power nature of the EnOcean module family, certain limitations apply to this project:

- Connecting additional sensors introduces an additional strain to the already very limited amount of power a STMxx module runs on, leading to energy conservation being the most critical problem for designing periphery.
- Communication between digital sensors and the EO3000I microcontroller are only possible with a logic level of 1.8V, requiring logic level conversion and introducing additional current consumption.
- Due to the very limited availability of energy, intervals between 2 successive transmissions are very long.

Specifically, an STM3xx module contains an 8051-based microcontroller with the following specifications [EnO14a]:

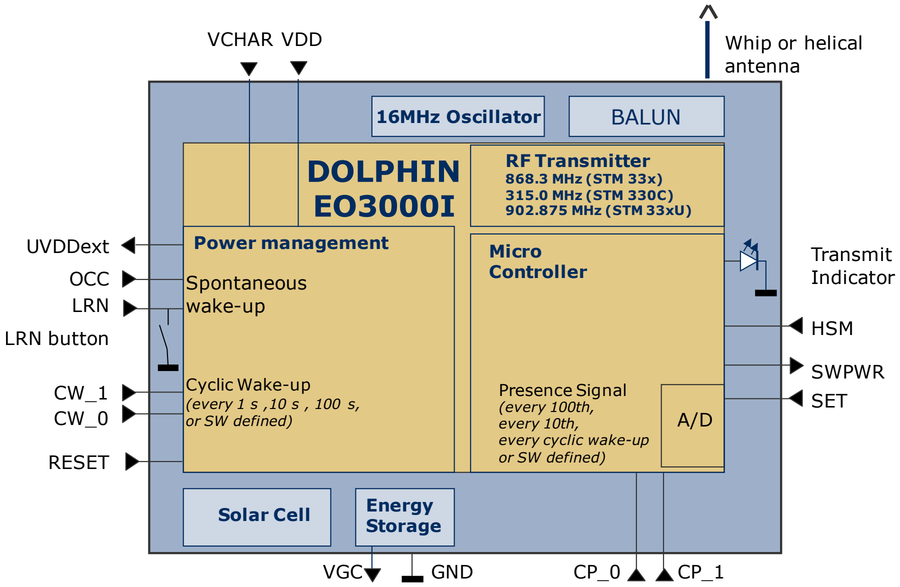


Figure 2.2: Block diagram of an STM3xx module [EnO14a]

- 32kb Flash, 2kB RAM
- 16MHz clock speed
- Integrated RF transceiver with a data rate of 125kbit/s
- Integrated ADCs and DACs
- 11 GPIO pins available through 20-pin connector
- UART and SPI

Apart from the EO3000I microcontroller an STM3xx module contains periphery that makes into a stand-alone system (see Figure 2.2):

- Energy source (typically a solar cell, other sources such Peltier elements possible)
- A 250mF Goldcap capacitor
- A charging circuit capable with the option of connecting an additional backup battery

An STM3xx module provides a 20-pin connector on the bottom side for connecting peripherals such as energy sources, energy storage or sensors. A full description of the available pins can be seen in table 2.1. At a minimum, an STM3xx module without additional circuitry uses 220nA when in deep sleep mode. Typical power usage of an EO3000I in different sleep states can be seen in table 2.2. The EO3000I can be woken by either a watchdog timer or external events. The configurable timer provides periodic

Table 2.1: Pin descriptions of the 20-pin connector of an STM3xx module [EnO14a] [EnO12b]

Pin	**Symbol**	**Function(s)**
1	UVDDExt	Ultra low power supply voltage regulator output, available while in deep sleep mode, only for use with WAKE pins. The current it limited by a 1.8mΩ resistor.
2	WAKE0	Wake input, must be connected to either UVDDExt or GND
3	WAKE1	Wake input, connected to UVDD by internal pull-up
4	VGC	Voltage Goldcap; It is possible to connect an external energy storage to this pin
5	GND	Ground
6	VCHAR	Charging voltage; It is possible to connect an external energy source to this pin
7	PROG_EN	Programming mode enable; connected to GND by internal 10kΩ pull-down resistor; active high
8	SWPWR	1.8V supply voltage for periphery, maximum output current 5mA; Only available when not in deep sleep mode
9	ADIO0	Digital IO / Schmitt Trigger input / PWM output / Analog input
10	ADIO3	Digital IO / Schmitt Trigger input / Analog input
11	ADIO1	Digital IO / Schmitt Trigger input / Analog input
12	ADIO4	Digital IO / PWM output / Analog input / Analog output
13	ADIO2	Digital IO / Schmitt Trigger input / Analog input
14	ADIO6	Digital IO / PWM output / Analog output / UART RX
15	RESET	Reset pin; active high
16	ADIO7	Digital IO / Analog output / UART TX
17	RSDADIO3	Digital IO / UART TX / SPI MOSI
18	SCLKDIO1	Digital IO / SPI CLK
19	WSDADIO2	Digital IO / UART RX / SPI MISO
20	SCSEDIO0	Digital IO / PWM output / SPI CS

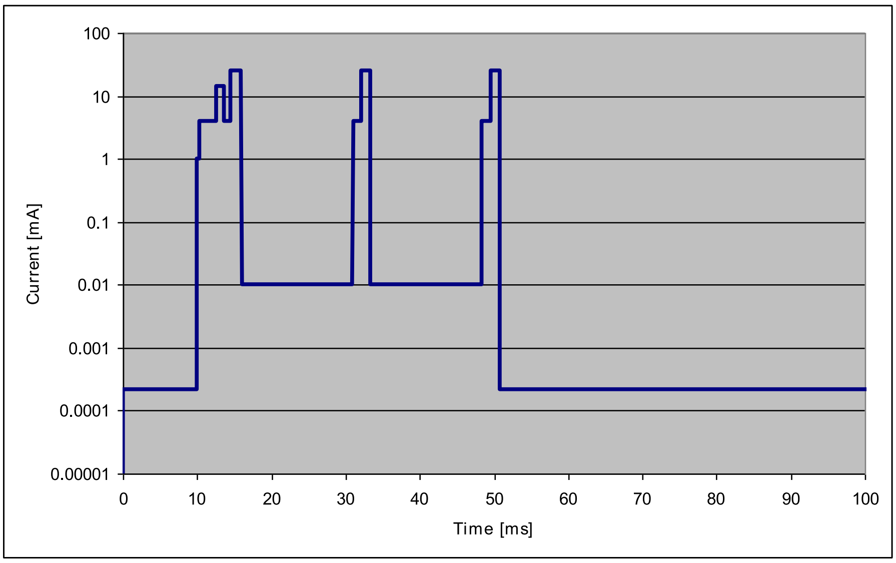


Figure 2.3: Power usage during a measurement and transmit cycle of an STM330 module [EnO13a, p. 22]

wake-up signals between 10ms and 46.6h. While many sensors provide the option to set thresholds and trigger pins of a connected microcontroller, this feature can not be used if the sensor is powered by the voltage regulator on the STM3xx since EnOcean modules don't provide power to external sensors while in deep sleep mode.

The STM3xx modules used for building the prototypes are powered by monocrystalline silicon solar cells [EnO13b] which charge the on-board goldcap capacitors. In addition to an on-board energy harvester, an STM3xx module provides the option to charge the on-board capacitor through the VCHAR pin. A battery can be connected to this pin in situations where energy harvesting is not sufficient or possible.

Table 2.2: EO3000I power usage in different states [EnO14a, p. 18]

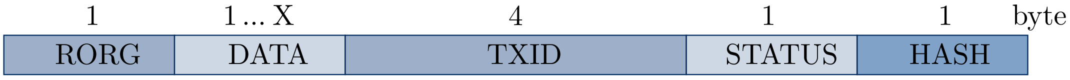
Symbol	**Mode**	**Conditions / Notes**	**Typ**	**Max**	**Units**
IDD_{OFF}	OFF	@ VDD=VOFF @ VDD = 1V @ 27°C	200 75		nA nA
IDD_{DS}	Deep Sleep	@ 27°C @ 85°C	220 2000	360 3100	nA nA
IDD_{FS}	Flywheel Sleep	@ 27°C @ 85°C	720 2300	1000 4000	nA nA
IDD_{SS}	Short Term Sleep	@ 27°C @ 85°C	8 25	10 35	μA μA
IDD_{SB}	Standby	Ultra low power blocks, voltage regulators and XTAL oscillator running	1.4	1.8	mA
IDD_{CPU}	CPU	Voltage regulators, XTAL, and-CPU 8051 at 16 MHz	23.4	30	mA
IDD_{TX}	Radio TX	@868 MHz and +6 dBm TX; power during transmission of high bits. CPU stopped	27.4	40	mA

2.5.1 The EnOcean Radio Protocol

EnOcean modules communicate with each other using the EnOcean Radio Protocol (ERP). This Section gives a brief overview of the aspects of this protocol that are relevant to this project. The lowest level of data in the ERP is a frame, which represents the encoded data, sent and received bit by bit on thy physical layer. Apart from the payload, a frame contains control and synchronization information. The actual payload of the frame which is encoded in the transmitter and decoded in the receiver is called a subtelegram. A subtelegram, shown in Figure 2.4 consists of 5 fields [EnO12a, p.5] [EnO14b, p. 17]:

- RORG - identifying the telegram type
- DATA - the subtelegram payload
- TXID - ID which identifies the transmitting module
- STATUS - information about whether the subtelegram is transmitted from a repeater
- HASH - checksum of all bytes in the telegram

It is worth noting that there is no size field in a subtelegram, instead the receiver counts up until the whole subtelegram is received and a valid HASH is transferred. To ensure redundancy and avoid data loss, a subtelegram is transmitted 3 times within a certain time period. These 3 transmissions form a telegram, which is the data unit of the network layer of the ERP. Relevant to this project is the small size of payload in telegrams, which is 4 bytes for a telegram with 4BS RORG value. One bit in the lowest payload data byte is reserved for the LRN flag, which specifies whether the telegram is a teach-in telegram. Teach-in telegrams are typically sent after pressing the LEARN button on an STM3xx module and are used for pairing the module with a receiver. The effective usable amount of data in a 4BS packet is therefore 3 bytes and 7 bit, which are used for 3 analog values and 7 digital values in the default STM330 firmware. For transmitting more data using 4BS telegrams, splitting the data into more than one telegram is an option. This should be avoided however, as any radio activity is a big drain on the power source.



1	1 ... X	4	1	1	byte
RORG	DATA	TXID	STATUS	HASH	

Figure 2.4: The 5 fields of an ERP subtelegram [EnO12a]

2.5.2 The EDK-300 development kit

For uploading firmware and configuration of EO3000I-based modules, EnOcean provides the EDK-300 development kit [EnO09] which consists of the EOP300 programmer and an adapter board for different modules. In this project, the EVA330-1 board is used, which provides a connector for one STM3xx module as well as access to all pins of the 20-pin connector of the STM3XX (see Figure 2.5), allowing external circuitry to be connected directly to the EVA300-1 board.

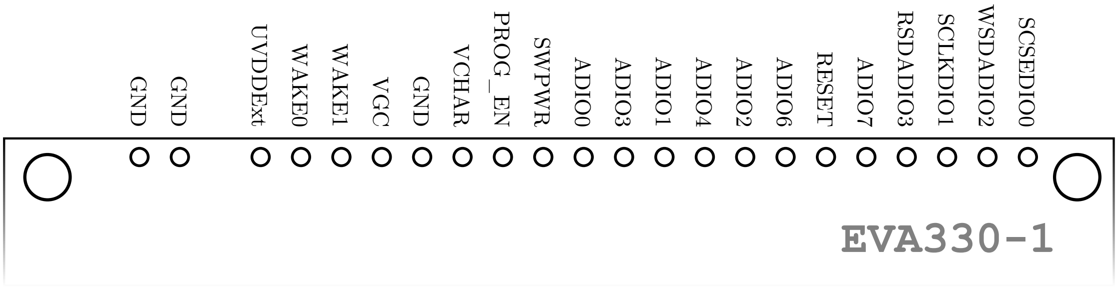


Figure 2.5: Pin outputs on the EVA330-1 development kit board

2.5.3 Interfacing an STM3xx module with external sensors

In addition to the digital IO, DACs and ADCs, the EO3000I offers hardware SPI and UART interfaces through a 2x10 pin connector with a pitch of 1mm. I^2C is not implemented in the hardware, however EnOcean GmbH released a reference software implementation [EnO11d], which will be used for connecting I^2C-capable sensors. The EO3000 runs on 1.8V, therefore any digital signals from sensors not compatible with a

1.8V logic level need to be level-shifted. In case of I^2C, a bidirectional level shifter is needed, such as the TXB0104 [Tex12]. While it is possible to supply a different operating voltage for the GPIO ports through the IOVDD pin[EnO14a, p. 15], this pin is not exposed onto the 20-pin connector of the STM3xx modules.
The full list of interfaces provided by the EO3000I GPIO0 and GPIO1[5]:

- 1 UART (2400-57600 baud)
- 1 SPI (max 2 MBit/s)
- 3 PWM ports
- 12 digital IO ports
- 4 analog outputs
- 5 analog inputs

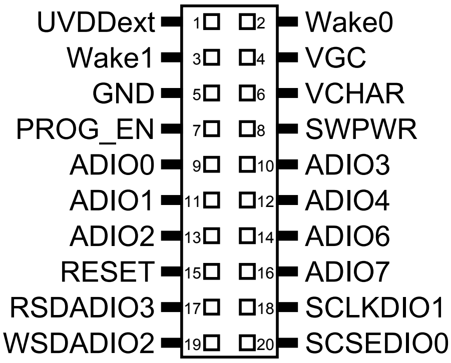


Figure 2.6: Pin descriptions of the 20-pin connector provided on STM3xx modules

External circuitry can be supplied with power through the SWPWR pin. This pin supplies 1.8V and a maximum of 5mA anytime the EO3000I is not in deep sleep mode. This means that any external circuitry will be completely shut down whenever the EO3000I is in deep sleep mode, unless an external power source is used.

---

[5] not all of the EO3000I GPIO pins are available on the 20-pin STM3xx connector

2.6 EnOcean Firmware development

Developing a firmware for EnOcean EO3000I-based modules requires the following software [EnO11a]:

- Keil C51 Compiler
- uVision IDE
- DolphinAPI
- DolphinView
- DolphinStudio

The *Keil C51 compiler* is needed to compile any firmware for the EO3000I as the source files provided by EnOcean depend on specific features and compiler directives. While it would be possible to use open source compilers such as the Small Device C Compiler (SDCC) [Dut15], doing so would require adaptation of all sources. The *uVision IDE* is not required, however it is recommended as it is the preferred IDE and builder for EnOcean firmware. The sources for the standard STM module firmwares are supplied by EnOcean as uVision projects.
DolphinView is a tool which connects to an EnOcean Transceiver running the EnOcean Gateway firmware (in this case the EnOcean USB300 [EnO14c] is used) and provides features for capturing and analyzing of telegrams. Using DolphinView, it is possible to capture and inspect the contents of any EnOcean telegrams which are sent by STM devices within the range of the transceiver.
DolphinStudio is a tool which aids firmware development by generating header files containing configuration data for the EO3000I. The following options can be configured in DolphinStudio:

- GPIO Pin functions
- Digital I/O pin directions
- External interrupts
- UART baud rate and buffer size
- SPI clock speed and modes
- Telegram buffers and maturity time
- Filter configuration
- Timer configuration

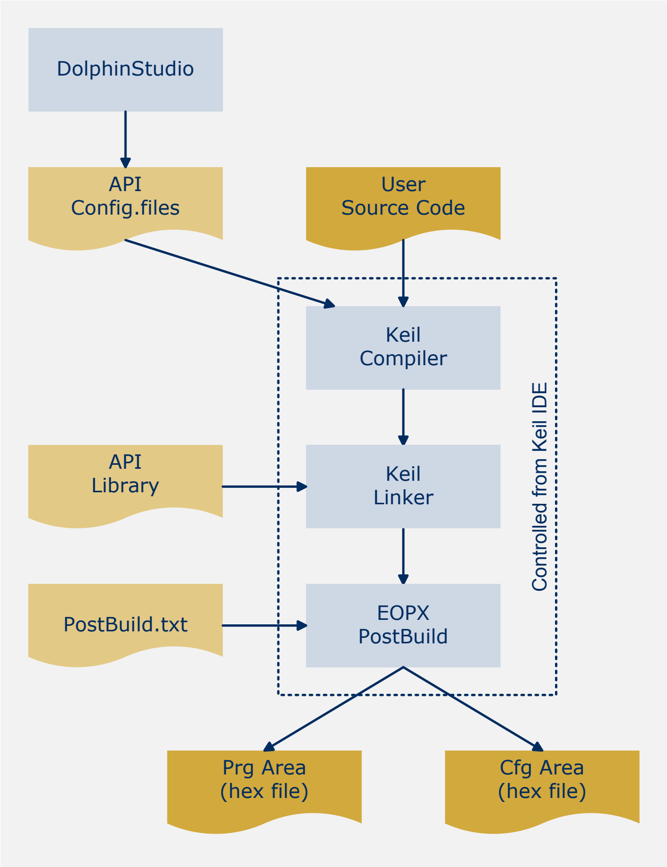


Figure 2.7: Flowchart of the compilation of STM3xx firmware [EnO11b]

Due to some features lacking in the uVision IDE, Eclipse was used as for this project instead, while still using the uVision builder. Instructions on how to use Eclipse for EnOcean firmware development are available in Appendix A.1.

Figure 2.7 shows a flowchart of the compilation of Firmware for an STM3xx module. It is recommended to base new firmware on the existing STM310 or STM330 firmwares which are supplied by EnOcean [EnO11a]. The standard STM310 Firmware (STMSEN) is composed of the following files:

- **EO3000I_CFG.c & EO3000I_CFG.h** - Files generated by DolphinStudio containing configuration data for the EO3000I registers
- **init.c** - contains routines for initializing the system after powering up and resuming from sleep modes
- **measurement.c** - contains routines for analog and digital measurement and readies data for being sent in a telegram

- **sending.c** - provides functions to send telegrams which transmit the measurement data
- **retransmission.c** - provides functions for redundant retransmissions, when configured
- **wakeup.c** - contains the ISR for wake-up as well as other functions relating to cyclic wake-up
- **main.c** - contains the main program which gathers measurement data, sends out a telegram and resumes deep sleep mode

In order to create a custom firmware that performs a measurement and sends telegrams, it is usually sufficient to create a new measurement routine to replace the *MeasureAnalog()* routine and change which values are included in the 4BS telegram. The methods *radio_ulpPrepareTelegram(TEL_RADIO_TYPE *pu8TxRadioTelegram, TEL_PARAM_TYPE *puTxParam)* and *radio_ulpSendTelegram()* are used to prepare and send a telegram.

2.6.1 Analog measurement

Analog measurement can be done using one or more of the 7 available ADCs on the STM3xx module.[6] Unlike other microcontrollers which contain an ADC that returns a value, where for example 0x00 describes 0V and 0xFF describes 1.8V, the ADCs in the EO3000I measure against an internal reference voltage and always have to be scaled using measurement values from the positive reference voltage (VDD) and the negative reference voltage (VSS) [EnO]. This is also done using the internal ADCs. The following listing provides an example of a typical measurement of the voltage connected to the ADIO1 pin:

```
sint16 s16negref;
sint16 s16posref;
sint16 s16adio1value;
uint16 u16adio1value;

// enable ADC
io_enableAnalog(1);
// measure positive reference (VDD) against internal voltage
io_ulpMeasAnalog(RVDD, &s16posref);
// measure negative reference (VSS) against internal voltage
io_ulpMeasAnalog(RVSS, &s16negref);
// measure ADIO_1 against internal voltage
io_ulpMeasAnalog(GPIO1_ADIO_1, &s16adio1value);
// disable ADC and restore radio functionality
```

---

[6] ADIO5 is not available on the 20-pin connector of the STM3xx module

```
io_enableAnalog(0);
// scale and calculate result
io_ulpScaleAnalog(s16posref, s16negref, s16adio1value,
    s16negref, 8, &u16adio1value); for ADIO_1
```

Listing 2.1: Example code for performing a measurement using the built-in ADC of the EO3000I

2.6.2 I^{2}C

As mentioned in earlier, EnOcean provides a reference implementation of I^{2}C for the EO3000I. This implementation was originally released as an example of how to connect an SHT21 humidity and temperature sensor to an EnOcean STM300 module, but can easily be adapted for connecting other devices. Using this implementation, either of the following pins can be used as the SDA and SCL lines [EnO11d, p. 5]:

- SCSEDIO0
- SCLKDIO1
- WSDADIO2
- RSDADIO3

2.7 Sensor options

With the requirements for collected data being presence detection, water flow detection and current detection, there is a wide array of sensors available that could on some way accomplish one or more of these tasks. Sensors could be used for different scenarios, e.g. a water flow sensor could be used to detect toilet usage as well as usage of a washing machine. Table 2.3 lists types of sensors and possible uses in an AAL environment.

Sensor type	**Possible application(s)**
Temperature sensor[7]	Measuring room temperature, measuring change of temperature on water pipes, measuring temperature on room heaters
Video camera	motion detection, presence detection
Reed switch	Door and window contacts
PIR Sensor	Motion detection, limited presence detection
Grid infrared array	Presence detection, detection of hot cooking plates
Inductive AC sensor	Detection of electric appliances being used
Light sensor	Measuring usage of lamps

Table 2.3: Sensor types and their possible uses in our specific use-case

---

[7] STM33x modules include a temperature sensor with a resolution of 0.16°C/LSB, however due to its location on the board it cannot be used as a contact temperature sensor.

2.8 Presence detection with infrared sensors

2.8.1 Available sensor modules

A recent addition to the available sensors for use in AAL environments are MEMS infrared sensor arrays, which allow contact-less temperature measurement of multiple areas at the same time. Sensors with a resolution of up to 8 x 8 pixels are available within a reasonable price range. All of the discussed sensors use MEMS[8] technology. A comparison of the sensors which have been considered for use can be seen in Table 2.4.

Table 2.4: Comparison of Infrared Grid Array modules

Sensor	**Melexis MLX90621**	**Omron D6T44**	**Panasonic AMG8831**
Pixel resolution	16 x 4	4 x 4	8 x 8
Temperature range	-20°C to 300°C	5°C to 50°C	0°C to 80°C
Operating voltage	2.6V	4.4 to 5.5V	3.3V
Data interface	I^2C	I^2C	I^2C
Current consumption	7mA	5mA	4.5mA
Field of view	60°x 15°, 40°x 10°, 120°x 30°	44.2°x 44.2°	60°x 60°

The Omron D6T44 was not chosen due to the high operating voltage and power consumption. The Omron D6T1616 was a promising alternative, operating on 3.3V and providing the highest pixel amount (16 x 16) of all the considered sensors. However, the D6T1616 was only produced in small quantities for developers and the manufacturer stopped development on this product, therefore it was of little use to try obtaining a sample for evaluation. The Melexis MLX90621 is a sensor mainly intended to be used in automobiles where its wide aspect radio of 1:4 allows tracking of 2 passengers [Mel12]. In the end, the Panasonic AMG8831 was chosen for offering a decent resolution and 3.3V operation. While the power usage of neither sensors fits the requirements of an ultra-low-power application, the AMG8831 offers the best trade-off in terms of resolution and power consumption.

2.8.2 Evaluation of the AMG8831

The AMG8831 is a 3.3V logic high-gain 8x8 pixel infrared grid array with a measuring range of 0°C-80°C per pixel [Pan12]. The pixel data is transferred over I^2C, with one pixel temperature being transferred in 12 bits (11 bits + sign), spread over 2 bytes. The temperature resolution of each pixel is 0.25°C/LSB [Pan11, p. 14]. The AMG8831

---

[8] Microelectromechanical systems

furthermore contains a thermistor for measuring the ambient temperature.

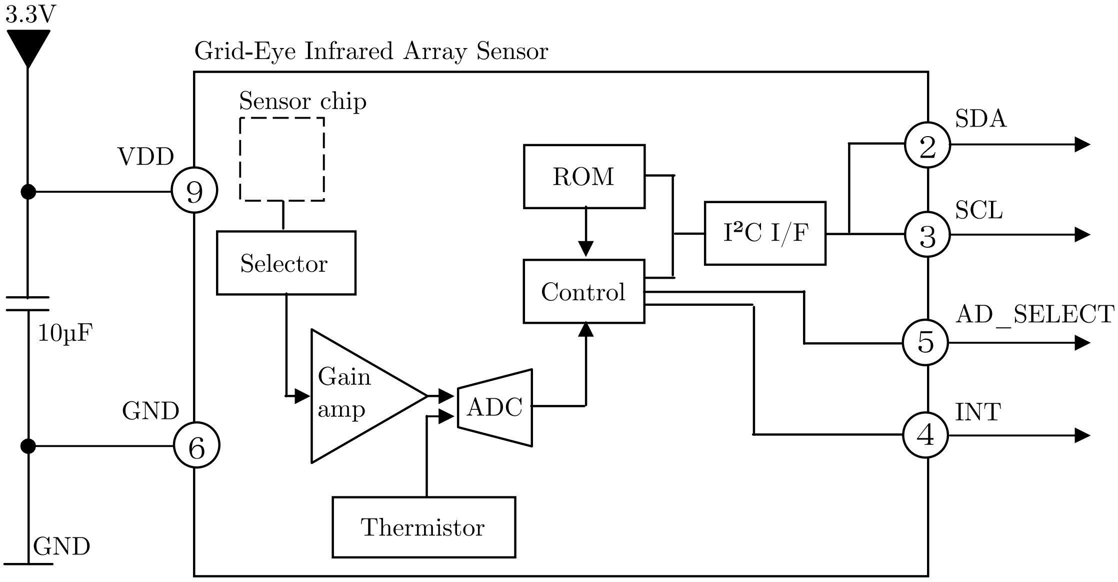


Figure 2.8: Block diagram of the AMG88xx module [Pan11, p. 1]

To retrieve a single pixel temperature, 2 registers have to be read at address X and X+1 (the pixel temperature registers start at 0x80 and end at 0xFF). Byte B_X and the lower 4 bit of byte B_{X+1} shifted to the left by 8 form a 12-bit value. The lower 13 bits are the temperature value and the MSB is the sign. The thermistor register works the same way and is located at address 0x0E to 0x0F. The resolution of the thermistor register is 0.0625°C/LSB.
The AMG88XX offers 4 different power modes:

- Normal operation, the default mode which the sensor is in after powering on. Measurements can only be made in this mode. (Typically 4.5mA consumption)
- Stand-by modes for short term sleep, either 10 seconds or 60 seconds. (Typically 0.8mA consumption)
- Sleep mode for lowest power consumption. (Typically 0.2mA consumption)

By sending specific commands, the sensor can be switched between the different modes. While it is possible to switch from stand-by into sleep mode, it is not possible to switch directly from sleep mode into stand-by mode. Figure 2.9 shows a graph of the different power states.
What has to be kept in mind when using the AMG8831 is the pixel noise, which has been around ±0.5°C in tests (See Section 3.1.4). While this is sufficient for presence detection in most cases, it might pose a problem in applications where precise temperature measurement is required.

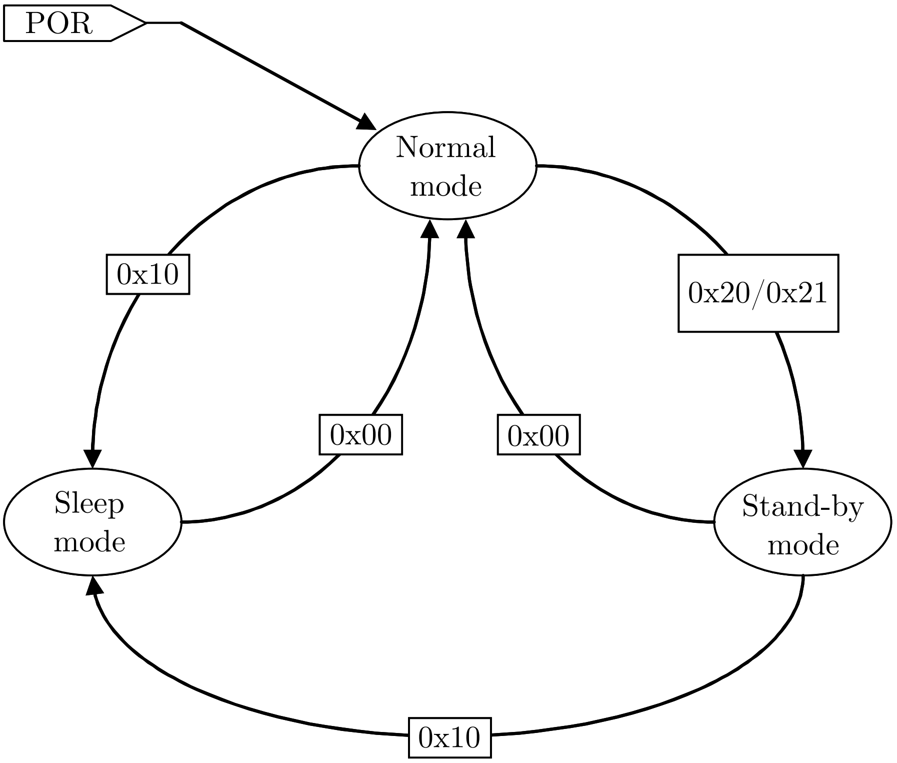


Figure 2.9: Graph of the different power states of an AMG88XX sensor [Pan11]

2.9 Non-invasive current detection

As part of a complete monitoring system, it is important to consider monitoring the usage of specific electric devices in a household, specifically devices that could pose a risk of harming the user if incorrectly used or malfunctioning. This includes mostly devices which consume a large current when turned on such as toasters, water boilers, cooking plates and blow-dryers. Being able to measure the time and duration these devices are turned on is useful for two reasons: If devices are not used for a long time, it can be an indicator that something is wrong. On the other hand, some devices — if turned on too long — could pose a hazard, for example a water cooker that is turned on for over 10 minutes.
The following criteria should be met by the current detection sensor node:

- Detect whether a device connected by a power line is turned on or off with reasonable accuracy
- Attachable to any power cable
- No required alterations to an existing setup

Typically, current measurement is done with a device that plugs in between the load and the power outlet. While providing the most accurate results, this method is not within the requirements of being non-invasive and also not usable for devices with fixed wiring. An advantage of invasive measuring however would be that the wireless sensor module itself could be powered from the same source, eliminating the need for energy harvesting. During the research for possible methods, capacitive voltage measurement was tested, which involved copper surface in close proximity to the wire, with signals being amplified by cascaded transistors. This method provides accurate results about the presence of 230V AC in a power cable while not being able to provide any useful output relating to the amount of current. While this method can only be used to detect voltage, it is worth mentioning as specific cases might call for detection of mains voltage instead of current detection.

A different method for measuring voltage with reasonable precision is measuring the electro-magnetic field around a single conductor. While sensors for this purpose are readily available, single conductors are not accessible and would require a specific power cable that exposed a single wire. An example where this method is used is the Extech DA30 [Ext02], a non-contact current detector. The sensor that is developed in this paper is roughly based on the circuit used in this device.

2.10 Water flow detection

When researching non-invasive water flow sensors, the obvious choice for accurately measuring water flow in a wide variety of pipes are ultrasonic water flow meters. Measuring water flow using ultrasonic signals works by placing an ultrasonic transmitter and an ultrasonic receiver on the same pipe with a known distance between them. When liquid flows through the pipe, this will either speed up or slow down the ultrasonic signal, making it possible to calculate the flow rate [rsh17]. The difficulty about using this technique is that these sensors are out of the reasonable price range for developing affordable sensor modules. Furthermore, ultrasonic water flow sensors likely won't fit the strict power requirements of an ultra-low-power energy harvesting system. While ultrasonic sensors offer a precise way to measure water flow, such accuracy is not strictly required. Simple information about whether water is flowing or not is sufficient in this context. Therefore, different options of detecting water flow have been explored.

Piezo-electric vibration sensors as well as *electret microphones* have been evaluated for their usability for detecting water flow by detecting vibration and noise when attached to pipes. In a simple test setup, both have been attached to a water tap with the output values being logged by the ADC of an ATMega328. The results of one measurement can be seen in Figure 2.10 and show only minor change. The idea has been discarded.

Another option for detecting water flow is measuring the change in surface temperature of the pipe. It can be assumed that in a home the cold water supply will be significantly colder than room temperature while the hot water supply will be significantly hotter. When no water is flowing, the pipes eventually approach room temperature. Based on these assumptions, a water flow detector will be developed in this project.

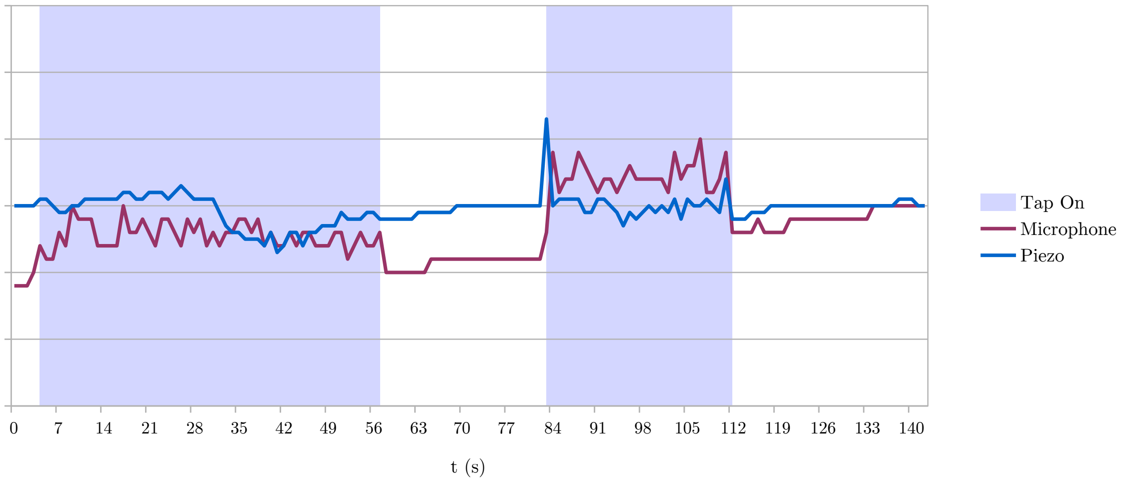


Figure 2.10: Data collected from a piezoelectric vibration sensor and a capacitor microphone attached to a metal water tap

CHAPTER 3

Experimental Design and Prototypes

Based on previous research, three sensor module prototypes were developed:

- A water flow detection module using I^2C temperature sensors
- An infrared sensor array module using the AMG8831 grid array sensor
- A non-invasive AC current detection module using an inductive current sensor

Figure 3.1 shows a diagram of a system with one of each modules, including the EnOcean USB300 gateway, which receives all telegrams sent by the modules.

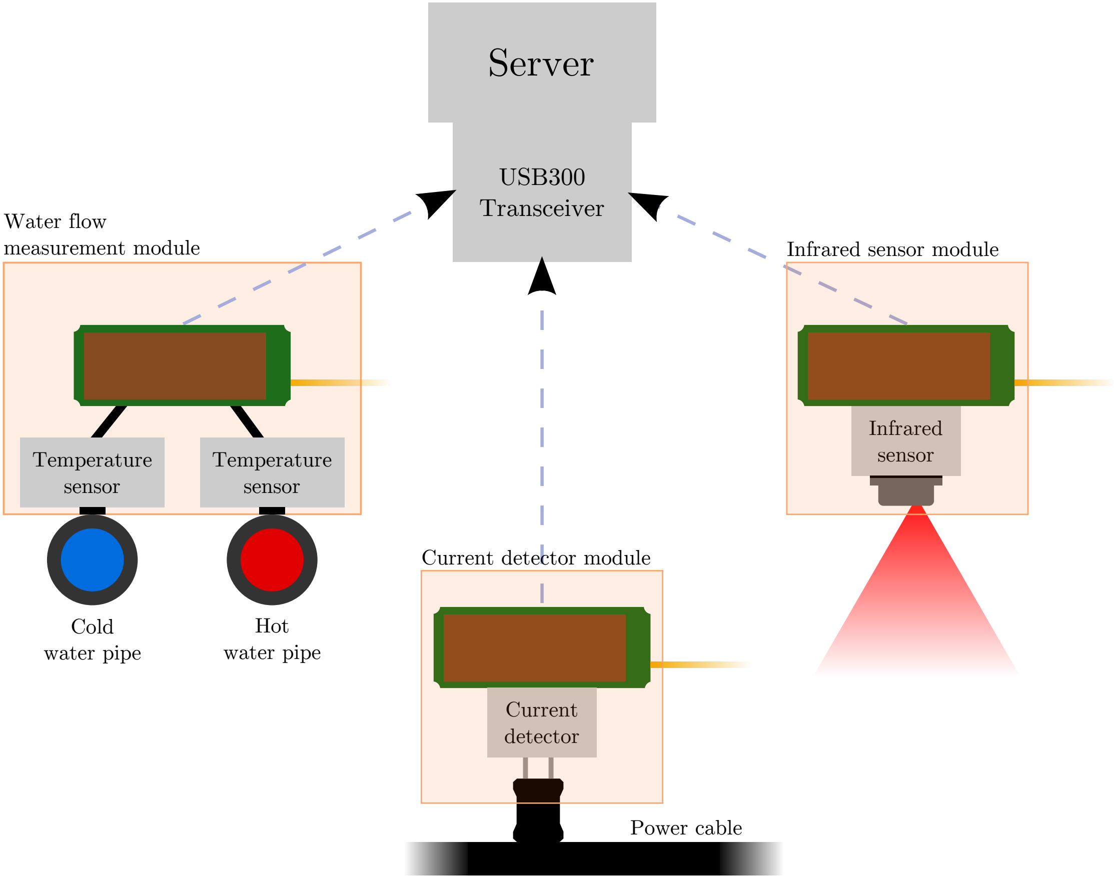


Figure 3.1: Diagram of a full system setup, showing the USB300 transceiver and the 3 different sensor modules

3.1 Infrared sensor array module

As discussed in section 2.8.2, the AMG8831 Grid-Eye Infrared Array sensor [Pan12] was selected to be used in a module which can perform presence detection and possibly other tasks (e.g. monitoring of cooking plates). Collected data will be locally processed and transmitted at regular intervals by this module.[1] The AMG8831 infrared sensor array module has the following specifications:

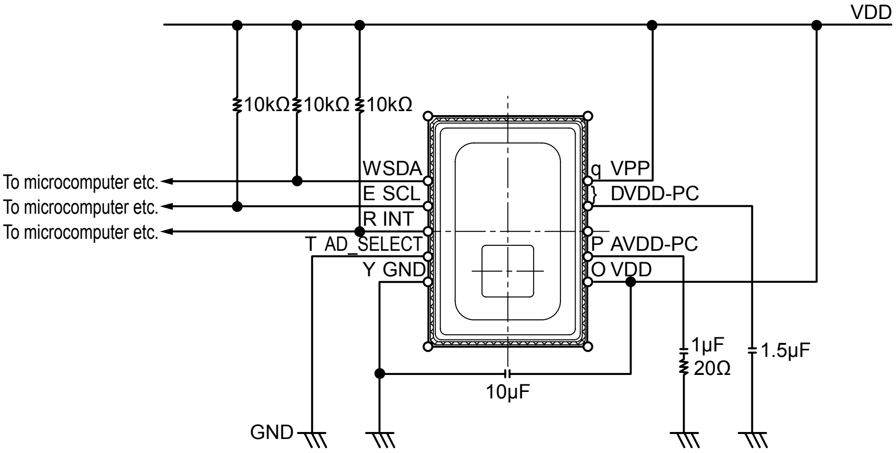
- 8x8 measurement field
- 60° coverage
- 12 bit temperature resolution
- 3.3V operating voltage

Tests will be performed with the sensor module mounted over a bed with the goal of detecting presence in the bed with reasonable accuracy.

3.1.1 Breakout board

Since the AMG8831 comes in a package similar to DFN[2] packages, a breakout board was necessary. An additional requirement for the breakout board was that it would be usable in a prototype of the module, requiring a small footprint.

Figure 3.2: Suggested minimal circuit for the AMG8831 [Pan12, p. 4]



The suggested minimal surrounding circuit for the AMG8831 [Pan12, p. 4] (see Figure 3.2) was used to design a circuit board with a footprint of 13mm x 14.4mm and a row of 100mil spaced pads for the I^2C and voltage supply pins. Since no more than one of

---

[1] While transfer of the whole array of temperature values would be preferred, due to constrains in packet size, only a simple presence detection flag will be transmitted

[2] Dual Flat No-leads

the sensors will be used, the configurable I²C device address was set to 1101000 via the AD_SELECT pin.

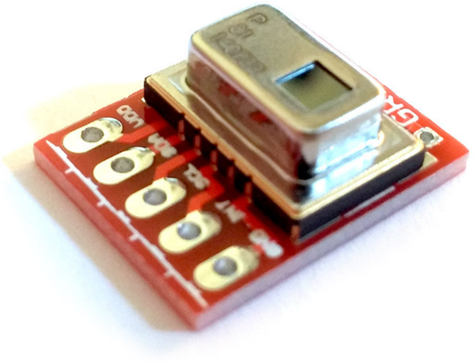


Figure 3.3: Breakout board with AMG8831 module soldered on

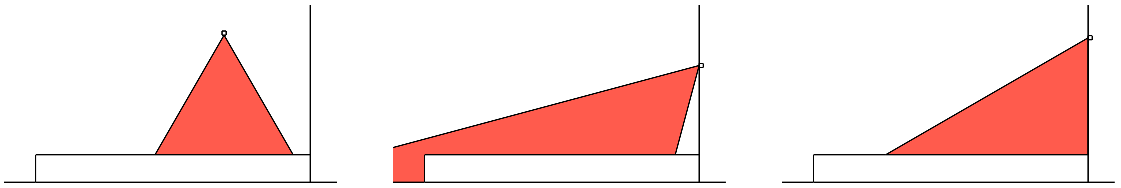
3.1.2 Mounting

This section discusses mounting options and their effect on the sensor readings, particularly in the case of monitoring presence in a bed. For both monitoring a bed and a cooking plate, the ideal mounting position would be directly above the area of interest with a vertical optical axis, not covering any areas that might influence the measurements. This is unlikely to be practical as in most cases this would require an extra assembly, making the sensor more obvious and decreasing user acceptance. In a real-world scenario, the sensor will most likely view the area of interest at an angle, leading to either the full area of interest not being fully covered or additional area being covered and introducing a possible source of wrong readings. Figure 3.4 shows 3 different possible configurations for mounting the sensor module over a bed, with the ideal configuration covering only the area of interest. Since this is unlikely to be possible, the experiments have been performed with the configurations (b) and (c).

3.1.3 Test setup

A test setup was built with the AMG8831 connected to an ATMega328 AVR microcontroller. An I²C-capable level shifter based on BSS138 FETs, shifter was used to allow I²C communication between the ATMEGA328 (with a logic level of 5V) and the AMG8831 (with a logic level of 3.3V).

A test program was written which initializes the AMG8831 and sets configuration registers. Whenever the PC sends a request, the ATMEGA328 retrieves the 64 12-bit (11 bits + sign) [Pan11, p. 14] temperature values from the AMG8831, converts them into human-readable format and transmits them to the PC along with the internal thermistor temperature.



(a) Sensor above the bed (ideal situation)

(b) Sensor mounted at a 45°angle on the wall

(c) Sensor mounted at a 60°angle on the wall

Figure 3.4: Different mounting options for the grid infrared sensor

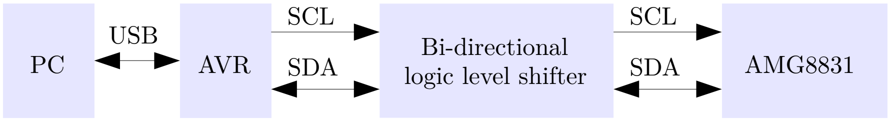


Figure 3.5: Test setup for the AMG8831

On the PC side, Java programs were developed to store the received data in log files as well as analyze and visualize the data. For all further heat-maps in this chapter, the gradient in Figure 3.6 is used.

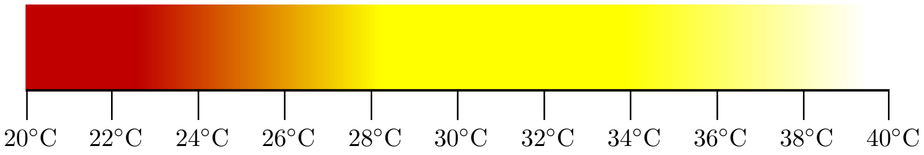


Figure 3.6: Heatmap used for visualization

3.1.4 Tests

The first data gathered were readings with the sensor pointing at a wall with a uniform temperature to test the noise level of the sensor. According to the data sheet, temperature readings have an accuracy of ±2.5°C [Pan12]. The connected computer was configured to acquire a reading of all pixel temperatures and the thermistor value every 30 seconds. Out of 380 measurements, the average values for each pixel were calculated. In Figure 3.7 the fluctuations in temperature of 20 pixels can be seen over a duration of 1000 seconds. The fluctuations for each single pixel typically stay within $\pm 0.5°C$ and the difference between the average temperature readings of the pixels was up to 2.5°C. It has to be noted that these measurements were not taken in a highly controlled environment.
For testing presence detection, the sensor was mounted above a bed with a 2m x 1m mattress at a height of 65cm and an angle of 45°, as seen in Figure 3.8. Over the course of a night, readings were acquired every 30 seconds well before going to bed and until a while after getting out of bed.

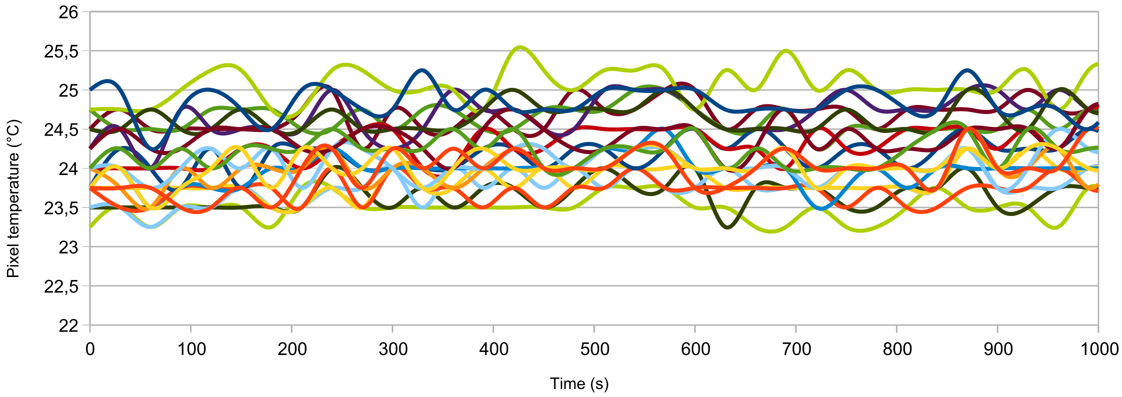


Figure 3.7: Graph of the noise of 20 sensor pixels over the duration of 1000 seconds, pointed at a surface with consistent temperature, with one measurement every 30 seconds.

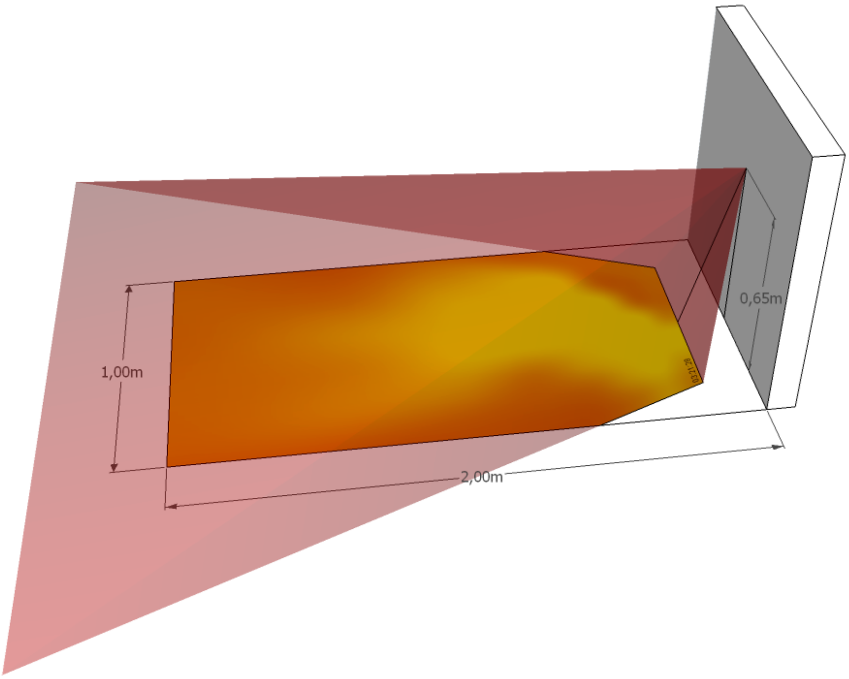
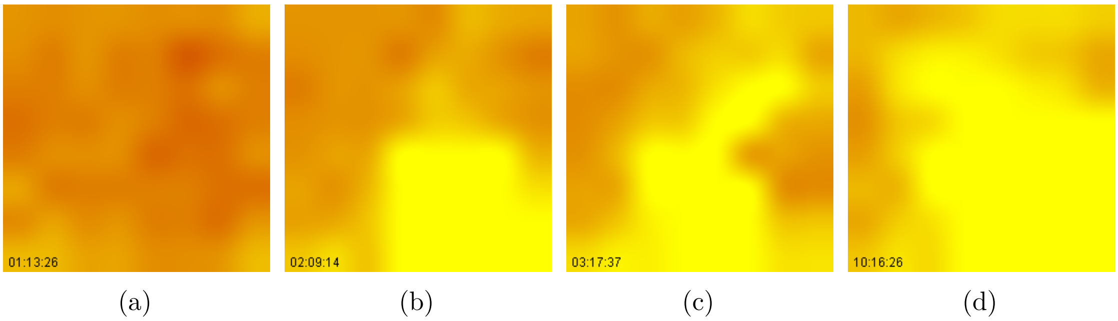


Figure 3.8: Initial test setup showing the location of the sensor, the field of view and the covered area.

Examples of the readings can be seen in Figure 3.9. For each set of 64 temperature readings, the maximum temperature as well as the average temperature were calculated. These values, along with the thermistor values were plotted on a graph. While it was fairly obvious on the chart which times the bed was occupied, it was difficult to set a threshold value to decide whether the bed was occupied or not. Since large parts of the



(a) (b) (c) (d)

Figure 3.9: Sensor readings: (a) empty bed, (b)(c)(d) occupied bed

body are covered when lying in bed leaving possibly only the head partially exposed, simply setting a temperature threshold for the maximum value would not work reliably. The change in average temperature in an occupied bed on the other hand is too small to make a definite decision.

As another step of processing the input, the variance of the 64 values was calculated. The variance should be low when the bed is unoccupied as it can be assumed that the unoccupied bed will have a uniform temperature. When the bed is occupied, the variance will be significantly higher as parts of the reading will have a higher temperature, and based on the location of the sensor it can be assumed that the whole sensor area will never be fully covered by a body. Therefore, the variance was chosen to be the best measure to determine whether the bed was occupied and calculated as well for each received sensor reading. Based on known data, a preliminary variance threshold of 0.8 was set with a variance below 0.8 meaning that the bed was not occupied. The threshold, applied to the variance graph, accurately shows times then the bed is occupied. All data can be seen in Figure 3.10.

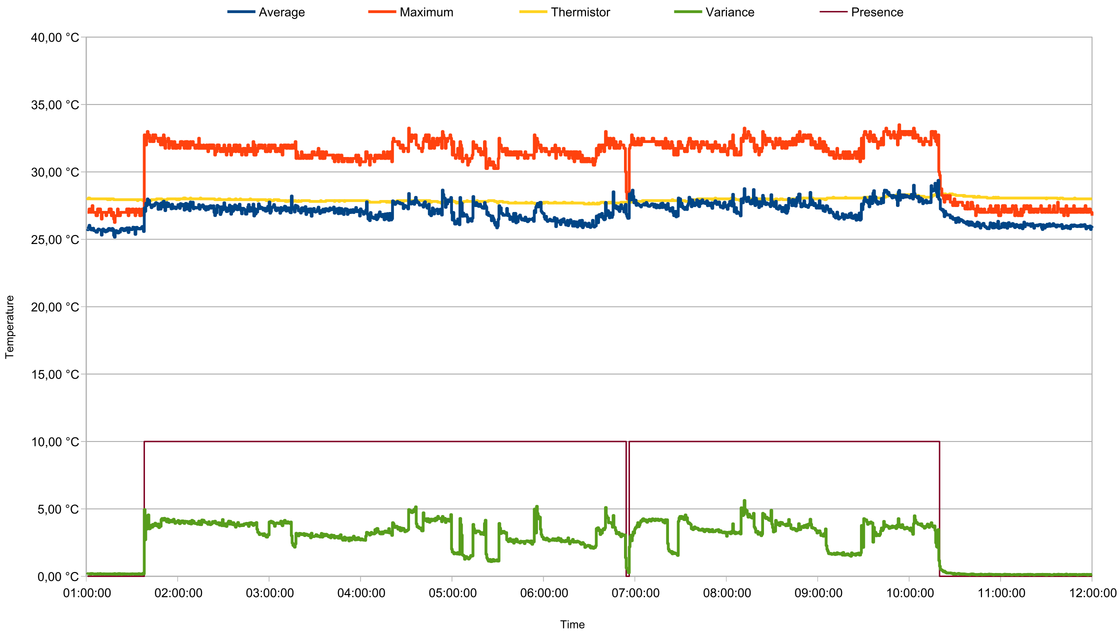


Figure 3.10: Graph showing minimum, average, variance, thermistor value and detected bed occupancy times of the first test.

Based multiple tests, it was concluded that a sensor location with increased coverage of the upper area of the bed (where the head is located) would lead to better readings, as this is the area where the body would most likely not be covered with a blanket. Figure 3.11 shows the revised test setup, with the sensor being mounted at a height of 85cm with the optical axis pointing down at a 60°angle. The threshold value for the occupancy detection based on the variance of the measured values was lowered to 0.7 which provided more accurate results in further tests. Results of a test with the revised setup can be seen in Figure 3.12.

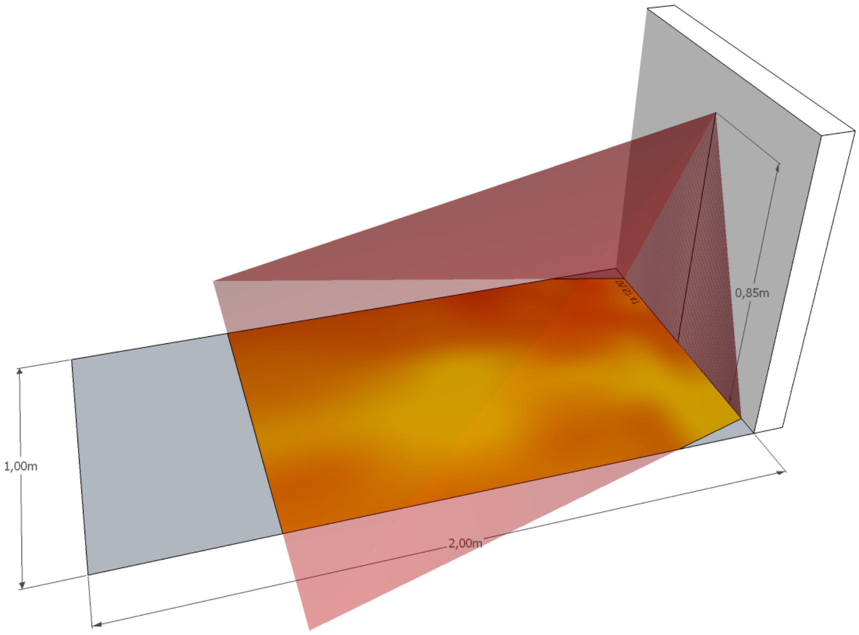


Figure 3.11: Revised test setup with the sensor positioned higher and at a 60°angle

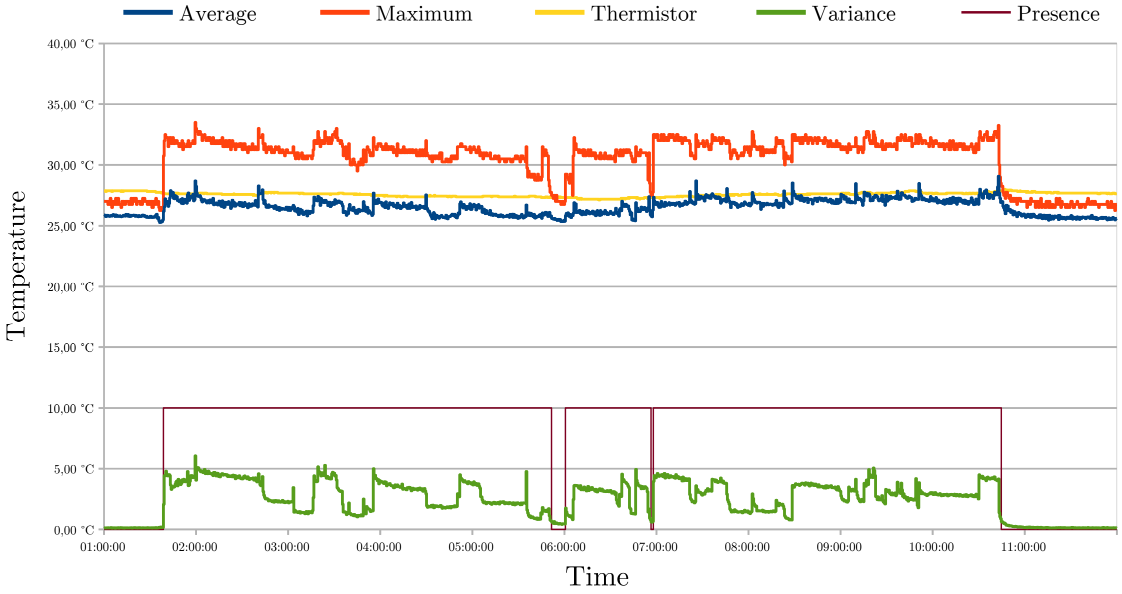


Figure 3.12: Graph showing minimum, average, variance, thermistor value and detected bed occupancy times with the revised test setup.

When a person occupies a previously unoccupied bed, the next reading will instantly detect the presence. However, when a person exits an occupied bed, the currently implemented presence detection will still wrongly assume that the person is present for up to 90 seconds. This is due to the heat imprint left on the bed by a person after they lie in the same place for a while. Figure 3.13 and Figure 3.14 show a situation where the person left the bed. The actual time of leaving and the detected time are 90 seconds (3 readings) apart. In this case, it can be seen that it takes about 6 minutes for the bed to return to ambient temperature.

While additional measures could be taken to improve the speed of detection of a person leaving the bed, doing so would increase the chance of false positive detection. It is possible to retroactively correct the system by looking for a sharp drop in the maximum temperature shortly before the system stops detecting a presence and assuming it as the time when the person left the bed. Since a delay in detection of 90 seconds is tolerable, it is, while worth noting, not seen as a problem in this context.

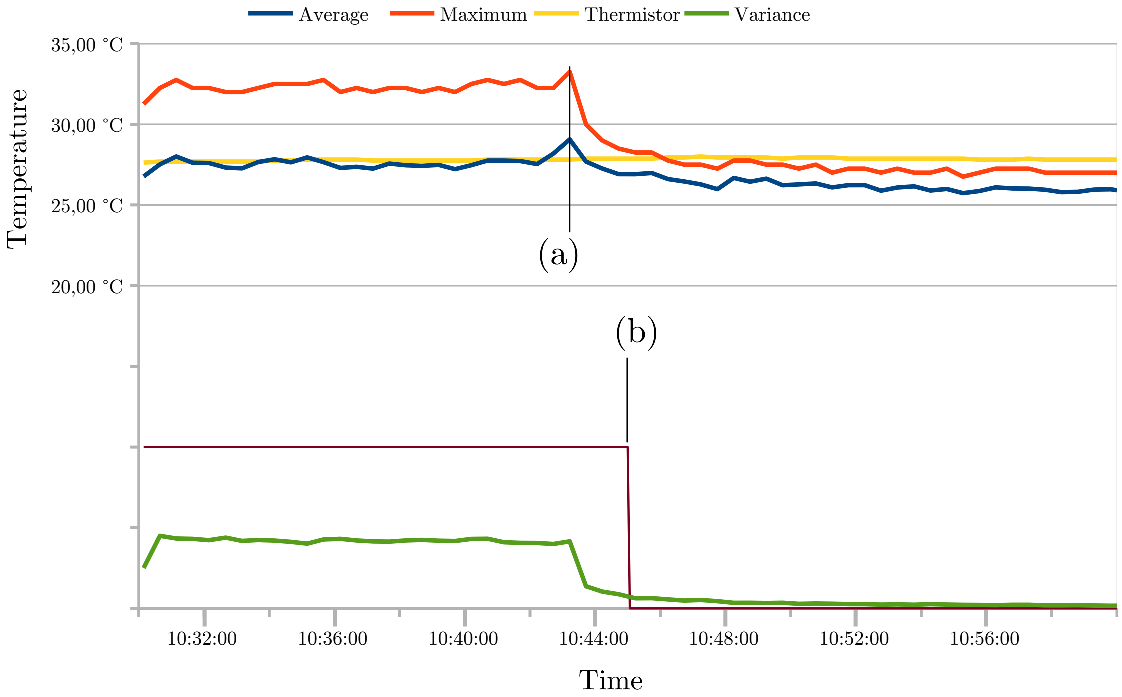


Figure 3.13: Graph showing: (a) time when the person leaves the bed; (b) time when the system stops detecting presence

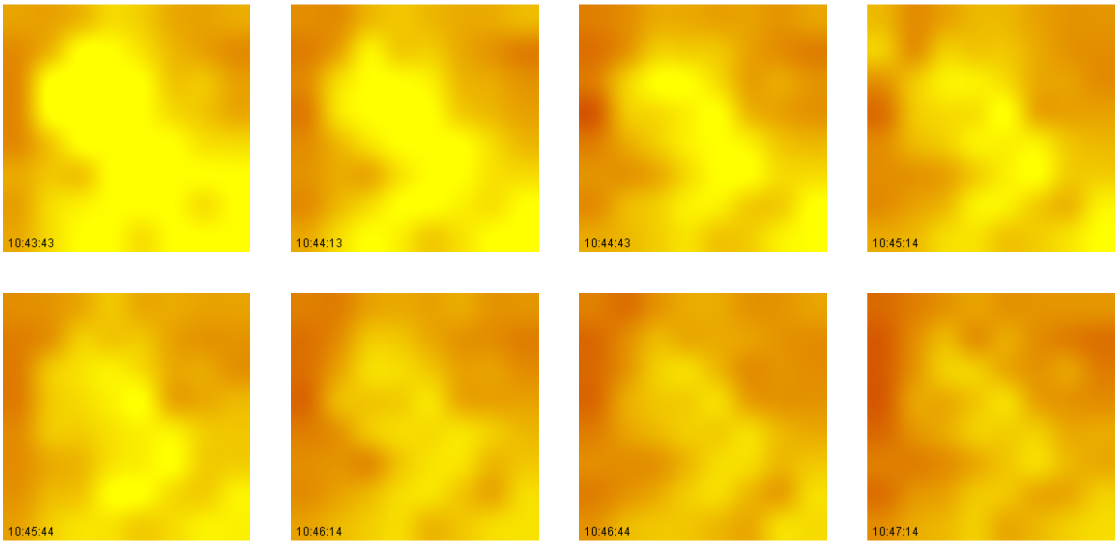


Figure 3.14: Visualization of the sensor readings in 30-second intervals after the bed is unoccupied (from the same data set as Figure 3.13)

3.1.5 Final prototype

Based on the tests, a prototype using an EnOcean STM310 module was developed. A diagram of the prototype can be seen in Figure 3.16. Ideally, the whole system would rely on energy harvesting, in this case an external battery is needed due to the 4.5mA current consumption of the AMG8831, which ideally should have 15s of setup time after powering on or returning from stand-by or sleep mode to stabilize the sensor output [Pan12, p.2]. In the prototype, the ATMega328 microcontroller retrieves the sensor values and calculates the variance. In a finished product, the code would ideally run completely on the EO3000I of the STM310, eliminating the need for an additional microcontroller. This setup requires maintenance due to the sensor and ATMEGA328 being powered by a battery. The STM310 does not need to supply power to periphery, making it possible to monitor the status of the battery and to send an alert if the battery needs to be changed.

The firmware of the ATMega328 does the following:

- Initialize the AMG8831 upon start-up
- Read all pixel temperature registers into a float array
- Calculate the average pixel temperature
- Calculate the variance of the 64 pixel values
- Change the value of an I/O pin depending on whether the variance is above a user-defined threshold

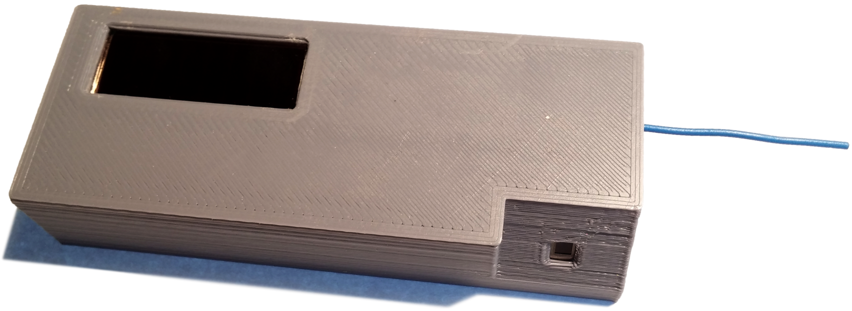


Figure 3.15: Prototype of the infrared sensor array module with 3D-printed case and a fixed 60° angle

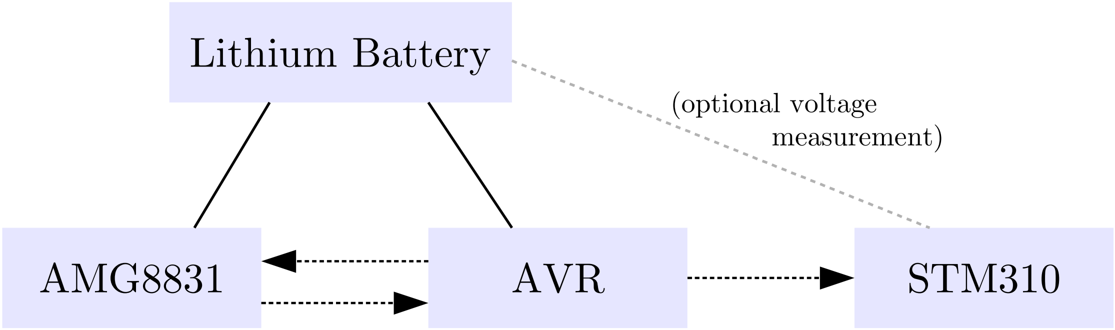


Figure 3.16: Diagram of the infrared sensor array module prototype

The function for reading the pixel temperature values into a float array is shown in Listing 3.1. On an 8MHz ATMega328, the execution time for the reading of the pixel temperature values from the AMG8831 has been measured to be 107ms (at an I^2C clock speed of 400kHz), while the execution for calculating the average and variance values is 4ms. The latter operations are done using float operations and could be optimized by doing them with Integers. This will only marginally shorten the total processing time however, considering the I^2C operations take significantly more time. Using the standard STMSEN firmware, the STM310 reads the output of the ATMega328 at intervals configured by the CONF0 and CONF1 pins. Due to the logic voltage difference, a simple voltage level translator is inserted between the output pin of the ATMEGA328 and the input pin of the STM310.

```
/**
 * Writes the temperature values in Celsius into a float array.
 * @param result Float array of the size 64 used for storing results
 */
void readValues(float result[]){
  byte pixelAddrL = AMG_REGISTER_T01L;
  for (int pixel = 0; pixel < 64; pixel++) {
    delay(1);
    Wire.beginTransmission(amg_addr);
    Wire.write(pixelAddrL);
    Wire.endTransmission();
    Wire.requestFrom(amg_addr, 2);
    byte lowerLevel = Wire.read();
    byte upperLevel = Wire.read();

    int temperature = ((upperLevel << 8) | lowerLevel);
    if (temperature > 2047) {
      temperature = temperature - 4096;
    }
    result[pixel] = temperature * 0.25;
    pixelAddrL +=2;
  }
}
```

Listing 3.1: The function to read the pixel temperature values from the AMG8831

3.2 Non-invasive current detection module

The method for non-invasive inductive current detection presented in this section works by filtering and amplifying the tiny currents created in an inductor placed near a 2-wire or 3-wire power cable carrying alternating current. Measuring the currents is challenging as the electric fields of the phase and neutral wires almost completely cancel each other out, an issue that does not exist when measuring on a single conductor. However, with large enough amplification of the much weaker current the inductor provides when measuring on a 2-wire or 3-wire power cable, it is still possible to detect the electric field which corresponds to the amount of current in the conductors. All further tests are performed with power cables carrying 230V AC.

3.2.1 Circuit design

The initial circuit design is roughly based on the circuit in the Extech DA30 [Ext02] non-contact AC current detector. Through reverse-engineering the circuit board it was determined that the detection circuit most likely consisted of an inductor with a connected operational amplifier and low-pass filter.

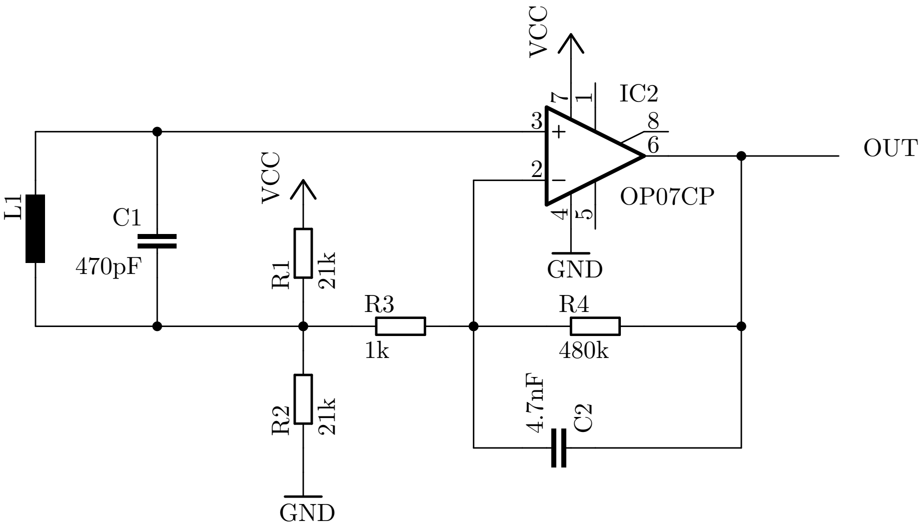


Figure 3.17: Current detection circuit

The test circuit for current detection uses an OP07 operational amplifier capable of 3.3V operation acting as a low pass and amplifier. In the further tests, an ideal values for L, R3 and R4 are determined. Initially, the circuit was built on a breadboard with an easy way to mount different power wires in front of the inductor.

3.2.2 Test setup

For finding the right component values and performing test measurements, a test setup was created by connecting the OUT pin of the circuit in Figure 3.17 to the ADC of an ATMEGA328 microcontroller and sending the logged values to a PC via USB. The reference voltage of the ATMEGA328 ADC was set to the supply voltage of the circuit (3.3V). The ATMEGA328 ADC has a resolution of 10 bit [Atm16], leading to a resolution of roughly 3.22mV/LSB. The inductor was placed orthogonally to the power cable as seen in figure 3.18.

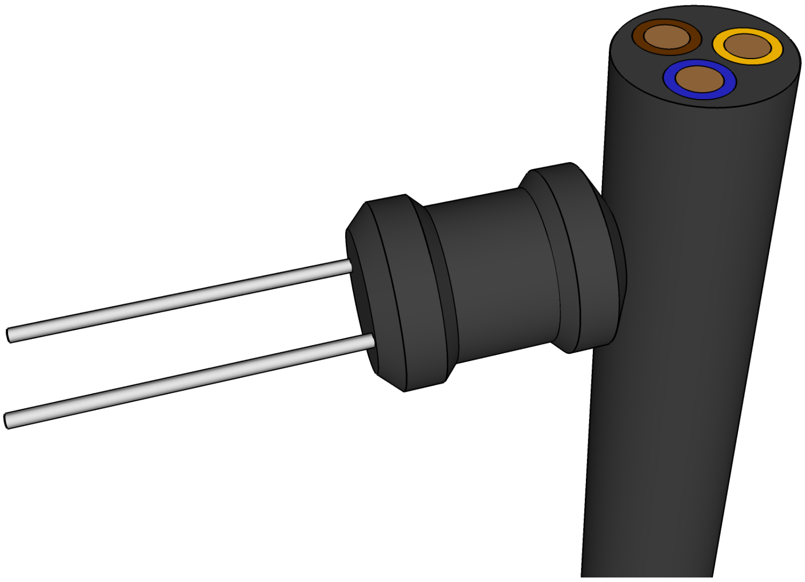


Figure 3.18: The inductor is positioned orthogonally on the power cable.

3.2.3 Experiments

The goal of the following experiments is to find out ideal configuration of the circuit for an ideal measurement of the load using the 10-bit ADC of an ATMEGA328. Each single measurement is done using the following code:

```
#define MEASURE_DURATION  20 // 20ms = 1 period at 50Hz
#define ADC_RANGE 1024 // ATMega328 ADC 10 bit Range
#define AREF  3.3 // Analog reference voltage [V]
// Measures the double amplitude of the analog input signal by calculating the difference between the minimum and the maximum
float measure_amplitude() {
  unsigned long endtime = millis() + MEASURE_DURATION;
  int val, i;
  int min = ADC_RANGE, max = 0;

  // Measure for 20ms
  for (i = 0; millis() < endtime; i++) {
    val = analogRead(A0);
    if (val > max)
      max = val;
    if (val < min)
      min = val;
  }
  return ((float) max - min) * ((AREF * 1000.0) / (float)
     ADC_RANGE); // result in mV
}
```

Listing 3.2: Function to measure the amplitude of the signal generated by the current sensor

The first test was performed with the following values:

- VCC = 3.3V
- L1 = 100μH
- R3 = 1kΩ
- R4 = 470kΩ
- Load = 1000W

The gain A_V of the initial configuration is therefore:

$$A_V = \frac{R4}{R3} = \frac{470\text{k}\Omega}{1\text{k}\Omega} = 470$$

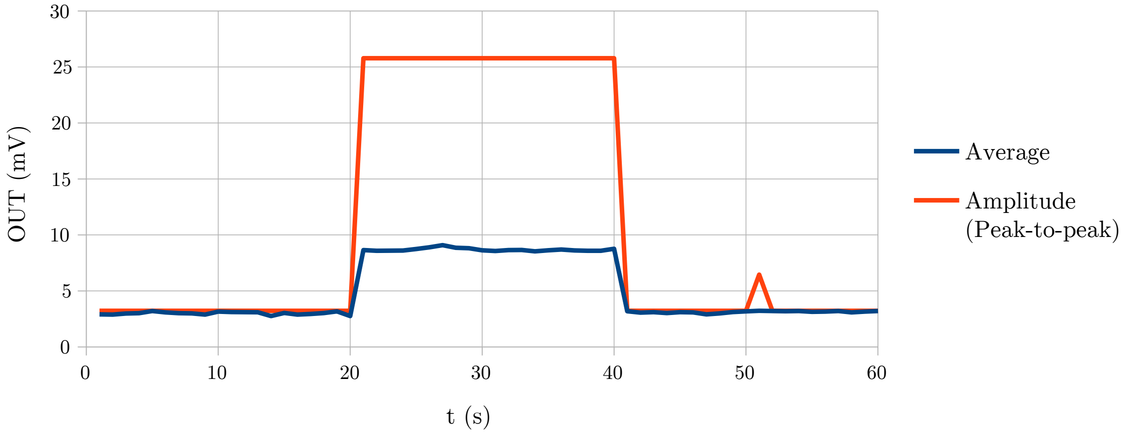


Figure 3.19: Results of the first test with a 1000W load. (A_V=470, L=100μH)

For a load of 1000W, the results were clear but very weak with a V_{PP} of about 25mV. Figure 3.19 shows the measurements of a 1000W load being turned on for 20 seconds with a reading once per second. To increase the accuracy of the measurements, the gain was increased to A_V=1424.24 by changing R3 to a 330Ω resistor.

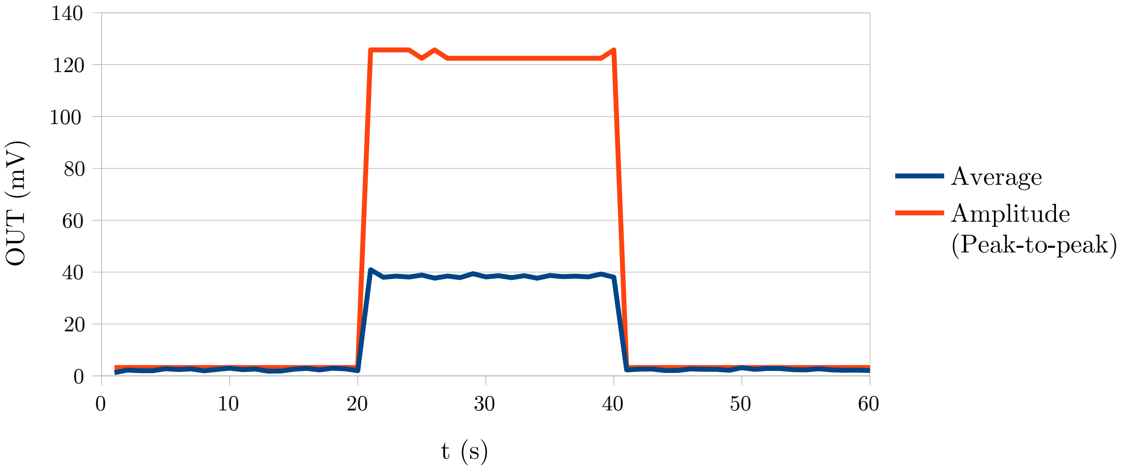


Figure 3.20: Results of the second test with a 1000W load. (A_V=1424.24, L=100μH)

Increasing the gain leads to more accurate readings, seen in Figure 3.20. To further improve the sensitivity, the inductor has been changed to L=10mH. In Figure 3.21, a measurement over one period with a load of 1000W is shown with the 3 different test configurations.

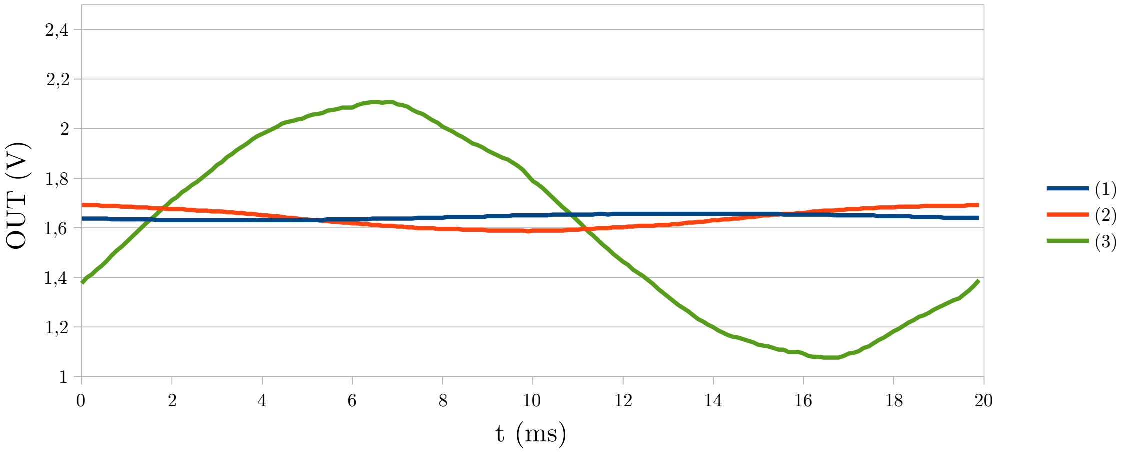


Figure 3.21: Measurements with a load of 1000W: (1) A_V=470, L=100μH, (2) A_V=1424.24, L=100μH, (3) A_V=1424.24, L=10mH

With optimal placement, the circuit is able to detect loads of 15W and higher accurately by applying a simple threshold to the measured V_{PP} values. For each device, the threshold has to be set separately as the output value is largely influenced by the location of the sensor and the type of power cable. Once the sensor is mounted in a fixed position, the sensor readings at a certain current flow stay consistent. The amount of load corresponding to the sensor output has been determined to be non-linear. This has been tested by placing the sensor on the cable of a power distributor and connecting loads between 4W and 1160W while measuring the exact wattage using a power meter. A graph of the results can be seen in Figure 3.22.

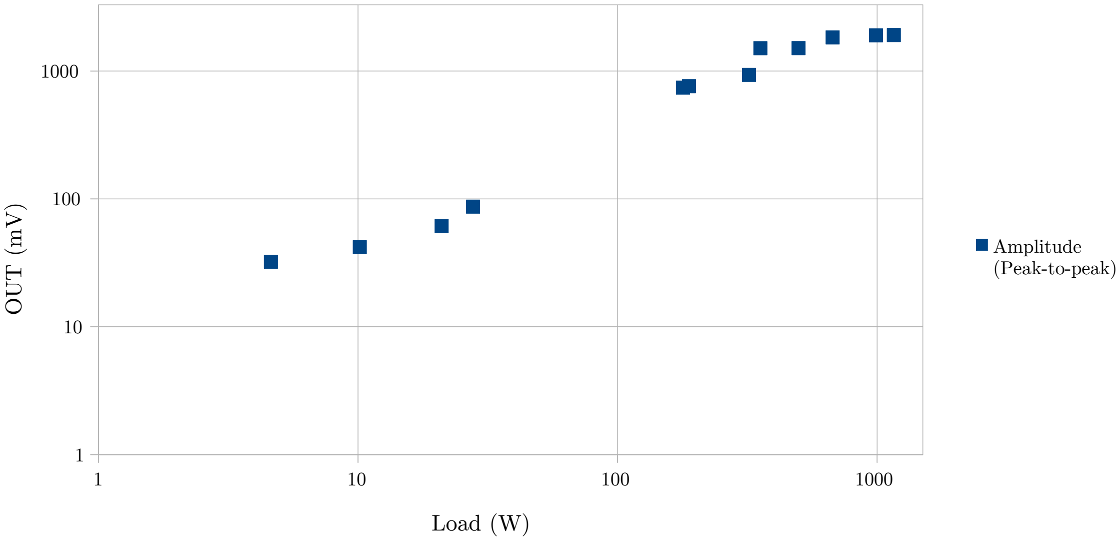


Figure 3.22: Different loads with the corresponding sensor output on a logarithmic graph

3.2.4 Influence of sensor location

In previous tests, it was determined that the location of the sensor relative to the power cable significantly influences the output of the sensor circuit. This is due to the different positions of the phase and neutral wires in relation to the inductor, depending on the angle at which the inductor is placed on the power cable. In a first measurement, it was confirmed that the two ideal positions with the maximum output values are 180° apart. The two least ideal positions are located 90° from each of the ideal positions. In a setup (1000W load, A_V=1424.24, L1 = 100μH), 10 measurements were taken at each 45°of power cable rotation. The power cable was 8mm in diameter with phase, neutral and protective earth wires. The results can be seen in Figure 3.23.

It was decided that more accurate measurements were necessary to determine the effect of the inductor location on the sensor output. A test setup was created to measure the output signal with the inductor rotated in 5° increments around the power cable between each successive measurement. This was done for a 3-wire power cable (see Figure 3.24) as well as a 2-wire power cable (see Figure 3.25). While with the round 3-wire power cable the placement of the inductor was unproblematic, the oval shape of the 2-wire power cable made it difficult to perfectly align the inductor to the center of the cable. This partially explains the distortions in the readings compared to the more smooth results with the 3-wire cable. Furthermore, the load connected to the 2-wire cable was just 20W (compared to the 1000W load connected to the 3-wire power cable), leading to more noise in the readings.

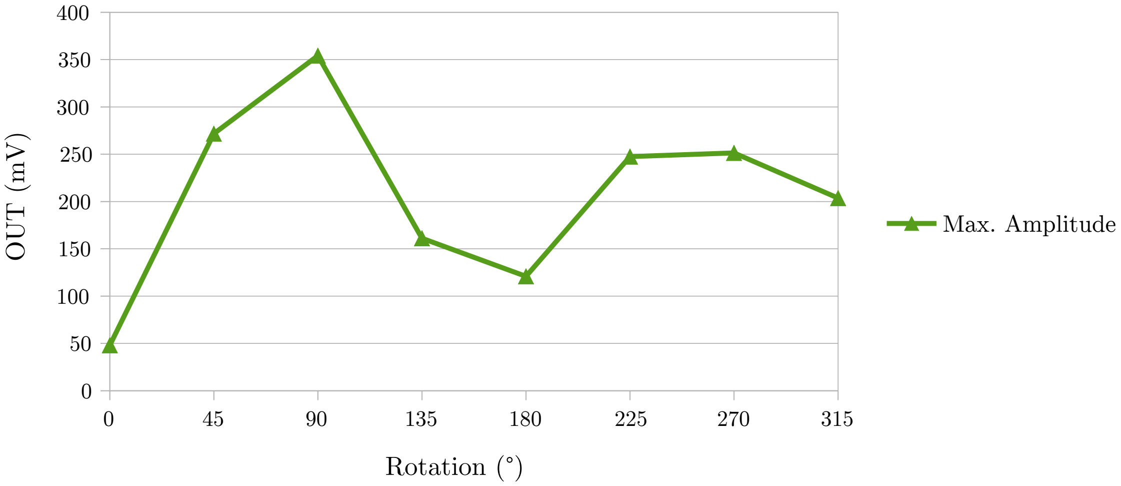


Figure 3.23: Output of the current detection circuit with the inductor placed in 45°steps around a 3-wire power cable with a 1000W load.

Figure 3.24: Output of the current detection circuit with the inductor placed in 5°steps around a 3-wire power cable with a 1000W load.

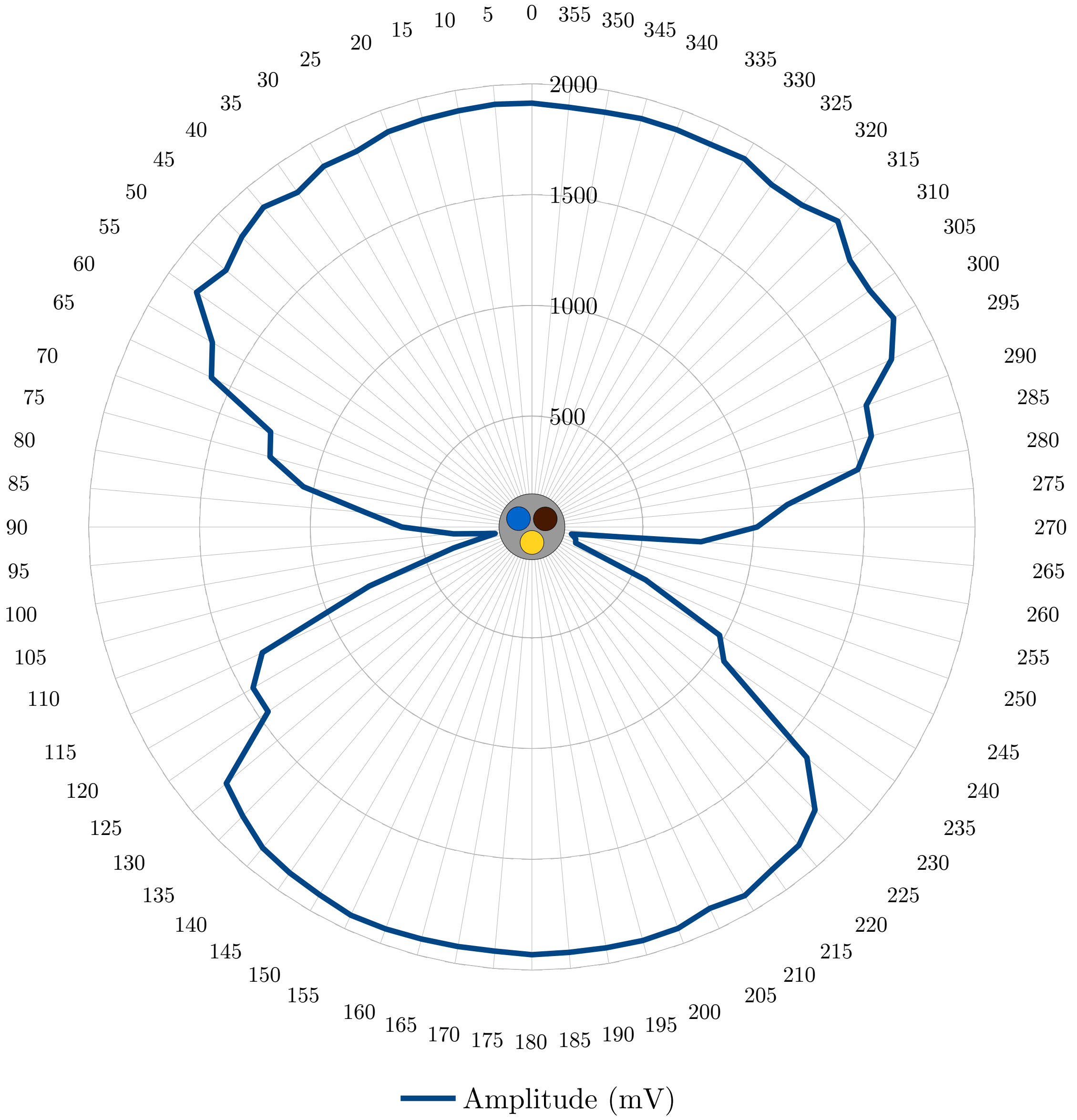
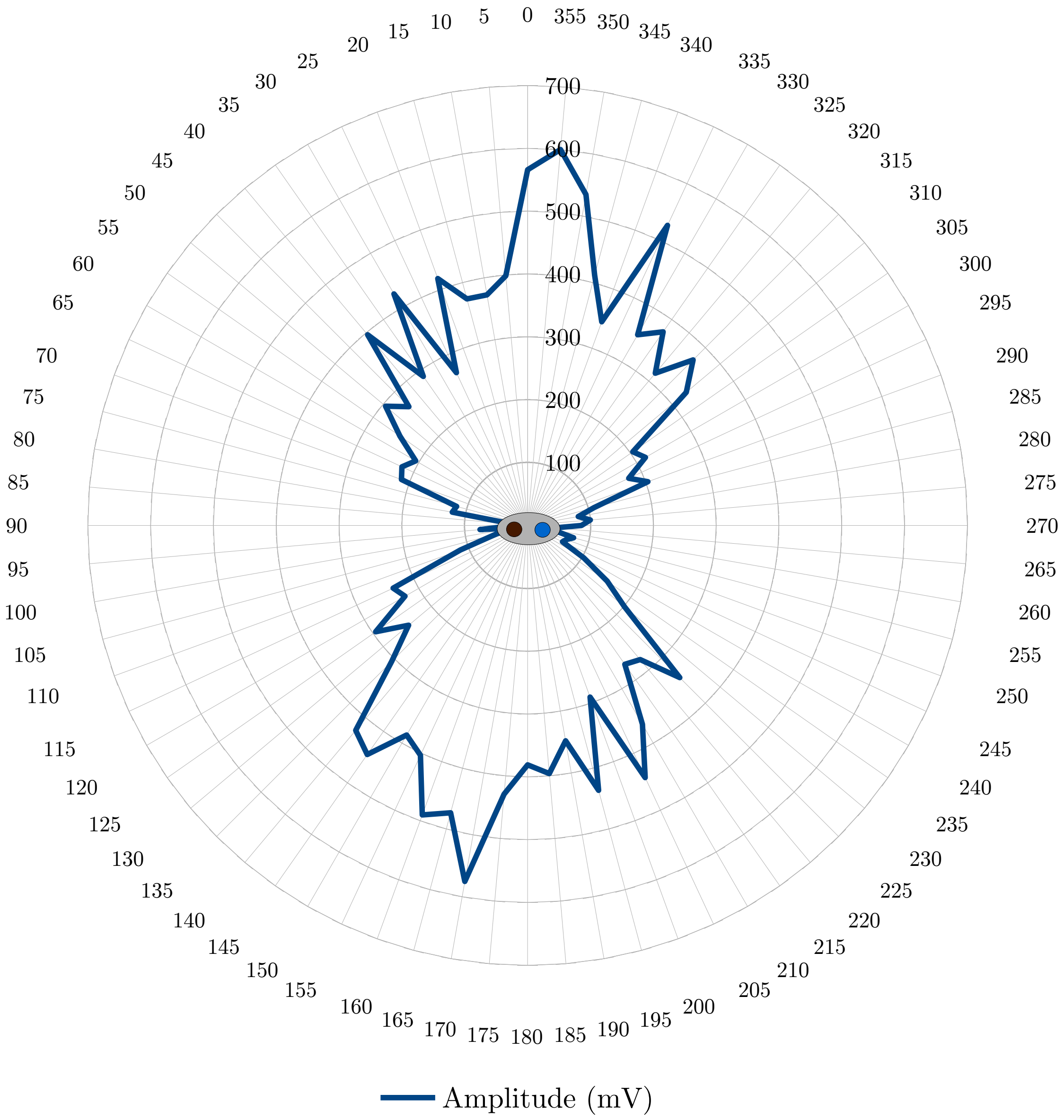


Figure 3.25: Output of the current detection circuit with the inductor placed in 5°steps around a 2-wire power cable with a 20W load.



3.2.5 Tests

The sensor circuit was tested with different electric devices. Figure 3.26 shows the readings of the sensor when attached to a 200W LCD TV. The TV was turned on 2 times which can be clearly seen on the graph. Small changes in power consumption occurred when switching channels which the sensor readings clearly show. For determining whether this appliance is turned on, a simple threshold would give accurate results.

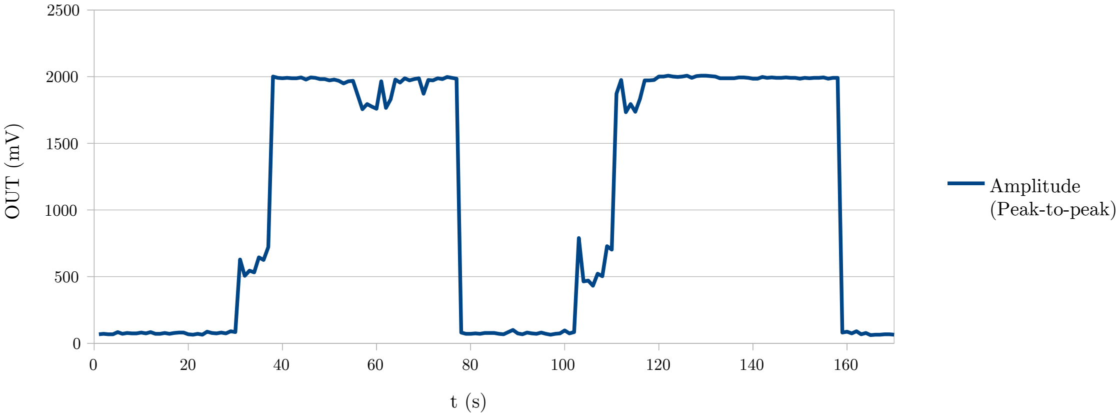


Figure 3.26: Sensor output when connected to the power cable of an LCD TV. The TV was turned on and off two times.

A rice cooker was tested with the sensor. The appliance offered 2 modes, cooking and warming up. When cooking, the cooker consumes 700W of power. What makes this a good candidate for testing is the fact that this device needs to be unplugged when not used. Whenever the device is plugged in, it will at least be in warming up mode and could pose a hazard when the user forgets to unplug it. Figure 3.27 shows the sensor readings when attached to the power cable of the rice cooker. After about 16 minutes of cooking, the cooker switches into warming mode automatically. While the sensor output seems small when the cooker is in warming mode compared to when the cooker is in cooking mode, it still can be accurately determined whether the cooker is plugged in or not. Due to the 2 different modes of operation being easy to tell apart, a system could keep track of all 3 possible states (cooking, warming and unplugged) of the device.

In a test performed to gather insight of the amount of information the sensor module can give about a device, both the non-invasive current sensor and the water flow sensor prototype were connected to the power cable and water supply pipe of a washing machine. A 30°C cycle was started and the results were logged at a rate of one measurement per second. The results can be seen in Figure 3.28. The water flow sensor readings clearly indicate the exact times the washing machine used any water, while the current sensor readings give a lot of information about the washing cycle. While this is not too practical in our context, it shows that this type of sensor could be used in different, more specific scenarios.

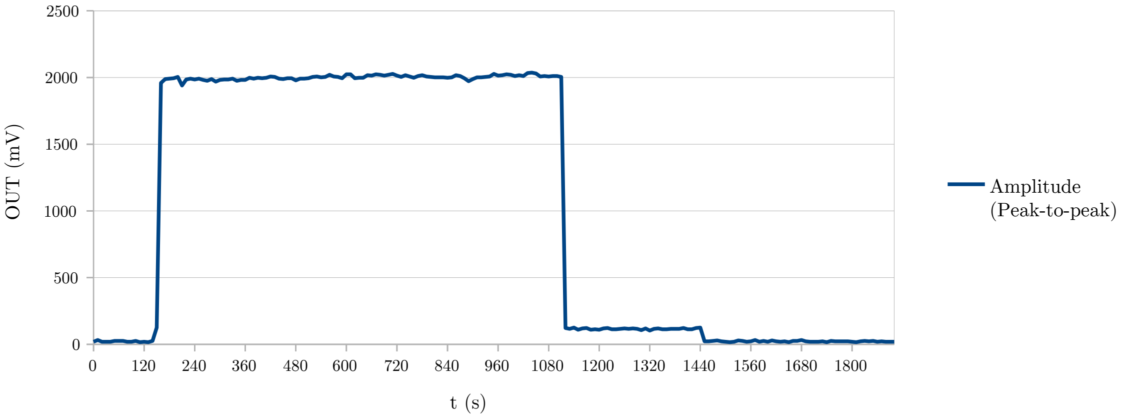


Figure 3.27: Sensor output when connected to the power cable of a 700W rice cooker. Both cooking and warming mode can be clearly seen in the graph

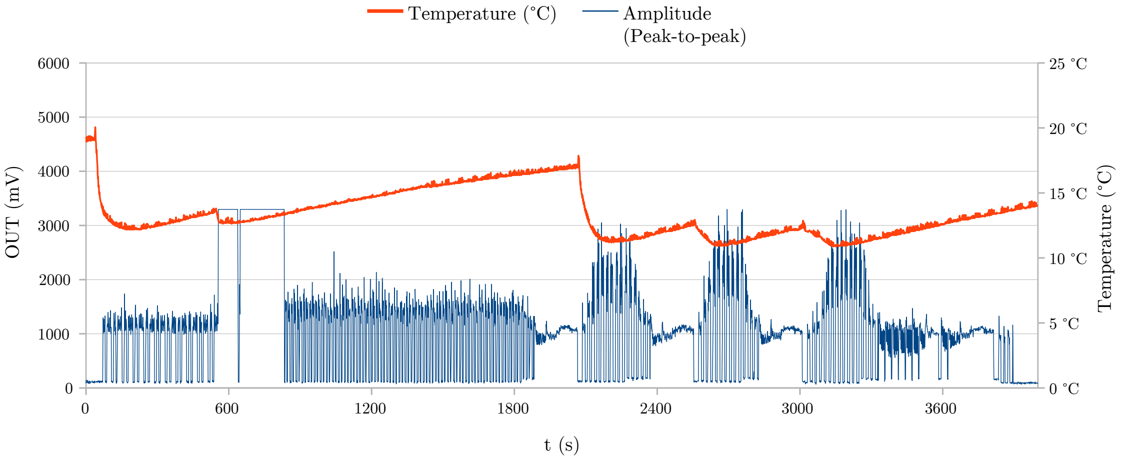


Figure 3.28: Sensor output when connected to a washing machine doing a 30°C cycle including tumble-drying. Includes temperature sensor data

3.2.6 Final Prototype

Due to the STM3xx only supplying 1.8V for periphery, the Operational amplifier had to be replaced by a suitable equivalent which operates at a supply voltage of 1.8V. For this, the conveniently pin compatible Microchip MCP6231 was chosen, which accepts a supply voltage in the range of 1.8V to 6.0V [Mic09]. The MCP6321 typically operates with a supply current of 20μA which is acceptable for use with the EO3000I, which for reference uses typically about 10μA when not sending radio packets [EnO13a, p. 22]. A transmit cycle for a 4BS packet takes about 40ms, while measuring one period of mains power takes at least 20 ms. This causes the wake cycles with the firmware taking the measurement to be 50% longer than the wake cycles using — for example — the default STM310 firmware which simply reads temperature and single values for digital and analog inputs.

For the final prototype, a smaller version of the sensor was built which can be easily attached to power cables. (see Figure 3.29) The sensor was connected to the power supply and the ADIO0 pin of the STM310 module and a custom firmware based on the default STMSEN firmware was developed which measures the V_{PP} value of the sensor for one period and transmits the readings at specified intervals.

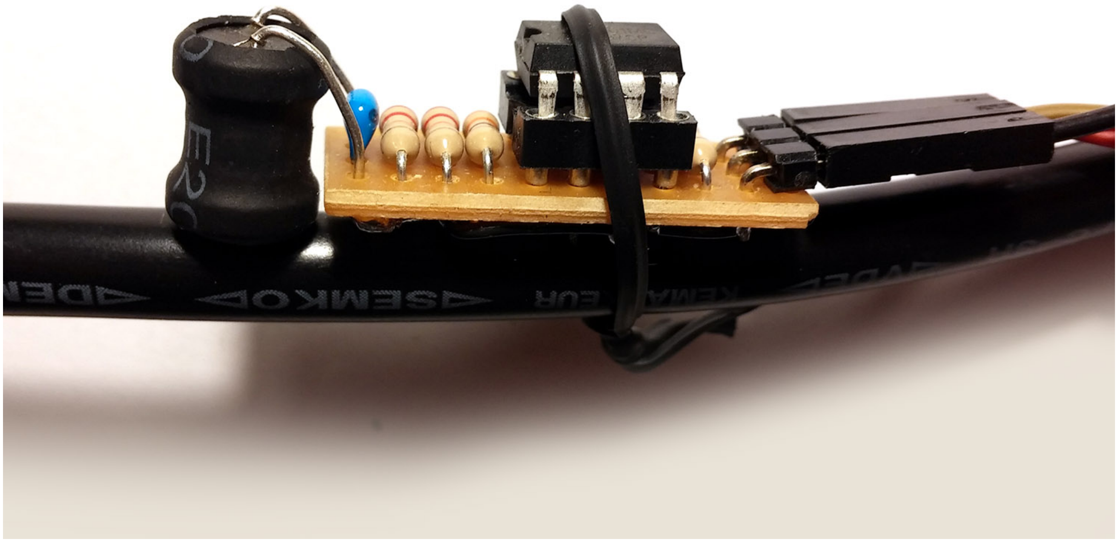


Figure 3.29: Current sensor prototype, attached to a 3-wire power cable

3.3 Water flow detection module

As discussed in Section 2.10, temperature sensors are being used to detect water flow. One or two of these sensors are attached to the warm and/or cold water pipe(s) to detect water flow by measuring changes in pipe temperature. Other sensors, such as a piezoelectric vibration sensor and a capacitor microphone have been unable to provide usable data. While ultrasonic flow measurement would be the most accurate method of measuring water flow, it has not been chosen due to the high price, low availability and possibly requirements in terms of power usage and supply voltage that the module would not be able to fulfill.
The *TMP75B* I^2C temperature sensor provides a resolution of 0.0625°C/LSB and accepts supply voltages between 1.4V and 3.6V [Tex14], making it ideal for operation in the test setup in a 3.3V environment while being able to connect to the STM3xx module without the need of level shifting. Furthermore, the TMP75B provides an alert signal, which is a simple logic output based on whether the measured temperature is above or below a configurable threshold. This could provide an alternative to I^2C communication and save a few milliseconds of processing time.

3.3.1 Test Setup

The test setup consists of an Arduino running at 5V with one or two TMP75B sensors connected via I^2C through a bidirectional logic level shifter. A TMP75B sensor is connected to each pipe that is to be measured using thermal paste and cable binders, pressing the upper side of the IC package against the pipe. (see Figure 3.30) The ATMega328 on the Arduino runs a simple firmware which reads the temperature registers, converts the temperature to Celsius and sends the data to a PC over a serial port at specified intervals.

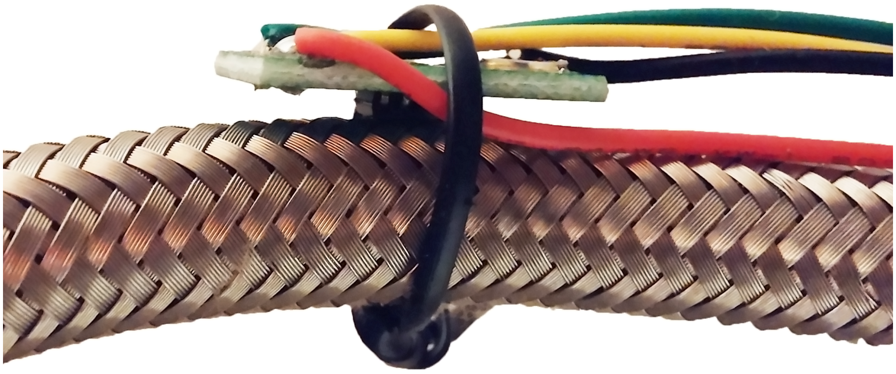


Figure 3.30: A temperature sensor connected to a flexible water tube

3.3.2 Tests

Tests have been performed in typical places where pipes would be accessible in a home and measurement would provide useful information. The first test was performed on the flexible tube (rubber, covered with metal mesh) connected to a toilet water tank. One sensor was connected with thermal paste and measurements were taken every 2 seconds. Figure 3.31 shows the result of the measurements. The toilet was flushed 3 times within 20 minutes. Each flushing can be clearly seen on the graph as a sharp drop in temperature, followed by a slow rise.

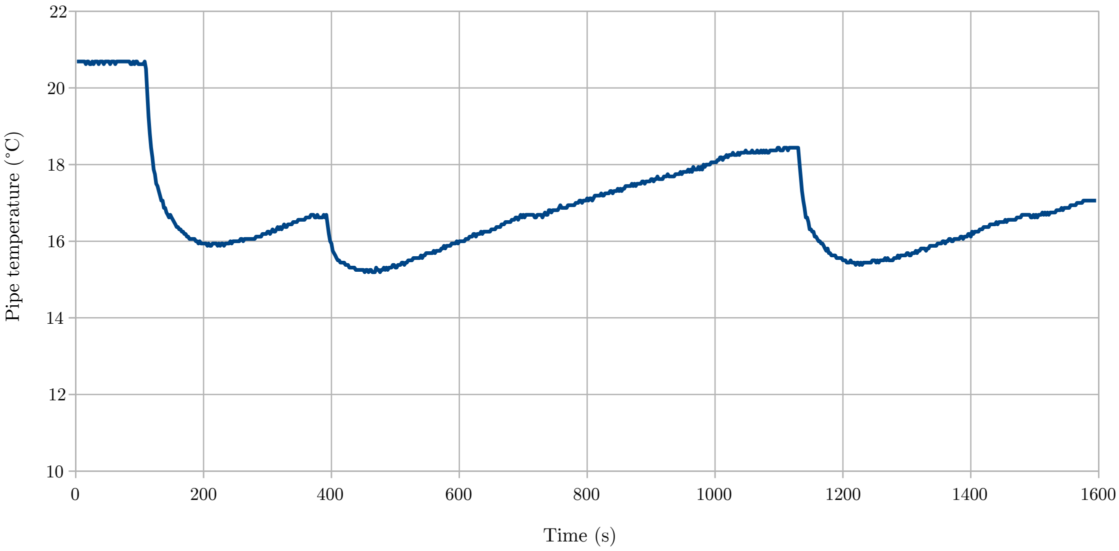


Figure 3.31: Temperature readings of the sensor connected to the flexible tube of a toilet water tank

Another test was done by connecting the sensors to the flexible tubes of a bathroom sink and measuring temperatures of both hot and cold water supply. Typical use has then been simulated over the period of 110 minutes and logged on a graph every 30 seconds. Figure 3.32 shows the results of the measurements. At position (a) the tap was turned on for about 10 seconds for washing hands in a middle position, clearly showing change on both hot and cold water pipes. At position (b), only cold water was used while at position (c) only hot water was used. Worth noting is how after (b), hot and cold water both approach almost the same temperature. Between position (d) and (e) hot water was turned on for 21 minutes to simulate an unusual situation. In contrast to turning the water on for a short while where a slow decline in temperature follows a steep climb, the temperature slowly rises after the steep climb, approaching the actual hot water temperature.

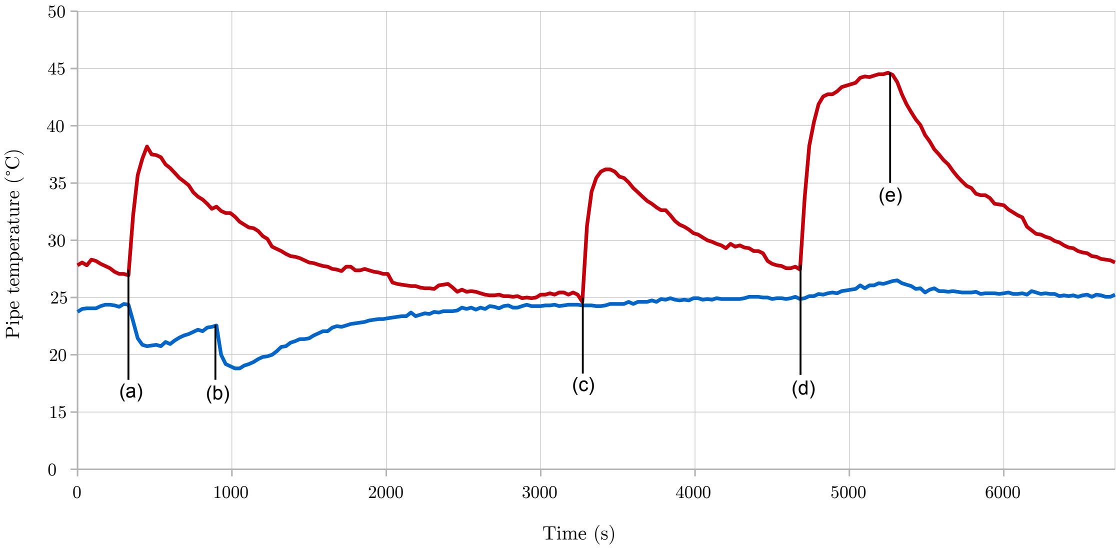


Figure 3.32: Temperature readings on hot (red line) and cold (blue line) water tubes of a bathroom sink while simulating typical usage (a)(b)(c) and an unusual event (d)(e)

One more noteworthy test was performed using both temperature sensors and the non-invasive current sensor developed in section 3.2. One temperature sensor was mounted on the water supply tube of a dishwasher while the current sensor was mounted on the power cable of the dishwasher. While simple on/off data would be sufficient for most applications that just monitor whether devices are being used, this test was performed to gather the maximum amount of data possible using those two sensors. The temperature sensor readings can be seen in Figure 3.28 and clearly shows every moment where the washing machine starts to use fresh water.

3.3.3 Final Prototype

A final prototype (shown in Figure 3.33) was developed by connecting two TMP75B sensors to an STM310 module via approximately 10cm long wires and pin headers, making it possible to remove one of the sensors when not needed. The I^2C reference implementation [EnO11d] has been used as base for a firmware that reads out both sensor values and transmits them in a telegram every 30 seconds.

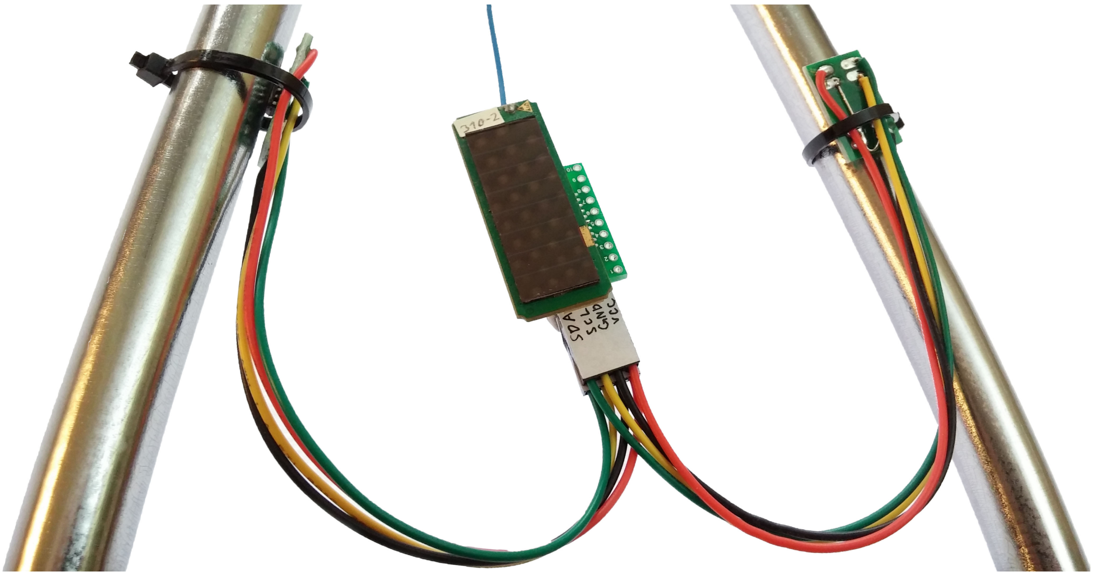


Figure 3.33: Prototype of the water flow detector module

CHAPTER 4

Results and Analysis

In this chapter, results of the experiments conducted using the three developed sensor modules will be discussed.

4.1 Presence detection

Presence detection was realized using an AMG8831 8x8 infrared grid array sensor module. The data from the module are read and processed by an ATMega328 which outputs a single bit indicating presence and supplies it to the STM310 module via a digital I/O port. The module was tested above a bed in various configurations and performed presence detection with a near 100% accuracy by calculating the variance of all 64 pixels and applying a threshold.
Possible scenarios where this sensor would have difficulty with providing conclusive readings were determined to be:

- When the ambient temperature is very close to body temperature
- When the whole body is covered (e.g. by a blanket)
- When another source of heat is within the field of view of the sensor

When the sensor module is placed in a way that it is not influenced by other sources of heat in a room, it can be assumed to work reliably in a wide variety of scenarios. Further tests would be necessary to determine the accuracy when the ambient temperature is very close to the body temperature. In case of inconclusive readings due to not enough difference between ambient temperature and body temperature, the system could be supplemented by a second sensor, for example a PIR sensor.

4.1.1 Power consumption

It is recommended for the AMG8831 to be in normal mode for at least 15 seconds prior to retrieving measurement values in order to get stable readings [Pan11]. While 4 seconds have been sufficient in tests, the sensor still needs to be in the most current consuming mode for a long time for a single reading. In order to provide the necessary power, a SAFT LS 14500 3.6V lithium battery [SAF09] was used as power supply for the ATMega328 and the AMG8831 because energy harvesting alone would not be able to supply the periphery. Due to this setup, the STM310 does not have to supply current to the periphery.

The LS14500 battery has a nominal capacity of 2600mAh[1] and provides a voltage of 3.6V. Using current consumption data from the ATMega328 and the AMG8831, an estimate of the operating time before the battery has to be changed can be calculated. The ATMega328 consumes $\mathrm{I_{AVR}}$=1.7mA of current while active, $\mathrm{I_{AVRPD2}}$=4.6μA in power-down mode with watchdog timer enabled and $\mathrm{I_{AVRPD}}$=0.1μA in power-down mode with watchdog timer disabled (making it rely on an external interrupt to wake up) [Atm16, p. 367]. The AMG8831 consumes $\mathrm{I_{AMG}}$=4.5mA in normal mode and $\mathrm{I_{AMGPD}}$=0.2mA in sleep mode. The time the ATMega328 needs to read and process the values from the AMG8831 is $\mathrm{T_{PROC}}$=111ms. In experiments it has been determined that a time $\mathrm{T_{STAB}}$=4000ms is sufficient for the stabilization of the pixel readings of the AMG8831. In a scenario where the STM300 wakes up the ATMega328 from power-down mode every 100s and the ATMMega328 performs a measurement, the combined continuous current consumption of the ATMega328 and the AMG8831 is:

$$I_{CONT} \approx \frac{(I_{AMG}+I_{AVRPD2})*T_{STAB}+(I_{AMG}+I_{AVR})*T_{PROC}}{100s} + I_{AVRPD} + I_{AMGPD} = 0.387mA$$

Using the 2600mAh lithium battery, a theoretical runtime approximation can be calculated:

$$T_{RUNTIME} \approx \frac{2600mAh}{I_{CONT}} = 280 \text{ days}$$

A significant improvement of runtime could be achieved by inserting a switch to enable the ATMEGA328 to completely shut off the power to the AMG8831 during sleep mode. The AMG8831 consumes 0.2mA of current during sleep mode, more than half of the 0.387mA continuous current. If the AMG8831 is completely cut off from the supply voltage while the ATMega328 is in sleep mode, the approximate continuous current is:

$$I_{CONT} \approx \frac{(I_{AMG}+I_{AVRPD2})*T_{STAB}+(I_{AMG}+I_{AVR})*T_{PROC}}{100s} + I_{AVRPD} = 0.187mA$$

Using the same 2600mAh battery, the theoretical runtime would therefore be:

---

[1] at 2mA +20°C 2.0V cut-off

$$T_{RUNTIME} \approx \frac{2600mAh}{I_{CONT}} = 579 \text{ days (about 1 year and 7 months)}$$

4.1.2 Suggested module design

A suggested design for an infrared sensor array module can be seen in Figure 4.1. As opposed to the prototype case, the possibility to rotate the sensor module is suggested as it would create more flexible mounting options.

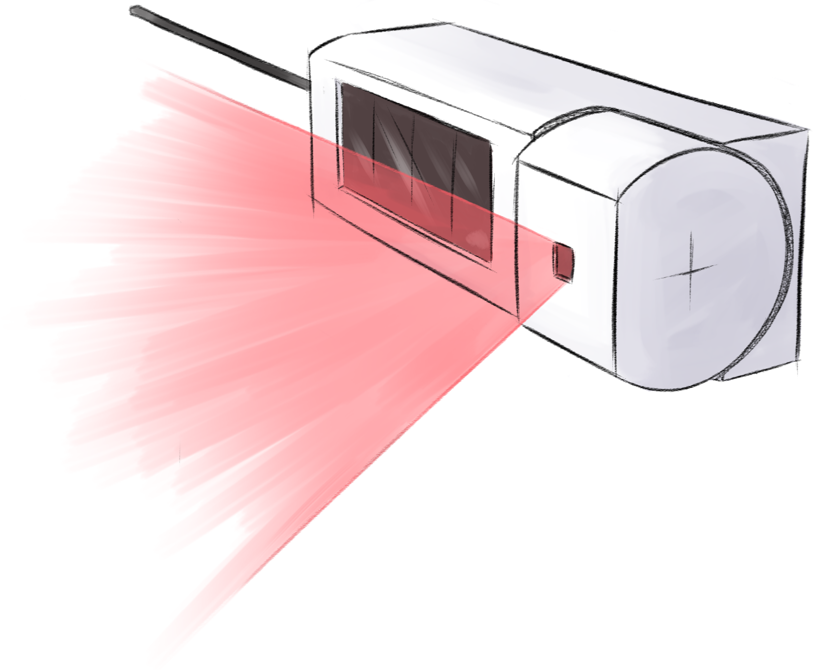


Figure 4.1: Concept for an infrared presence detection module

4.2 Current detection

A circuit was developed in order to detect current flowing through a power cable. This was done by measuring the current generated by an inductor that was placed orthogonally onto a power cable. When the a significant current is flowing through the power cable, the generated electro-magnetic field is picked up by the inductor. Due to the phase and neutral wires mostly canceling each others' electro-magnetic fields, high amplification was required to generate an output signal of usable range.
The circuit was able to detect loads of 15W and higher accurately. While the circuit was designed to detect currents, the readings are accurate enough to draw conclusions

about the current state that the measured device is in (e.g. when a device has different power modes). A test was done using the sensor and measuring the current in the power cable of a washing machine (see 3.2.5), producing promising results and opening up the possibility of using this sensor for more specific applications than simple ON/OFF detection of devices.
The final prototype consists of an EnOcean STM310 module with custom firmware which performs measurements and transmits the measured sensor output at a specified interval. The position of the inductor on the power cable is critical. Ideally the inductor should be placed 90°from the two positions that provide the lowest output reading. (see section 3.2.2) A possible solution to avoid placing the sensor in an unwanted position is using two inductors and place them 90°from each other around the inductor, making the largest possible angle between the optimal measuring position and one of the inductors 45°.

4.2.1 Power consumption

Measuring one period of 50Hz mains current on a power cable takes $\mathrm{T_{MEAS}}$=20ms. The operational amplifier typically consumes a supply current of $\mathrm{I_Q}$=20μA [Mic09]. The charge needed to transmit a telegram is $\mathrm{Q_{TRANS}}$=100μC and the time needed is $\mathrm{T_{TRANS}}$=30ms [EnO12b, p. 20]. The current that the STM310 consumes is $\mathrm{I_{STM}}$=3.7mA in CPU mode and $\mathrm{I_{STMSLEEP}}$=0.20μA in sleep mode [EnO14a, p. 18] [EnO10]. If one measurement is made and transmitted every 100 seconds then an approximation of the required continuous current is:

$$I_{CONT} \approx \frac{(I_Q+I_{STM})*T_{MEAS}+I_Q*T_{TRANS}+Q_{TRANS}}{100s} + I_{SLEEP} = 1.95\mu A$$

For reference, the STM310 module using default firmware with the same wake-up and transmit cycles consumes a continuous current of 1.6μA [EnO10]. In case the module does not have access to light and requires a battery, the theoretical runtime on a 225mAh CR2032 cell (not taking into account self-discharge) would be approximately :

$$T_{RUNTIME} \approx \frac{225mAh}{I_{CONT}} = 4808 \text{ days (about 13 years)}$$

4.2.2 Suggested module design

Figure 4.2 shows a suggested design for a finished module. The case has to be designed in a way that it can be attached to a wide range of power cables which can vary in thickness. The suggested way of doing this is either using screws or strong clamps to hold the power cable in place. It is important that the sensor is completely fixed to the power cable, as any movement would change the measurements due to the readings largely depending on the location of the sensor. Due to the few components needed, a very compact PCB can be designed which makes the final module barely bigger than the STM310 itself.

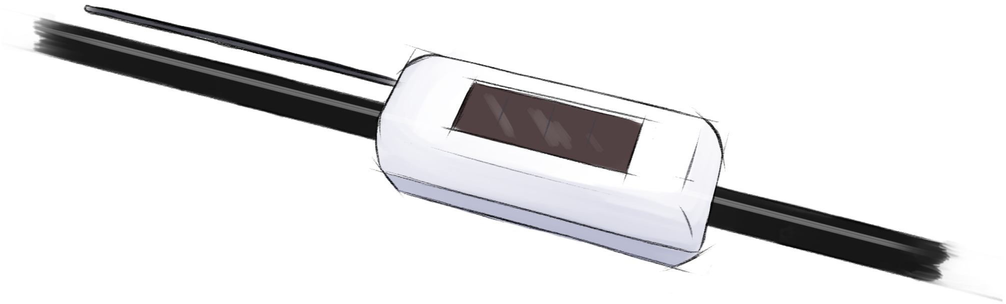


Figure 4.2: Concept for a non-invasive current detection module

4.3 Water flow detection

Detection of water flow was achieved by connecting temperature sensors directly to water pipes using thermal paste. The sensors responded quickly to changes in the surface temperature of the pipe, which correlate with water flow. The requirement for this method of water flow detection to work is a significant difference (±5°C) in water temperature relative to the surrounding air temperature.

The water flow detection module can be used in any place where pipes are accessible, for example:

- Sink
- Shower
- Toilet
- Dishwasher
- Washing machine

When not in use, the pipe had the temperature T_{IDLE} which was typically within ±2°C of the surrounding air temperature. In experiments, the response to water flowing through pipes was near instant. After turning on hot or cold water, the maximum temperature T_{PEAK} was usually measured 2 minutes later. After T_{PEAK} was reached, the surface temperature of the pipe usually approached T_{IDLE} to under 20% of the difference between T_{PEAK} and T_{IDLE} within 20-35 minutes depending on the type of pipe.

The final prototype module consists of an EnOcean STM310 running a firmware based on the I^2C reference implementation [EnO11d], which reads and transmits the temperature data from one or two temperature sensors at an interval of 30 seconds if significant change in temperature is measured.

4.3.1 Power consumption

The conversion and and I^2C transmission time of the TMP75B is approximately T_{MEAS} = 30ms. The quiescent current I_{Q} = 6.5μA when the CR1 and CR0 registers set to 4 conversions per second [Tex14, p. 5]. The charge needed to transmit a telegram is Q_{TRANS}=100μC and the time needed is T_{TRANS}=30ms [EnO12b, p. 20]. The current that the STM310 consumes is I_{STM}=3.7mA in CPU mode and I_{STMSLEEP}=0.20μA in sleep mode [EnO14a, p. 18] [EnO10]. When using two TMP75B modules and performing measurement and transmission every 100 seconds, the continuous current consumption is approximately:

$$I_{CONT} \approx \frac{(2*I_Q+I_{STM})*T_{MEAS}+2*I_Q*T_{TRANS}+Q_{TRANS}}{100s} + I_{SLEEP} = 2.31\mu A$$

In case the module does not have access to light and requires a battery, the theoretical runtime on a 225mAh CR2032 cell (not taking into account self-discharge) would be approximately :

$$T_{RUNTIME} \approx \frac{225mAh}{I_{CONT}} = 4040 \text{ days (about 11 years)}$$

4.3.2 Suggested module design

A suggested design for a finished module can be seen in Figure 4.3 with a base module containing the STM310 and up to 2 connected water sensor modules.

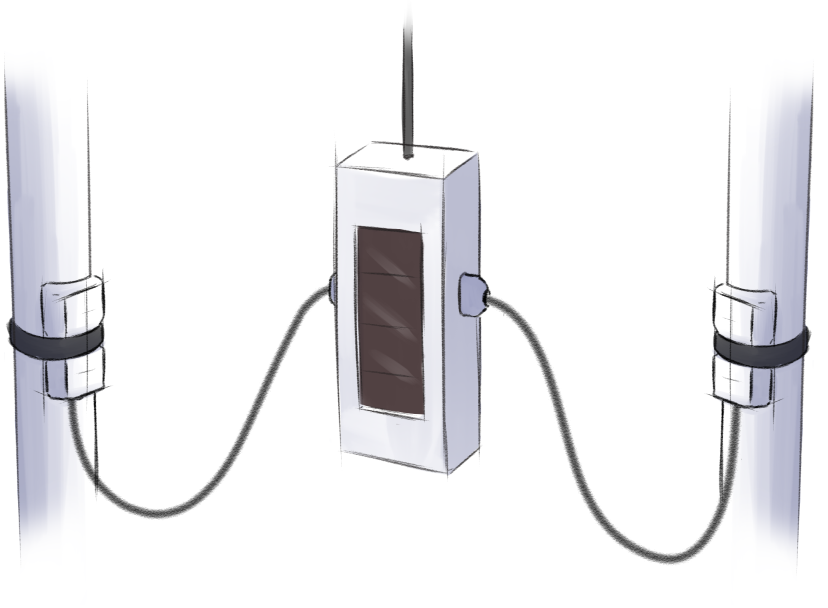


Figure 4.3: Concept for a water flow detection module

CHAPTER 5

Conclusion

Wireless energy harvesting sensor nodes provide a practical way to monitor specific points of interest in the home environment of older and disabled people. These data can be used to detect deviations in daily activities and possibly hazardous situations. By using a monitoring system using these nodes, the need for monitoring tasks by care-workers and/or relatives can be decreased and enable the person in need of assistance to live a more independent life.

In this paper, prototypes of wireless sensor nodes were developed for the purposes of presence detection, monitoring of water flow and monitoring of possibly hazardous electric devices. The sensor modules were developed with focus on low energy usage, easy installation/removal and non-obtrusiveness. The sensor module prototypes have been successfully generating sufficiently accurate data for considering them for the use in an AAL system. The used EnOcean modules offer a compelling versatility due to not needing an external power source[1]. While relying on energy harvesting provides several advantages over other systems which are relying on batteries or line power, the restrictions have to be considered. The intervals between data transmissions are high, the data rates are low and there is a possibility of modules running out of energy because of lack of light.

The grid infrared sensor module provides excellent presence detection ability and has been shown to be able to detect presence in bed with very good accuracy. The module can be easily adapted for other use in other locations in the home for presence detection. Other usage scenarios such as monitoring cooking plates are also possible.

The water flow detection module was realized using TMP75B I^2C sensors which were attached to water pipes. The sensors measured the difference in temperature that warm or cold water flowing through the pipes causes. Using knowledge about the ambient temperature, it is possible to accurately measure whether water is flowing through pipes.

The current detection module operates by measuring the amplified signal generated by an

---

[1] Unless the external circuit requires more power than possible with energy harvesting alone

inductor which is placed orthogonally on a power cable with current flow. Experiments were showing that the exact placement on the conductor is critical. With the right placement it is possible to determine with great accuracy whether a 15W+ load is turned on or off.

In conclusion, the advances in technology that make it possible to create cheap, flexible sensor modules provides a hopeful vision about a future where homes can be easily fitted with assisting technology, providing a higher level of safety without compromising privacy.

APPENDIX A

Appendices

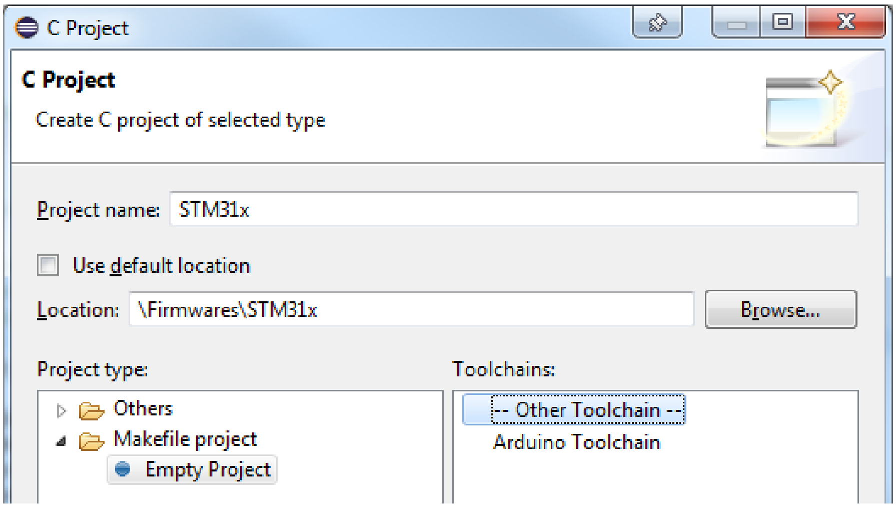
A.1 Setting up Eclipse for EO3000I firmware development

This section describes how to set up Eclipse for editing, compiling and uploading STM3xx firmwares using the Keil C51 compiler. This allows the benefits of Eclipse (e.g. resolving references and auto-completion) to be used while developing STM3xx firmware.
Prerequisites:

- Eclipse
- CDT Extension
- Keil C51 IDE
- Existing Keil Project

A.1.1 Creating the Project

Create a new empty C project in Eclipse and choose the existing project folder as location.



C Project

Create C project of selected type

Project name: STM31x

☐ Use default location

Location: \Firmwares\STM31x

Project type:

- Others
- Makefile project
  - Empty Project

Toolchains:

- -- Other Toolchain --
- Arduino Toolchain

Figure A.1: C Project wizard

A.1.2 Using the uVision Builder

In order to use the uVision builder to build the project, a simple .bat file has to be created. While it would be possible to do all the build steps manually, the easiest option is to simply point the uVision builder and the project file and execute the build command. It has to be kept in mind that any new source files have to be added within the uVision IDE.

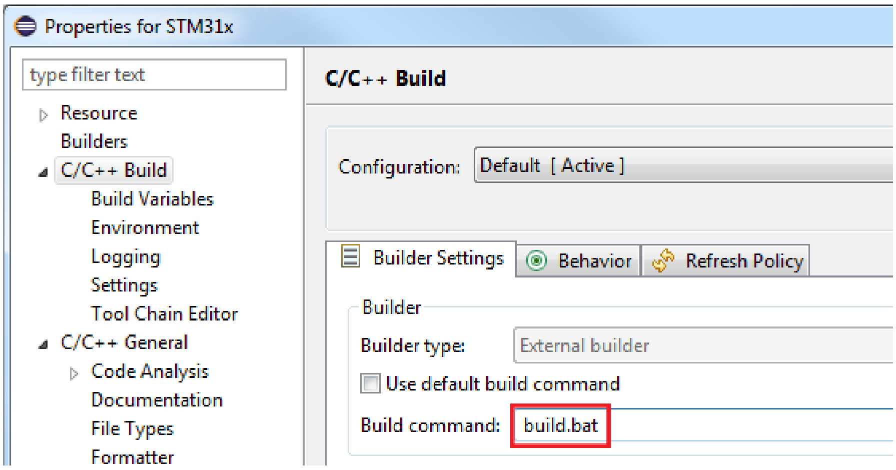
```
@echo off
C:\KEIL\UV4\UV4.exe -b -j0 STM3xx.uvproj -o "output.log"
type output.log
```

Listing A.1: build.bat - executes the uVision builder

For building and additional uploading, the .bat file can include the EOPX tool to upload the compiled program.

```
@echo off
C:\KEIL\UV4\UV4.exe -b -j0 STM3xx.uvproj -o "output.log"
type output.log
echo --------------------------------------------------------------
echo Uploading program:
echo --------------------------------------------------------------
"C:\Program Files\EnOcean\DolphinStudio\eopx.exe" -port EOVR6XMPA -write -fprg "C:\Projects\STM3xx\STM3xx_868.hex" -fcfg "C:\Projects\STM3xx\STM3xx_868_cfg.hex"
```

Listing A.2: build_upload.bat - executes the uVision builder and uploads the firmware



Configuration: Default [Active]

Builder type: External builder

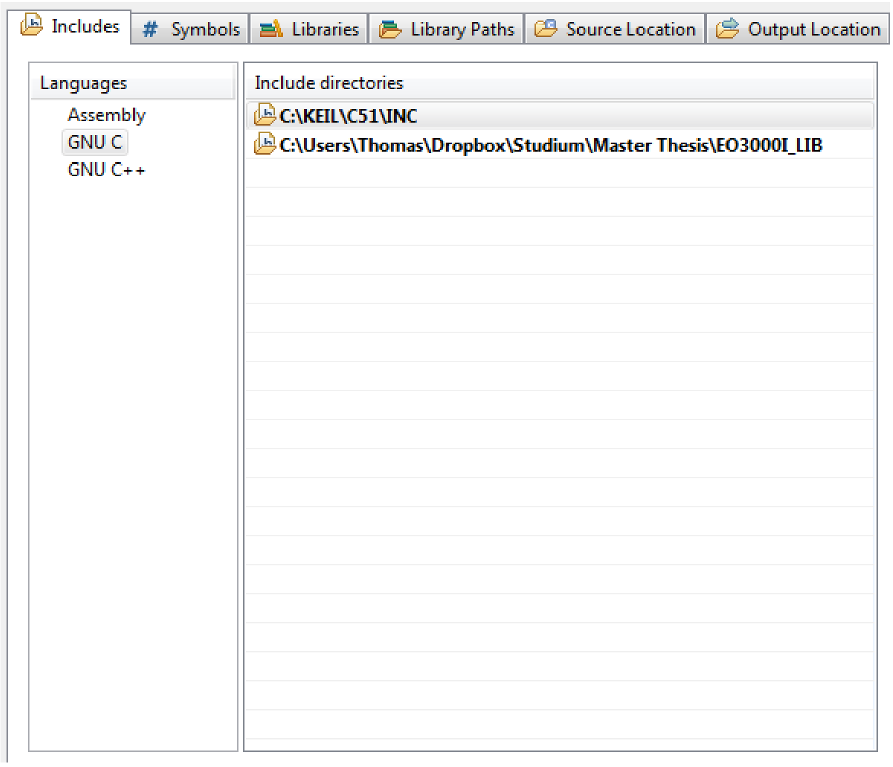
☐ Use default build command

Build command: build.bat

Figure A.2: Adding custom build configurations

A.1.3 Including the C51 and EO3000I API libraries

The C51 library as well at the EO3000I API need to be included. The paths for the C51 include folder (usually C:\KEIL\C51\INC) and the EO3000I API folder have to be added under Properties -> C/C++ General -> Paths and Symbols -> Includes -> GNU C.



C:\KEIL\C51\INC

C:\Users\Thomas\Dropbox\Studium\Master Thesis\EO3000I_LIB

Figure A.3: Including C51 and EO3000I libraries

A.1.4 Adding symbols

Since the Keil C51 compiler uses a few extra keywords, these need to be added as well. The EO3000I API also contains some typedefs that can be added to be recognized in Eclipse. This is important to prevent the code editor from showing errors because of unknown types.

The directives _at_(A), xdata, data, idata and code are specific to the Keil C51 compiler, the other ones are typedefs from the EO3000I API. Important: the _at_ directive only works without a whitespace and with the brackets, so for example the line

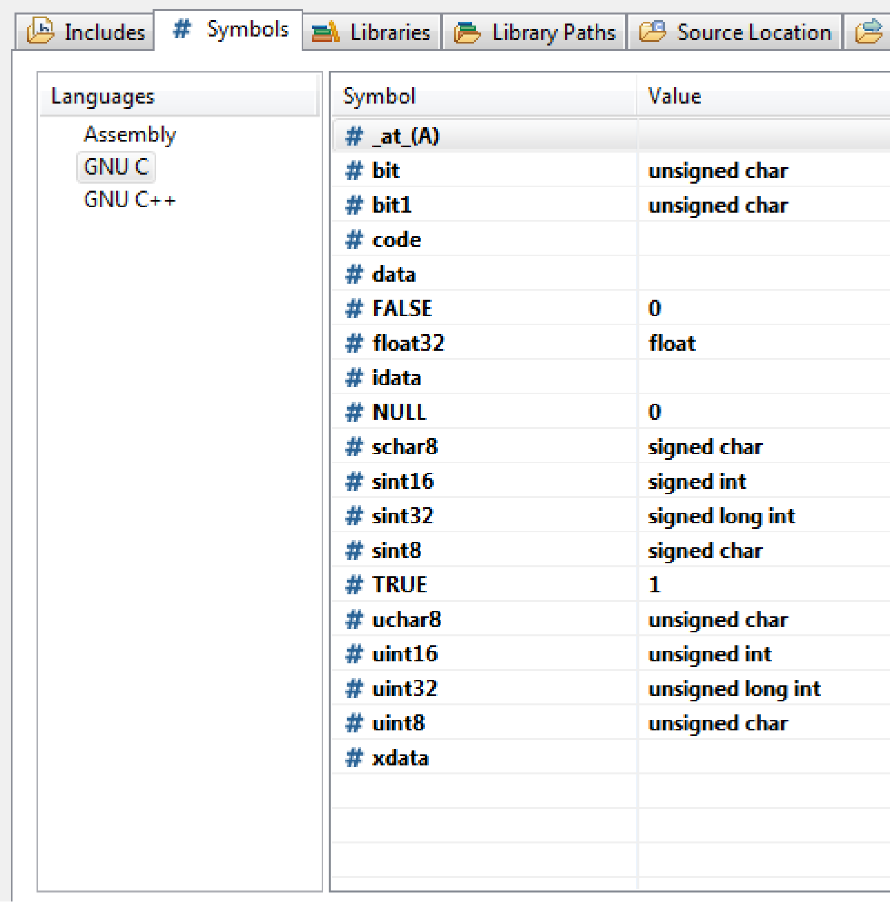
```
        CONFIG_AREA    xdata    unigcfg _at_ CFG_ADDR;
```

has to be modified to

```
        CONFIG_AREA    xdata    unigcfg _at_(CFG_ADDR);
```

A.1.5 Handling compiler messages

For handling the error messages and integrating them into Eclipse, a custom Error parser is necessary. Under Window -> Preferences -> C/C++ Build -> Settings and add a new Error Parser, these rules have to be added:

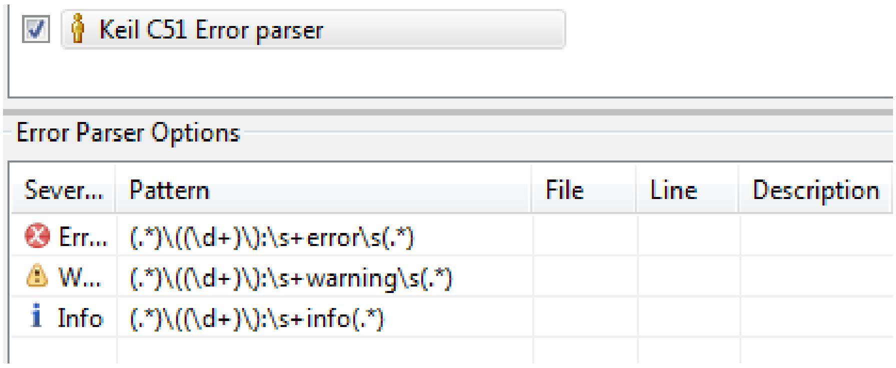


Symbol	Value
# _at_(A)	
# bit	unsigned char
# bit1	unsigned char
# code	
# data	
# FALSE	0
# float32	float
# idata	
# NULL	0
# schar8	signed char
# sint16	signed int
# sint32	signed long int
# sint8	signed char
# TRUE	1
# uchar8	unsigned char
# uint16	unsigned int
# uint32	unsigned long int
# uint8	unsigned char
# xdata	

Figure A.4: Adding keywords and typedefs

```
                Error          (.*)\((\d+)\):\s+error\s(.*)
                Warning        (.*)\((\d+)\):\s+warning\s(.*)
                Info           (.*)\((\d+)\):\s+info(.*)
```

Ensure that the parser is active for the specific project under Properties -> C/C++ Build -> Settings.



☑ Keil C51 Error parser

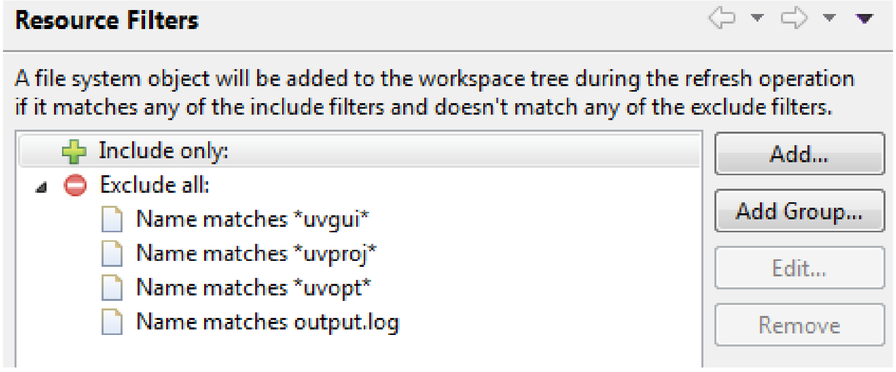
Error Parser Options

Sever...	Pattern	File	Line	Description
Err...	(.*)\((\d+)\):\s+error\s(.*)			
W...	(.*)\((\d+)\):\s+warning\s(.*)			
Info	(.*)\((\d+)\):\s+info(.*)			

Figure A.5: Creating a custom Error parser

A.1.6 Hiding unnecessary files

Since the Project Explorer displays all the project files from the uVision IDE, these can be simply hidden with a few rules. Under Properties -> Resource -> Resource Filters add these rules under *Exclude all*:



Resource Filters

A file system object will be added to the workspace tree during the refresh operation if it matches any of the include filters and doesn't match any of the exclude filters.

- Include only:
- Exclude all:
  - Name matches *uvgui*
  - Name matches *uvproj*
  - Name matches *uvopt*
  - Name matches output.log

Figure A.6: Excluding uVision project files in the Project Explorer

Using these steps, the Eclipse IDE should now provide proper code highlighting, error display, auto-completion, referencing of types to the C51 and EO3000I header files as well as build and upload commands.

Bibliography

[Ass16] AAL Association. About. http://www.aal-europe.eu, 2016.

[Atm16] Atmel Corporation. *ATmega328/P Data sheet*, June 2016.

[CCTS08] K A Cook-Chennault, N Thambi, and A M Sastry. Powering mems portable devices: a review of non-regenerative and regenerative power supply systems with special emphasis on piezoelectric energy harvesting systems. *Smart Materials and Structures*, 17(4):043001, 2008.

[Dut15] Sandeep Dutta. Sdcc 3.5.1 user guide, 2015.

[EFH09] Birgid Eberhardt, Uwe Fachinger, and Klaus-Dirk Henke. Better health and ambient assisted living (aal) from a global, regional and local economic perspective. Discussion Papers 2009/3, Technische Universität Berlin, School of Economics and Management, 2009.

[EnO] EnOcean GmbH. *EO3000I API 2.6.1.0 Manual*.

[EnO09] EnOcean GmbH. *EnOcean Developer Kit EDK300(C) User Manual*, August 2009. Version 1.0.

[EnO10] EnOcean GmbH. *STM 300 THERMO OR BATTERY POWERED - Power Supply Alternatives to Solar Panel*. Application Note 209. February 2010.

[EnO11a] EnOcean GmbH. *Developing custom firmware for STM 31X and STM 330*. Application Note 505. June 2011.

[EnO11b] EnOcean GmbH. *Dolphin In-Circuit programming - Updating Firmware in the field*. Application Note 502. June 2011.

[EnO11c] EnOcean GmbH. *ECS 300/310 Solar Panel - Design Considerations*. Application Note 207. February 2011.

[EnO11d] EnOcean GmbH. *I^2C DOLPHIN Interface - Connecting Dolphin based modules to sensorics through I^2C*. Application Note 508. September 2011.

[EnO12a] EnOcean GmbH. *EnOcean Radio Protocol*, October 2012. Version 1.0.

[EnO12b] EnOcean GmbH. *Scavenger Transmitter Module STM 31x / STM 31xC User Manual*, October 2012. Version 1.04.

[EnO13a] EnOcean GmbH. *Scavenger Transmitter Module STM 330 / STM 331 / STM 330C / STM 332U / STM 333U User Manual*, June 2013. Version 1.15.

[EnO13b] EnOcean GmbH. *Solar cells ECS 300 and ECS 310 Data Sheet*, October 2013.

[EnO13c] EnOcean GmbH. *STM 3xy wireless sensor transmitter module family.* June 2013.

[EnO14a] EnOcean GmbH. *Dolphin Core Description V1.1.* July 2014.

[EnO14b] EnOcean GmbH. *EnOcean Serial Protocol 3 (ESP3)*, July 2014. Version 1.27.

[EnO14c] EnOcean GmbH. *USB 300 / USB 300C / USB 300U and USB 400J Data Sheet*, September 2014.

[EnO15] EnOcean GmbH. *ECO 200 Energy Harvester Data Sheet*, August 2015.

[Esp15] Espressif Systems. *ESP8266 Data sheet*, June 2015. Version 4.3.

[Eur] Eurostat. Population structure and ageing. *Eurostat yearbook*, June 2016.

[Ext02] Extech Instruments Corporation. *Model DA30 AC Current Detector User's Guide*, March 2002. Rev 1.3.

[FLPM$^+$13] Francisco J. Fernandez-Luque, David Pérez, Félix Martínez, Ginés Domènech, Isabel Navarrete, Juan Zapata, and Ramón Ruiz. An energy efficient middleware for an ad-hoc aal wireless sensor network. *Ad Hoc Netw.*, 11(3):907–925, May 2013.

[KKK11] Heung Soo Kim, Joo-Hyong Kim, and Jaehwan Kim. A review of piezoelectric energy harvesting based on vibration. *International Journal of Precision Engineering and Manufacturing*, 12(6):1129–1141, 2011.

[LFR$^+$09] A. Lombardi, M. Ferri, G. Rescio, M. Grassi, and P. Malcovati. Wearable wireless accelerometer with embedded fall-detection logic for multi-sensor ambient assisted living applications. In *2009 IEEE Sensors*, pages 1967–1970, Oct 2009.

[M$^+$07] Montenegro et al. Transmission of ipv6 packets over ieee 802.15.4 networks. *RFC 4944*, 09 2007.

[Mel12] Melexis NV. *MLX90621 16x4 pixel thermal imager product flyer*, 2012.

[Mic09] Microchip Technology Inc. *MCP6231/1R/1U/2/4 20 μA, 300 kHz Rail-to-Rail Op Amp Data Sheet*, 2009.

[MM05] Loreto Mateu and Francesc Moll. Review of energy harvesting techniques and applications for microelectronics (keynote address). volume 5837, pages 359–373, 2005.

[MSM16] Auday A. H. Mohamad Mahmoud Shuker Mahmoud. A study of efficient power consumption wireless communication techniques/ modules for internet of things (iot) applications. *Advances in Internet of Things*, 6:19–29, 2016.

[Org08] ZigBee Standards Organization. Zigbee specification. 2008.

[Pan11] Panasonic Corporation. *Specifications for Infrared Array Sensor AMG88XX*, August 2011.

[Pan12] Panasonic Corporation. *AMG88XX High Precision Infrared Array Sensor based on Advanced MEMS Technology*, January 2012.

[PML13] M. Piñuela, P. D. Mitcheson, and S. Lucyszyn. Ambient rf energy harvesting in urban and semi-urban environments. *IEEE Transactions on Microwave Theory and Techniques*, 61(7):2715–2726, July 2013.

[RM13] P. Rashidi and A. Mihailidis. A survey on ambient-assisted living tools for older adults. *IEEE Journal of Biomedical and Health Informatics*, 17(3):579–590, May 2013.

[rsh17] Ultrasonic flow meters, February 2017.

[SAF09] SAFT. *LS 14500 Primary lithium battery Data sheet*, September 2009.

[Sam15] Samsung Electronics Co., Ltd. *ARTIK 1 Data Sheet (Advanced)*, April 2015. Rev. FS 0.2.

[Sch16] Marco Schwartz. How to run your esp8266 for years on a battery. *https://openhomeautomation.net/esp8266-battery/*, 2016.

[Soc15] IEEE Computer Society. Ieee standard for low-rate wireless networks. *IEEE Standards Association*, Std 802.15.4$^{\mathrm{TM}}$-2015, 2015.

[Tex12] Texas Instruments Incorporated. *4-bit bidirectional voltage-level translator with automated direction sensing and* $\pm 15kV$ *ESD protection*, May 2012.

[Tex14] Texas Instruments Incorporated. *1.8-V Digital Temperature Sensor with Two-Wire Interface and Alert*, October 2014.

[TRM09] E. O. Torres and G. A. Rincon-Mora. Electrostatic energy-harvesting and battery-charging cmos system prototype. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 56(9):1938–1948, Sept 2009.

[ZS09] Carsten Bormann Zach Shelby. *6LoWPAN: The wireless embedded Internet.* Wiley, 11 2009.