Universitat Pompeu Fabra

Bachelor Thesis

Open Sensor Network

Author:
Alejandro Andreu Isábal

Supervisor:
Dr. Jaume Barceló

A thesis submitted in fulfilment of the requirements for the degree of Graduate in Telematic Engineering

in the

Research Group Name
Department or School Name

April 2013

Declaration of Authorship

I, Alejandro Andreu Isábal, declare that this thesis titled, 'Open Sensor Network' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ____

Date: ____

“Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.”

Dave Barry

UNIVERSITAT POMPEU FABRA

Abstract

Escola Superior Politècnica
Department or School Name

Graduate in Telematic Engineering

Open Sensor Network

by Alejandro Andreu Isábal

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too. . .

Acknowledgements

The acknowledgements and the people to thank go here, don't forget to include your project advisor…

Contents

List of Figures

List of Tables

Abbreviations

ADC **A**nalog to **D**igital **C**onverter

LoS **L**ine **o**f **S**ight

Physical Constants

Speed of Light $c = 2.997\ 924\ 58 \times 10^{8}\ \mathrm{ms}^{-\mathrm{S}}$ (exact)

Symbols

a	distance	m
P	power	W (Js^{-1})
ω	angular frequency	$rads^{-1}$

For/Dedicated to/To my…

Chapter 1

Chapter Title Here

1.1 Welcome and Thank You

Welcome to this LaTeX Thesis Template, a beautiful and easy to use template for writing a thesis using the LaTeX typesetting system.

If you are writing a thesis (or will be in the future) and its subject is technical or mathematical (though it doesn't have to be), then creating it in LaTeX is highly recommended as a way to make sure you can just get down to the essential writing without having to worry over formatting or wasting time arguing with your word processor.

LaTeX is easily able to professionally typeset documents that run to hundreds or thousands of pages long. With simple mark-up commands, it automatically sets out the table of contents, margins, page headers and footers and keeps the formatting consistent and beautiful. One of its main strengths is the way it can easily typeset mathematics, even *heavy* mathematics. Even if those equations are the most horribly twisted and most difficult mathematical problems that can only be solved on a super-computer, you can at least count on LaTeX to make them look stunning.

1.2 Learning LaTeX

LaTeX is not a WYSIWYG (What You See is What You Get) program, unlike word processors such as Microsoft Word or Apple's Pages. Instead, a document written for LaTeX is actually a simple, plain text file that contains *no formatting*. You tell LaTeX how you want the formatting in the finished document by writing in simple commands amongst the text, for example, if I want to use *italic text for emphasis*, I write the

'`\textit{}`' command and put the text I want in italics in between the curly braces. This means that LaTeX is a "mark-up" language, very much like HTML.

1.2.1 A (not so short) Introduction to LaTeX

If you are new to LaTeX, there is a very good eBook – freely available online as a PDF file – called, "The Not So Short Introduction to LaTeX". The book's title is typically shortened to just "lshort". You can download the latest version (as it is occasionally updated) from here:
http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf

It is also available in several other languages. Find yours from the list on this page:
http://www.ctan.org/tex-archive/info/lshort/

It is recommended to take a little time out to learn how to use LaTeX by creating several, small 'test' documents. Making the effort now means you're not stuck learning the system when what you *really* need to be doing is writing your thesis.

1.2.2 A Short Math Guide for LaTeX

If you are writing a technical or mathematical thesis, then you may want to read the document by the AMS (American Mathematical Society) called, "A Short Math Guide for LaTeX". It can be found online here:
http://www.ams.org/tex/amslatex.html
under the "Additional Documentation" section towards the bottom of the page.

1.2.3 Common LaTeX Math Symbols

There are a multitude of mathematical symbols available for LaTeX and it would take a great effort to learn the commands for them all. The most common ones you are likely to use are shown on this page:
http://www.sunilpatel.co.uk/latexsymbols.html

You can use this page as a reference or crib sheet, the symbols are rendered as large, high quality images so you can quickly find the LaTeX command for the symbol you need.

1.2.4 LaTeX on a Mac

The LaTeX package is available for many systems including Windows, Linux and Mac OS X. The package for OS X is called MacTeX and it contains all the applications you need – bundled together and pre-customised – for a fully working LaTeX environment and workflow.

MacTeX includes a dedicated LaTeX IDE (Integrated Development Environment) called "TeXShop" for writing your '`.tex`' files and "BibDesk": a program to manage your references and create your bibliography section just as easily as managing songs and creating playlists in iTunes.

1.3 Getting Started with this Template

If you are familiar with LaTeX, then you can familiarise yourself with the contents of the Zip file and the directory structure and then place your own information into the '`Thesis.cls`' file. Section 1.5 on page 6 tells you how to do this. Make sure you read section 1.7 about thesis conventions to get the most out of this template and then get started with the '`Thesis.tex`' file straightaway.

If you are new to LaTeX it is recommended that you carry on reading through the rest of the information in this document.

1.3.1 About this Template

This LaTeX Thesis Template is originally based and created around a LaTeX style file created by Steve R. Gunn from the University of Southampton (UK), department of Electronics and Computer Science. You can find his original thesis style file at his site, here:
`http://www.ecs.soton.ac.uk/~srg/softwaretools/document/templates/`

My thesis originally used the '`ecsthesis.cls`' from his list of styles. However, I knew LaTeX could still format better. To get the look I wanted, I modified his style and also created a skeleton framework and folder structure to place the thesis files in.

This Thesis Template consists of that modified style, the framework and the folder structure. All the work that has gone into the preparation and groundwork means that all you have to bother about is the writing.

Before you begin using this template you should ensure that its style complies with the thesis style guidelines imposed by your institution. In most cases this template style and layout will be suitable. If it is not, it may only require a small change to bring the template in line with your institution's recommendations.

1.4 What this Template Includes

1.4.1 Folders

This template comes as a single Zip file that expands out to many files and folders. The folder names are mostly self-explanatory:

Appendices – this is the folder where you put the appendices. Each appendix should go into its own separate '`.tex`' file. A template is included in the directory.

Chapters – this is the folder where you put the thesis chapters. A thesis usually has about seven chapters, though there is no hard rule on this. Each chapter should go in its own separate '`.tex`' file and they usually are split as:

- Chapter 1: Introduction to the thesis topic
- Chapter 2: Background information and theory
- Chapter 3: (Laboratory) experimental setup
- Chapter 4: Details of experiment 1
- Chapter 5: Details of experiment 2
- Chapter 6: Discussion of the experimental results
- Chapter 7: Conclusion and future directions

This chapter layout is specialised for the experimental sciences.

Figures – this folder contains all figures for the thesis. These are the final images that will go into the thesis document.

Primitives – this is the folder that contains scraps, particularly because one final image in the 'Figures' folder may be made from many separate images and photos, these source images go here. This keeps the intermediate files separate from the final thesis figures.

1.4.2 Files

Included are also several files, most of them are plain text and you can see their contents in a text editor. Luckily, many of them are auxiliary files created by LaTeX or BibTeX and which you don't need to bother about:

Bibliography.bib – this is an important file that contains all the bibliographic information and references that you will be citing in the thesis for use with BibTeX. You can write it manually, but there are reference manager programs available that will create and manage it for you. Bibliographies in LaTeX are a large subject and you may need to read about BibTeX before starting with this.

Thesis.cls – this is an important file. It is the style file that tells LaTeX how to format the thesis. You will also need to open this file in a text editor and fill in your own information (such as name, department, institution). Luckily, this is not too difficult and is explained in section 1.5 on page 6.

Thesis.pdf – this is your beautifully typeset thesis (in the PDF file format) created by LaTeX.

Thesis.tex – this is an important file. This is the file that you tell LaTeX to compile to produce your thesis as a PDF file. It contains the framework and constructs that tell LaTeX how to layout the thesis. It is heavily commented so you can read exactly what each line of code does and why it is there. After you put your own information into the '`Thesis.cls`' file, go to this file and begin filling it in – you have now started your thesis!

vector.sty – this is a LaTeX package, it tells LaTeX how to typeset mathematical vectors. Using this package is very easy and you can read the documentation on the site (you just need to look at the '`vector.pdf`' file):
`http://www.ctan.org/tex-archive/macros/latex/contrib/vector/`

lstpatch.sty – this is a LaTeX package required by this LaTeX template and is included as not all TeX distributions have it installed by default. You do not need to modify this file.

Files that are *not* included, but are created by LaTeX as auxiliary files include:

Thesis.aux – this is an auxiliary file generated by LaTeX, if it is deleted LaTeX simply regenerates it when you run the main '`.tex`' file.

Thesis.bbl – this is an auxiliary file generated by BibTeX, if it is deleted, BibTeX simply regenerates it when you run the main tex file. Whereas the '`.bib`' file contains all the references you have, this '`.bbl`' file contains the references you have actually cited in the thesis and is used to build the bibliography section of the thesis.

Thesis.blg – this is an auxiliary file generated by BibTeX, if it is deleted BibTeX simply regenerates it when you run the main '`.tex`' file.

Thesis.lof – this is an auxiliary file generated by LaTeX, if it is deleted LaTeX simply regenerates it when you run the main '`.tex`' file. It tells LaTeX how to build the 'List of Figures' section.

Thesis.log – this is an auxiliary file generated by LaTeX, if it is deleted LaTeX simply regenerates it when you run the main '`.tex`' file. It contains messages from LaTeX, if you receive errors and warnings from LaTeX, they will be in this '`.log`' file.

Thesis.lot – this is an auxiliary file generated by LaTeX, if it is deleted LaTeX simply regenerates it when you run the main '`.tex`' file. It tells LaTeX how to build the 'List of Tables' section.

Thesis.out – this is an auxiliary file generated by LaTeX, if it is deleted LaTeX simply regenerates it when you run the main '`.tex`' file.

So from this long list, only the files with the '`.sty`', '`.bib`', '`.cls`' and '`.tex`' extensions are the most important ones. The other auxiliary files can be ignored or deleted as LaTeX and BibTeX will regenerate them.

1.5 Filling in the '`Thesis.cls`' File

You will need to personalise the thesis template and make it your own by filling in your own information. This is done by editing the '`Thesis.cls`' file in a text editor.

Open the file and scroll down, past all the '`\newcommand`...' items until you see the entries for '`University Name`', '`Department Name`', etc....

Fill out the information about your group and institution and ensure you keep to block capitals where it asks you to. You can also insert web links, if you do, make sure you use the full URL, including the '`http://`' for this.

The last item you should need to fill in is the Faculty Name (in block capitals). When you have done this, save the file and recompile '`Thesis.tex`'. All the information you filled in should now be in the PDF, complete with web links. You can now begin your thesis proper!

1.6 The '`Thesis.tex`' File Explained

The `Thesis.tex` file contains the structure of the thesis. There are plenty of written comments that explain what pages, sections and formatting the LaTeX code is creating. Initially there seems to be a lot of LaTeX code, but this is all formatting, and it has all been taken care of so you don't have to do it.

Begin by checking that your information on the title page is correct. For the thesis declaration, your institution may insist on something different than the text given. If this is the case, just replace what you see with what is required.

Then comes a page which contains a funny quote. You can put your own, or quote your favourite scientist, author, person, etc... Make sure to put the name of the person who you took the quote from.

Next comes the acknowledgements. On this page, write about all the people who you wish to thank (not forgetting parents, partners and your advisor/supervisor).

The contents pages, list of figures and tables are all taken care of for you and do not need to be manually created or edited. The next set of pages are optional and can be deleted since they are for a more technical thesis: insert a list of abbreviations you have used in the thesis, then a list of the physical constants and numbers you refer to and finally, a list of mathematical symbols used in any formulae. Making the effort to fill these tables means the reader has a one-stop place to refer to instead of searching the internet and references to try and find out what you meant by certain abbreviations or symbols.

The list of symbols is split into the Roman and Greek alphabets. Whereas the abbreviations and symbols ought to be listed in alphabetical order (and this is *not* done automatically for you) the list of physical constants should be grouped into similar themes.

The next page contains a one line dedication. Who will you dedicate your thesis to?

Finally, there is the section where the chapters are included. Uncomment the lines (delete the '`%`' character) as you write the chapters. Each chapter should be written in its own file and put into the 'Chapters' folder and named '`Chapter1`', '`Chapter2`, etc... Similarly for the appendices, uncomment the lines as you need them. Each appendix should go into its own file and placed in the 'Appendices' folder.

After the preamble, chapters and appendices finally comes the bibliography. The bibliography style (called '`unsrtnat`') is used for the bibliography and is a fully featured style that will even include links to where the referenced paper can be found online. Do not under estimate how grateful you reader will be to find that a reference to a paper is just a click away. Of course, this relies on you putting the URL information into the BibTeX file in the first place.

1.7 Thesis Features and Conventions

To get the best out of this template, there are a few conventions that you may want to follow.

One of the most important (and most difficult) things to keep track of in such a long document as a thesis is consistency. Using certain conventions and ways of doing things (such as using a Todo list) makes the job easier. Of course, all of these are optional and you can adopt your own method.

1.7.1 Printing Format

This thesis template is designed for single sided printing as most theses are printed and bound this way. This means that the left margin is always wider than the right (for binding). Four out of five people will now judge the margins by eye and think, “I never noticed that before.”.

The headers for the pages contain the page number on the right side (so it is easy to flick through to the page you want) and the chapter name on the left side.

The text is set to 11 point and a line spacing of 1.3. Generally, it is much more readable to have a smaller text size and wider gap between the lines than it is to have a larger text size and smaller gap. Again, you can tune the text size and spacing should you want or need to. The text size can be set in the options for the ‘`\documentclass`’ command at the top of the ‘`Thesis.tex`’ file and the spacing can be changed by setting a different value in the ‘`\setstretch`’ commands (scattered throughout the ‘`Thesis.tex`’ file).

1.7.2 Using US Letter Paper

The paper size used in the template is A4, which is a common – if not standard – size in Europe. If you are using this thesis template elsewhere and particularly in the United States, then you may have to change the A4 paper size to the US Letter size. Unfortunately, this is not as simple as replacing instances of ‘`a4paper`’ with ‘`letterpaper`’.

This is because the final PDF file is created directly from the LaTeX source using a program called '`pdfTeX`' and in certain conditions, paper size commands are ignored and all documents are created with the paper size set to the size stated in the configuration file for pdfTeX (called '`pdftex.cfg`').

What needs to be done is to change the paper size in the configuration file for `pdfTeX` to reflect the letter size. There is an excellent tutorial on how to do this here: `http://www.physics.wm.edu/~norman/latexhints/pdf_papersize.html`

It may be sufficient just to replace the dimensions of the A4 paper size with the US Letter size in the `pdftex.cfg` file. Due to the differences in the paper size, the resulting margins may be different to what you like or require (as it is common for Institutions to dictate certain margin sizes). If this is the case, then the margin sizes can be tweaked by opening up the `Thesis.cls` file and searching for the line beginning with, '`\setmarginsrb`' (not very far down from the top), there you will see the margins specified. Simply change those values to what you need (or what looks good) and save. Now your document should be set up for US Letter paper size with suitable margins.

1.7.3 References

The '`natbib`' package is used to format the bibliography and inserts references such as this one [1]. The options used in the '`Thesis.tex`' file mean that the references are listed in numerical order as they appear in the text. Multiple references are rearranged in numerical order (e.g. [2, 3]) and multiple, sequential references become reformatted to a reference range (e.g. [1–3]). This is done automatically for you. To see how you use references, have a look at the '`Chapter1.tex`' source file. Many reference managers allow you to simply drag the reference into the document as you type.

Scientific references should come *before* the punctuation mark if there is one (such as a comma or period). The same goes for footnotes[1]. You can change this but the most important thing is to keep the convention consistent throughout the thesis. Footnotes themselves should be full, descriptive sentences (beginning with a capital letter and ending with a full stop).

To see how LaTeX typesets the bibliography, have a look at the very end of this document (or just click on the reference number links).

[1] Such as this footnote, here down at the bottom of the page.

1.7.4 Figures

There will hopefully be many figures in your thesis (that should be placed in the 'Figures' folder). The way to insert figures into your thesis is to use a code template like this:

```
\begin{figure}[htbp]
  \centering
    \includegraphics{./Figures/Electron.pdf}
    \rule{35em}{0.5pt}
  \caption[An Electron]{An electron (artist's impression).}
  \label{fig:Electron}
\end{figure}
```

Also look in the source file. Putting this code into the source file produces the picture of the electron that you can see in the figure below.

FIGURE 1.1: An electron (artist's impression).

Sometimes figures don't always appear where you write them in the source. The placement depends on how much space there is on the page for the figure. Sometimes there is not enough room to fit a figure directly where it should go (in relation to the text) and so LaTeX puts it at the top of the next page. Positioning figures is the job of LaTeX and so you should only worry about making them look good!

Figures usually should have labels just in case you need to refer to them (such as in Figure 1.1). The '`\caption`' command contains two parts, the first part, inside the square brackets is the title that will appear in the 'List of Figures', and so should be short. The second part in the curly brackets should contain the longer and more descriptive caption text.

The '`\rule`' command is optional and simply puts an aesthetic horizontal line below the image. If you do this for one image, do it for all of them.

The LaTeX Thesis Template is able to use figures that are either in the PDF or JPEG file format.

1.7.5 Typesetting mathematics

If your thesis is going to contain heavy mathematical content, be sure that LaTeX will make it look beautiful, even though it won't be able to solve the equations for you.

The "Not So Short Introduction to LaTeX" (available here) should tell you everything you need to know for most cases of typesetting mathematics. If you need more information, a much more thorough mathematical guide is available from the AMS called, "A Short Math Guide to LaTeX" and can be downloaded from:
`ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf`

There are many different LaTeX symbols to remember, luckily you can find the most common symbols here. You can use the web page as a quick reference or crib sheet and because the symbols are grouped and rendered as high quality images (each with a downloadable PDF), finding the symbol you need is quick and easy.

You can write an equation, which is automatically given an equation number by LaTeX like this:

```
\begin{equation}
E = mc^{2}
  \label{eqn:Einstein}
\end{equation}
```

This will produce Einstein's famous energy-matter equivalence equation:

$$E = mc^2 \tag{1.1}$$

All equations you write (which are not in the middle of paragraph text) are automatically given equation numbers by LaTeX. If you don't want a particular equation numbered, just put the command, '`\nonumber`' immediately after the equation.

1.8 Sectioning and Subsectioning

You should break your thesis up into nice, bite-sized sections and subsections. LaTeX automatically builds a table of Contents by looking at all the '`\chapter{}`', '`\section{}`' and '`\subsection{}`' commands you write in the source.

The table of Contents should only list the sections to three (3) levels. A '`\chapter{}`' is level one (1). A '`\section{}`' is level two (2) and so a '`\subsection{}`' is level three (3). In your thesis it is likely that you will even use a '`\subsubsection{}`', which is level four (4). Adding all these will create an unnecessarily cluttered table of Contents and so you should use the '`\subsubsection*{}`' command instead (note the asterisk). The asterisk (*) tells LaTeX to omit listing the subsubsection in the Contents, keeping it clean and tidy.

1.9 In Closing

You have reached the end of this mini-guide. You can now rename or overwrite this pdf file and begin writing your own '`Chapter1.tex`' and the rest of your thesis. The easy work of setting up the structure and framework has been taken care of for you. It's now your job to fill it out!

Good luck and have lots of fun!

Guide written by —
Sunil Patel: www.sunilpatel.co.uk

Chapter 2

State Of The Art

There are already some initiatives that make use of wireless sensor networks. The word "initiative" is not intended to refer just to companies but also to individuals who belong to the do it yourself (DIY) movement.

At the time of writing, we can distinguish between two main kind of sensor networks, company or community driven networks, depending on who shapes the system.

However, during the last years sensor networks have not been used just because of their usefulness but also with data sociability and smart cities in mind. As an example, a citizen could present to the authorities actual noise levels at night, as well as knowing if a storm is coming because some friend shared his pressure and humidity levels on a social networking site.

These networks can originate a big amount of data, which along the Internet of Things create the necessity of storing this information and making it always available for further usage. Cosm (former Pachube) and Sen.se are good examples of this kind of websites and are described below.

2.1 Company-led sensor networks

2.1.1 Libelium

This is one of the biggest companies in the world built around wireless sensor networks. It offers the mechanisms and tools to deploy/build systems around the Internet of Things, smart cities and M2M communications.

The majority of the products they sell are focused on one specific application, such as waste management, structural health, etc. These are intended to be bought by system integrators for end users. However, they also offer the so-called “Waspmote”, which is a sensor device for developers that can be freely customized and reprogrammed, since this is an open source product.

Their products are being widely used across more than 75 countries and they are definitely one of the leaders of the wireless sensor network industry.

2.2 Community-led sensor networks

2.2.1 Air Quality Egg

This is a sensor network that aims, as its own name indicates, to measure the air quality through NO_2 and CO levels.

Each user is supposed to connect their egg to their local network via an Ethernet inferface. Then, a bunch of outdoor sensors are placed outside and communicate their readings to the base station (the egg) wirelessly through a radio frequency transmitter. Finally the data is sent in real time to Cosm, an open data portal that will be described in the next section.

It is worth mentioning that the AQE project is completely open, hence anyone can improve the platform as well as building his own egg from scratch without having to actually buy one. All the information related to the hardware, software and sensor calibration can be found in their wiki.

2.2.2 Smart Citizen

Although this is a very young project (still in beta stage) it intends to create the biggest community around social sensing.

It was initially crowdfounded in 2012 through a Goteo campaign and they are planning on going to Kickstarter soon.

The Smart Citizen platform allows its users to precisely geolocate their data and see other users’ information. There is also a very big emphasis in data sociability, since every value or datastream (set of values that represent an individual sensor) can be shared through any social networking site or even inside the same web application.

Openness is as well one of their main values, since every piece of code (including the website) is open source licensed.

2.3 Open data services

All gathered information must be stored in some place, and this is where open data portals come in play.

These websites provide users with an open API so they can upload new values, create new feeds (representation of an environment), retrieve the data end even create customized triggers. This way, we cannot only sense but act to certain kinds of events. Because data itself is worthless, one of their most important features are data visualization tools.

Good examples of these sites are, as mentioned before, Cosm and Sen.se. Both are free to use and very easy to interact with (mainly through HTTP packets).

Chapter 3

Methodology

Three essential blocks form a sensor network. Namely sensors, processors and communication devices[4]. In the next sections all of them will be explained and in the next chapter I will show how are these related. There is even a case where a device (Digi XBee®) fulfills two of these roles.

3.1 Sensors

We now live in a world where we hear a lot the word "sensor", but what is exactly a sensor? *A sensor is a converter that measures a physical quantity and converts it into a signal which can be read by an observer or by an instrument.*[5]

This might (and is) quite simple, but complexity resides on calibration and coming up with actual useful applications. Since every sensor from the same family is equal in terms of design but different in reality (due to small random variations during the fabrication process) output has to be adjusted to agree with a given standard. When it comes to applications, RFID tags can be used to determine wether a book is on the right spot in a library or not, or with a very intense light beam we can detect how the blood flows through a vein thus succesfully sensing heart rate. These are just two imaginative uses for nowaday sensors.

In the first (and only at the time) iteration of this pilot only environmental factors have been measured since those have been tested over and over for the last years and they serve as a proof of concept for this network.

3.1.1 Aosong DHT22[6]

The DHT22 is a low cost humidity and temperature sensor designed by Aosong Electronics, a Chinese corporation. This is a digital sensor, which means that the output is represented in the form of bits, thus requiring some amount of computational power to "interpret" the results. The output format is precisely described in the datasheet of this product, and luckily there already are some library implementations to work with the DHT22. Then, this device will only work with platforms that allow digital input, such Arduino.

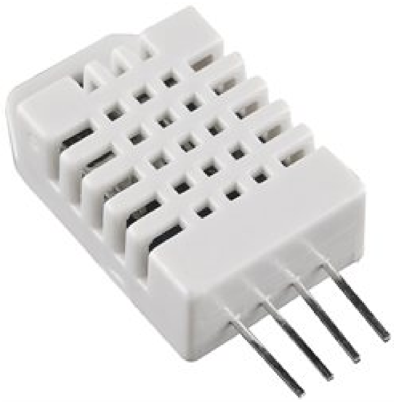

FIGURE 3.1: DHT22 humidity and temperature sensor.

3.1.2 Emartee Mini Sound Sensor[7]

Manufactured by Emartee, can also be found by the name of "Emartee part number 4021", and can be used to measure noise levels among other uses. Esentially, it consists on a microfone with a built-in amplifier onto a breakout board, which can be useful to work directly with perfboards or breadboards (both construction bases for rapid prototyping of electronic circuits).

The output signal is analog and is increased by a factor that allows an Arduino or any device with analog I/O pins to detect it easily[8]. Its operating voltage is 5V.

3.1.3 Sharp GP2Y1010AU0F[9]

This is an inexpensive optical dust sensor, used to measure air quality. It is made out of an infrared emitting diode which with a well positioned phototransistor can measure the reflected IR rays thus detecting dust levels in the air. This device, which can be powered with up to 7V gives an output voltage (analog) proportional to dust density in the air. Some of its applications are air monitoring and air conditioning.

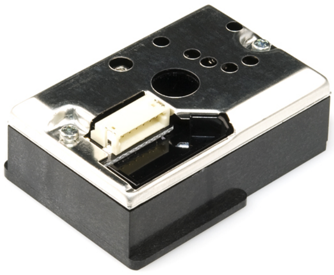

FIGURE 3.2: Sharp GP2Y1010AU0F optical dust sensor.

Surprisingly, this detector, which is worth at the time of writing about $12, gives very precise results, similar to those offered by an expensive laser particle counter.[10]

3.2 Digi XBee® Wireless RF Module[11]

These radio modules are based on the IEEE 802.14.4 standard and provide an unexpensive, low power, low rate communication. They mainly use ISM bands. Despite their size provide us with many interesting features, such as 128-bit encryption, over-the-air configuration and ADC/digital input/output pins.

FIGURE 3.3: Digi XBee® Wireless RF Module

There are two versions of these modules, named “Series 1” and “Series 2”. The older one (Series 1) implements the previously mentioned IEEE 802.15.4 standard which allows point to point (or star) topologies. On the other hand however, the latter implements a standard specification called *ZigBee*. This protocol, although more complex has mesh networking capabilities which can be a key feature in some sensor networks.

This is why for the sake of this project "Series 2" has been chosen. It is worth saying that each of the two versions can transmit with different power levels thus varying the effective communication range[12]. More detailed information can be found in the table below.

Version	Power	Indoor range	LoS range[1]
Series 1	1mW	30m	100m
Series 1 PRO	63mW[2]	90m	1600m
Series 2	2mW	40m	120m
Series 2 PRO	63mW[2]	90m	1500m

Table 3.1: Comparison of different versions of XBee®.

Also, if extra range is needed (up to 40km in line of sight) there are also XBee® devices that transmit in lower ISM bands (900 and 868 MHz). However, when transmitting in these frequencies neither ZigBee nor IEEE 802.15.4 can be used. DigiMesh™ networking protocol is the only option and it is property of Digi International Inc.

3.3 Arduino

Arduino is the leading protopying platform nowadays. It is completely open source including the schematics of the hardware itself, which is a single-board microcontroller. Anyone can program the board through a programming language very similar to C/C++ and based on Wiring. To upload a sketch (a program) to the microcontroller they also have developed an Arduino IDE based on Processing.

The amount of projects related to this platform is incredibly big, and it has gained huge popularity amongst designers, hackers, programmers and hobbyists these past years. It offers several advantages over similar devices, because it is really cheap, cross-platform and has every benefit inherent to the open source initiative. Also, like other open projects Arduino comes in many "flavours" depending on the characteristics of the project.

As it can be seen on the next picture, the board has many input/output pins that are compatible with analog and digital values. It's not just that but also it can establish a serial communication with a computer so interaction between programs and the platform can take place.

[1]LoS range refers to line of sight range, where a straight line can be drawn from the transmitter to the receiver. In this situation, there are no obstacles between them and better bitrates and/or ranges can be achieved.

[2]This output power can be obtained using high gain antennas.

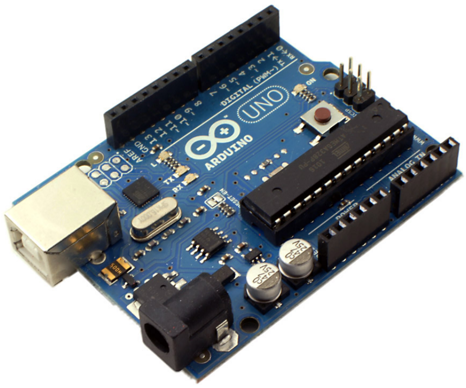

FIGURE 3.4: Arduino UNO prototyping platform.

Arduino UNO is the one "flavour" that has been chosen to perform this project, since it is cheap and also the most common. That means all shields[1] work by default on it and the community is the biggest. In particular, this model has the following features[13]:

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

TABLE 3.2: Characteristics of the Arduino UNO.

[1]A shield is another board plugged on top of the Arduino to extend its functionalities.

Appendix A

Appendix Title Here

Write your Appendix content here.

Bibliography

[1] A. S. Arnold, J. S. Wilson, and M. G. Boshier. A simple extended-cavity diode laser. *Review of Scientific Instruments*, 69(3):1236–1239, March 1998. URL http://link.aip.org/link/?RSI/69/1236/1.

[2] Carl E. Wieman and Leo Hollberg. Using diode lasers for atomic physics. *Review of Scientific Instruments*, 62(1):1–20, January 1991. URL http://link.aip.org/link/?RSI/62/1/1.

[3] C. J. Hawthorn, K. P. Weber, and R. E. Scholten. Littrow configuration tunable external cavity diode laser with fixed direction output beam. *Review of Scientific Instruments*, 72(12):4477–4479, December 2001. URL http://link.aip.org/link/?RSI/72/4477/1.

[4] Chee-Yee Chong and Srikanta P Kumar. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.

[5] Wikipedia. Sensor — wikipedia, the free encyclopedia, 2013. URL http://en.wikipedia.org/w/index.php?title=Sensor&oldid=548402365. [Online; accessed 3-April-2013].

[6] Ltd Aosong Electronics Co. Digital-output relative humidity & temperature sensor/module dht22, 2013. URL https://cdn.shopify.com/s/files/1/0045/8932/files/DHT22.pdf?100745. [Online; accessed 6-April-2013].

[7] Emartee. Mini sound sensor - emartee.com, 2013. URL http://www.emartee.com/product/42148/Mini%20Sound%20Sensor%20%20Arduino%20Compatible. [Online; accessed 6-April-2013].

[8] Emily Gertz and Patrick Di Justo. *Environmental Monitoring with Arduino: Building Simple Devices to Collect Data about the World Around Us*. Make, 2012.

[9] Sharp. Gp2y1010au0f compact optical dust sensor, 2006. URL http://sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y1010au_e.pdf. [Online; accessed 6-April-2013].

[10] Chris Nafis. Monitoring your air quality, 2012. URL http://www.howmuchsnow.com/arduino/airquality/. [Online; accessed 4-April-2013].

[11] Digi International. Xbee series 2 oem rf modules, 2007. URL ftp://ftp1.digi.com/support/documentation/90000866_A.pdf. [Online; accessed 6-April-2013].

[12] Robert Faludi. *Building Wireless Sensor Networks: with ZigBee, XBee, Arduino, and Processing*. O'Reilly Media, Incorporated, 2010.

[13] Arduino. Arduino - arduinoboarduno, 2013. URL http://arduino.cc/en/Main/arduinoBoardUno. [Online; accessed 6-April-2013].