# Anti-thief system

# EPS Final report

Carlos Manuel Andrés López-Peláez

26/01/2011

# TOC

# 1. Abstract

In this report you will read about a project made by four students at ENIT, school of engineers. The project has a technical aspect and a management aspect. The given task to the students was to research different solutions to make solar panels less attractive for burglars. The burglars steal the panels to use the panels as batteries.

One of the goals of the project was to learn how to manage a project by using MS project and how to write needed documents for the progress of a project. Another goal was to earn deeper understanding in electronics and different ways of communication between micro controllers. The main part of the report is about the prototype that was made by the students and with assistance from the supervising professors.

So the report contains the different tools they learned and used during the project, a research section of all the different ideas, different ways of communication and the developing of the prototype.

During the project the students had 4 management reviews and 5 technical reviews. During these reviews the progress of the project was displayed and you can find the minutes of meeting and the powerpoints in a digital folder that is in the possession of the EPS supervisors or visit the website http://anti-thief.weebly.com/.

## 1.1 Solarcom

Solarcom is an enterprise located in Tarbes, France. The enterprise specializes in constructing solar fields. They do not produce the panels only the assembling and solution for the costumer.

Mr. Fosse was the contact from Solarcom and he was the person that had the power to decide what was going to be developed.

# 2. Management

## 2.1 Introduction

This chapter contains the context of the management and teamwork lectures. It tells you what was taught to the students and how the used the knowledge they gain.

Before starting with the project the students gain knowledge in management and teamwork. With the professor Philippe Fillatreau, as their teacher in management, they learned the different stages of a project and the tools that help you to manage a project. The lectures were both theoretical and practical. The theoretical lectures contained:

- Definition of a project
- Life cycle of a project
- Scope of a project
- Project planning
- Time management tools
- Project monitoring
- False ideas about projects

The practical lectures were used to get knowledge and experience of how to use the program Microsoft Project which was used during the project as a planning and monitoring tool.

The student group also gained experience in teamwork before starting the project. The teamwork lectures were made by a British professor named David Swetnam. The students gained knowledge about teamwork and the common mistakes that are made in everyday life and business environment.
The lectures contained:

- Cultural aspects
- Understanding different personalities
- Practical session

During the project the students experienced some problems. They had to change the planning and create new tasks and all this is explained later in the management chapter.

## 2.2 Teamwork

Many of the topics that were discussed during the teamwork lectures may seem obvious, but they are very hard in practicum. The students experienced this during practical experiments.

The students made a Belbin test to easier understand their character and the character of the other members of the group. It's important to understand different types of personalities and their needs if a project is going to be successful.

A good team satisfaction is as important as client satisfaction and both of them need to be considered all through the project.

## 2.3 Scope of the project

### 2.3.1 Clients

At the start of a project you should identify the clients of the project. A good question to ask yourself when you start a project is "Who are we trying to satisfy?".

The clients for this project were:

- Solarcom who is only interested in the technical solution.
- EPS supervisors consider both the management work and the technical work equally.

### 2.3.2 Objectives

The objectives are important to define so that the goals of the project are the same for all the members of the project group. The objectives are established after listening to the client's needs.

The objectives for this project were:

- Should be as cheap as possible.
- Shall consume as little energy as possible.
- The module should be unable to be removed from the solar panel.

- Makes the panel only compatible inside a specific area.

### 2.3.3 Deliverables

After the clients are identified you ask yourself "How will I satisfy them?" In this project the members thought that it was necessary to make a prototype.

The deliverables for this project were:

- At the end of the project we are going to deliver a prototype.
    - The minimum is to develop a prototype functional for a standard panel with a short distance between the master and the slave microcontroller.
    - The maximum is to develop a prototype functional for all different panels in a huge range of distances between the master and slave microcontroller.
- We are going to deliver a state of art that contains our research on every solution and all the alternatives.
- We will prepare the reports at the middle and the end of the project.
- We will deliver a folder with all our minutes of meeting and our powerpoint presentations.

### 2.3.4 Risks

Before starting a project it is necessary to identify all the possible risks that it may occur. This is very important because it is a waste of money running a project that proves to be impossible to finish. Many companies think that if the risk is higher than 30 % the project won't start.

The identified risks for this project were:

- 25 year warranty risk. That means that every component in our solution needs to work for 25 years because that is the warranty of the panel.
- The black box risk. It means that even if our solution is a very good one, we might not establish our goal because the thieves might get access of the panel by connecting to the already existing black box.

- Power-communication risk. It means that the solution might need more power than the panel can provide.
- The temperature for the resin. The area where the box would be placed can reach very high temperature and it is not certain that the resin manages the temperature.

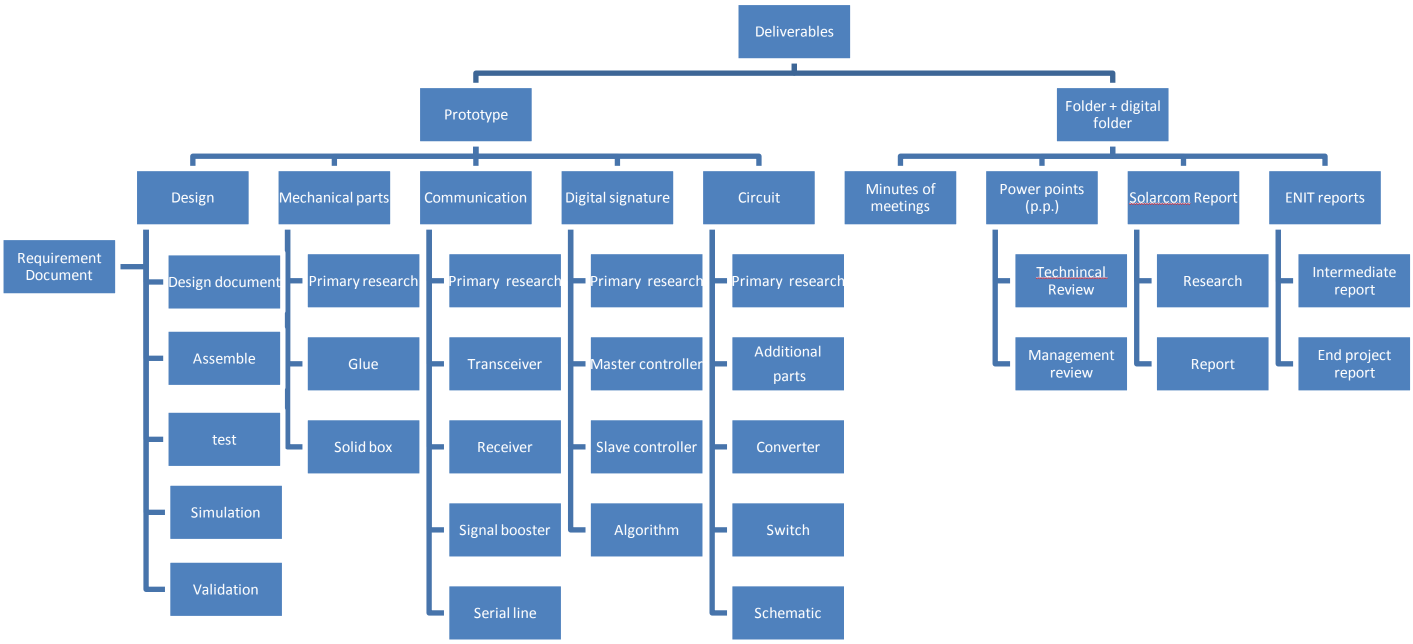
## 2.4 Project planning

After defining the scope of the project and having it approved by both the students and the clients it is time to start planning the project. If you do not have a plan for the project you will never know what's necessary to be working on, how long and which task is more urgent.

### 2.4.1 WBS (Work Breakdown Structure)

A very effective way to visualize the project is to build a WBS. The WBS is a tool that starts with the deliverables and then you break down the different deliverables in to sub deliverables. This makes it easier to establish the list of tasks that will be explained later on.

The WBS for this project was:




**Picture 1: In this picture, the WBS for the project is shown. It is a hierarchic table that starts with the deliverables as the top box**

### 2.4.2 List of tasks

During the workload we defined the responsibilities, the resources and the time we needed for each task. It is important to have one person responsible for every task for the progress of the project. According to that, each member must be informed about its responsibilities. The given responsibilities response to each member's skills and knowledge, but also give the opportunity to each member to earn and achieve knowledge.

See the full list of task in Appendix 2 List of tasks.

## 2.5 The new planning

There was a critical moment just before the Christmas holiday. Some of the group members were starting to feel worried. They thought that it was too much to do the last three weeks.

The tasks that needed to be done:

- Try if it was possible to send a high frequency signal in the power line.
- Build all the circuits and get them to work properly.
- Make a new oral presentation.
- Write the final report

The big problem was that some of the members didn't have experience or expertise in the subject.

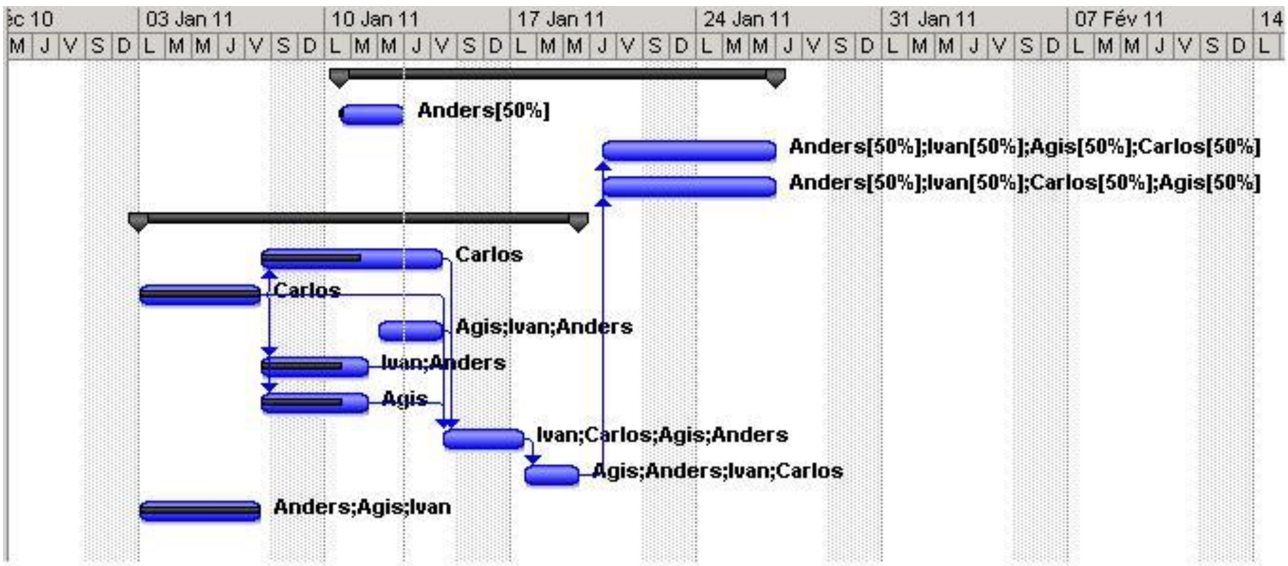
After a meeting with the management supervisor we discussed the problem. He made it clear that the students needed to make a choice of what to continue working on, make new priorities and identify new risks. The worst thing that could occur was if the project was ending not stable.
To end a project not stable is when you don't have time to close the project properly. If you for example trying to do everything and don't have time to put it in the report, it useless for the next persons that might continue on the project.

After the meeting with the management supervisor the group met the technical supervisors who were more optimistic. They told the students that there was no problem and that they would assist the group a lot during the last three weeks. A meeting was planned with professor Dixnuef on the first day after the Christmas holiday

A new planning were made and the group starting to work with the assistance of the technical supervisors.

| | | WBS | Nom de la tâche | Durée | Début | Fin |
|---|---|---|---|---|---|---|
| 1 | | **A** | **⊟ Management** | **57 jours?** | **Lun 10/01/11** | **Mer 26/01/11** |
| 2 | | A.1 | Management review ‹ | 2 jours? | Lun 10/01/11 | Mer 12/01/11 |
| 3 | | A.2 | Final report | 4 jours? | Jeu 20/01/11 | Mer 26/01/11 |
| 4 | | A.3 | Oral presentation | 4 jours? | Jeu 20/01/11 | Mer 26/01/11 |
| 5 | | **B** | **⊟ Prototype** | **11 jours?** | **Lun 03/01/11** | **Mer 19/01/11** |
| 6 | | B.1 | The code | 4 jours? | Ven 07/01/11 | Ven 14/01/11 |
| 7 | ✓ | B.2 | Build Master | 4 jours? | Lun 03/01/11 | Ven 07/01/11 |
| 8 | | B.3 | Build Slave | 2 jours? | Mer 12/01/11 | Ven 14/01/11 |
| 9 | | B.4 | Build filter | 2 jours? | Ven 07/01/11 | Mar 11/01/11 |
| 10 | | B.5 | Build multiplier | 2 jours? | Ven 07/01/11 | Mar 11/01/11 |
| 11 | | B.6 | Assably | 1 jour? | Ven 14/01/11 | Lun 17/01/11 |
| 12 | | B.7 | Testing | 2 jours? | Lun 17/01/11 | Mer 19/01/11 |
| 13 | ✓ | B.9 | Test sending signal | 4 jours? | Lun 03/01/11 | Ven 07/01/11 |

Picture 2 Show the new list of task that were made for the last three weeks




Picture 3 Show the new Gantt chart for the last three weeks of the project

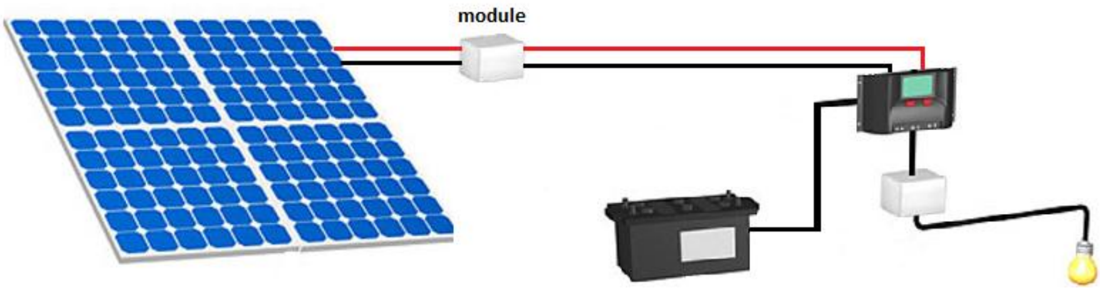
There were no new risks found and the project was on the right track again.

# 3. The Project

This chapter contains the progress of the project in the technical expect, from ideas to results. The prototype was made in order to make the solar panels less attractive to burglars.

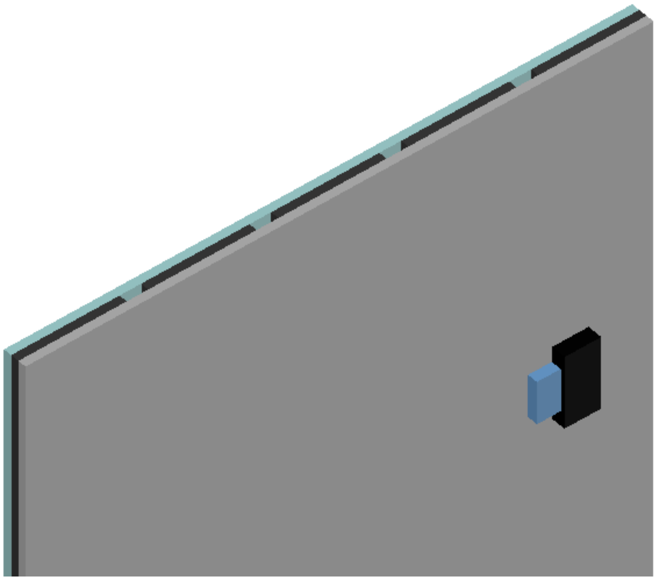
## 3.1 Introduction

The initial idea was to add an electronic module between the solar panel and the regulator that check if the panel is connected to the correct regulator and if so enable the supply of the voltage.




Picture 4 Describes the initial idea how to solve Solarcom's problems with the burglars

To achieve that, it is necessary to have a communication system between the regulator and the panel. Furthermore, it is important to develop a module that control the signal sent by the way of communication and switches the panel on. The module also has to be impossible to take off from the panel.

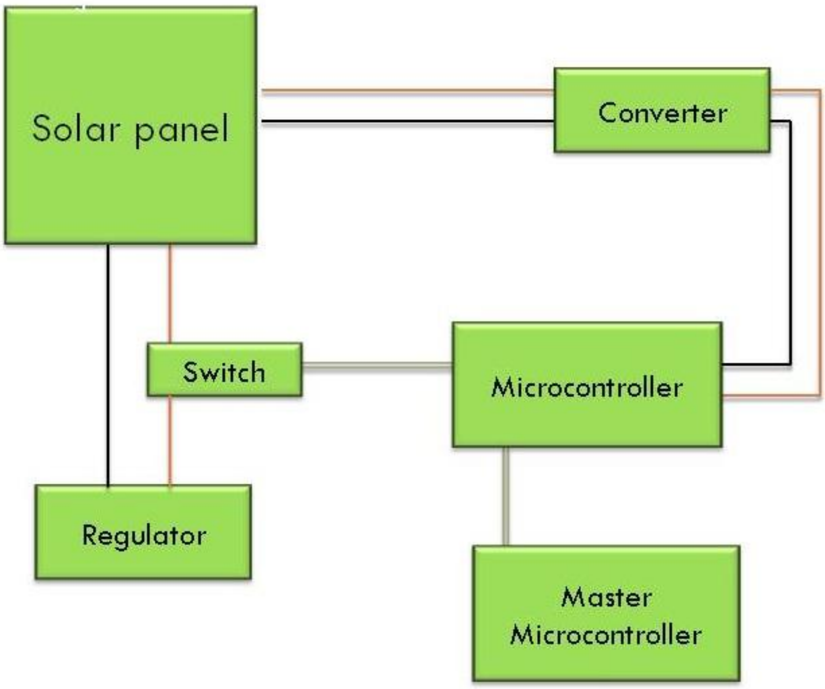


Picture 5 Show you how the initial idea of where to place the module made by Anti-thief system. Notice the already existing black box

To for fill the initial idea just presented, one solution would be to create an anti-thief system using microcontrollers for sending and receiving coded information. Because this project was presented as an electronic project, the group of students decided to develop the solution containing electric components.
So the main functions of this prototype could be:

- The converter that provides power to the microcontroller.
- The switch that controls the output of the panel.
- The master controller that sends a digital signature and the slave controller that receives and checks it.
- The communication between the two controllers
- The assembly on the panel.

The picture below explains the general solution. The reason why the master controller does not have any electrical supply is because it is not certain where it would be placed. It depends on the communication.

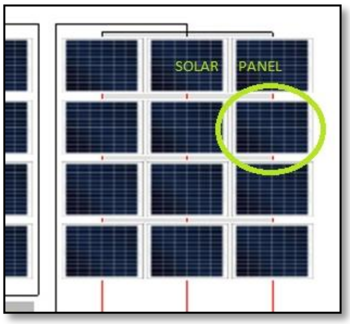



Picture 6 Describes the general solution which consist solar panel, regulator, switch, converter, master and slave controller

It is important to mention that the converter, the switch and the microcontroller should be on the same module, as explained earlier, impossible to be removed by the thief.

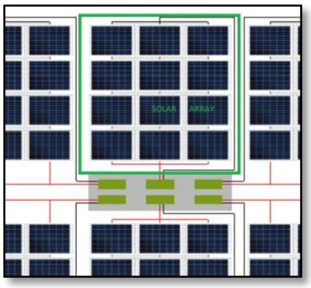
## 3.2 Definitions

1. Solar panel → The smallest solar part that cannot be divided.



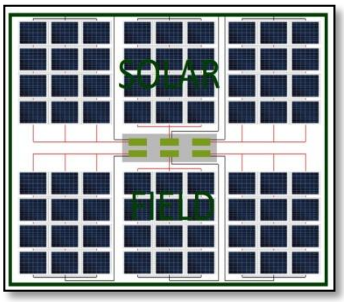

Picture 7 Explains the definition of the solar panel

2. Solar array→ Matrix of some solar panels. They can be connected either in line or in parallel. They share the same structure.



Picture 8 Explains the definition of the solar array

3. Solar field→ The matrix of solar panels and regulators explained in the next definition all together. Actually it is the whole installation.

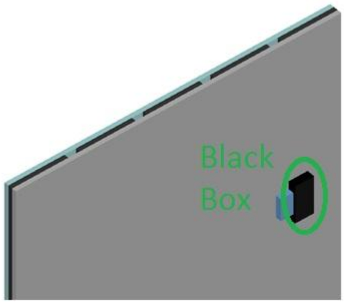



Picture 9 Explains the definition of the solar field

4. Regulator→ Controls and stabilizes the power of the solar arrays.


Picture 10 Gives you an image where the regulators are found in a solar field

5. Microcontroller→ Computes, manages and controls the good functioning of the anti-thieft system.
6. The black box→ Case placed in the backside of the solar panels. It cannot be removed, touched or modified.




Picture 11 Shows you the already existing black box

## 3.3 Research

### 3.3.1 Introduction

Before starting this project, it was necessary to make a research and to develop a small library containing a big amount of datasheets and information about different parts of the project.

The product should be compatible with different sizes of solar panels and compatible with different sizes of solar fields. So the product has to work for these criteria.

- The solar panel's output voltage can be from 12V to 60V.
- The line voltage usually achieved the voltage 200V.
- And a current up to 10 A.

This research contains information about the different possible solutions for every different function of the prototype.

- Ways of communication
- Ways of secure the system
- Switches
- Converters
- Controllers
- Assembly materials

The assembly materials were a task that never got completely finished. It is explained later on.

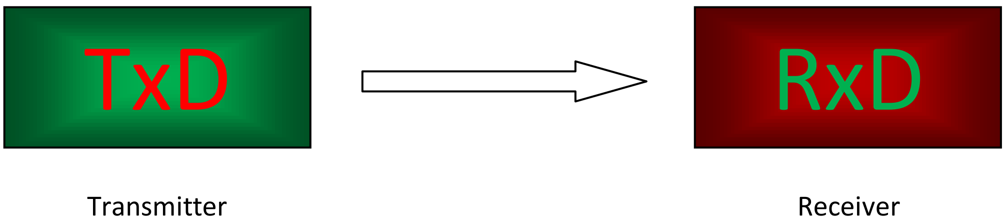
### 3.3.2 Different modes of communication

There are three modes to send the information between two DTEs (data terminals equipments):

1. Simplex (sx)

   Simplex communication is produced only in one direction between the transmitter which send the signal and the receiver which receives it. The receiver is not allowed to send any signal back to the transceiver. This mode is usually used when the interaction between the two devices is not needed. One example of that would be the radio broadcast.

***Simplex transmission***

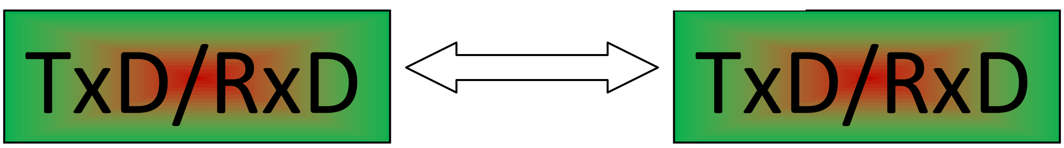



Picture 12 Shows the simplex transmission

2. Half-Duplex (hdx)

   In half-duplex communication the information can be sent in both directions, but not at the same time. It means that both terminals can be transmitters and receivers but only one is sending data at one time. The walkie talkie would be a good example of this transmission mode.

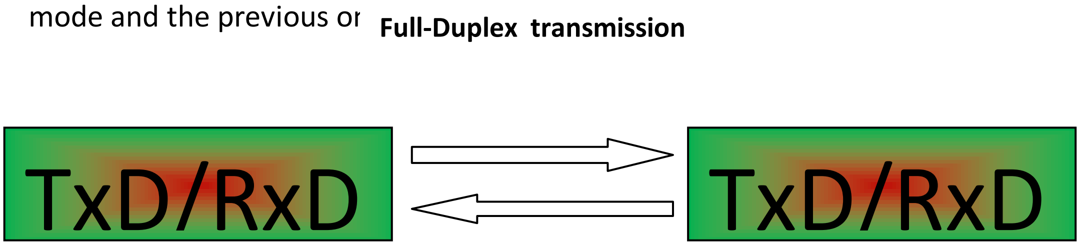
***Half-Duplex transmission***




Picture 13 Shows the half-duplex transmission

3. Full-duplex (fdx)

This mode lets both DTE's send and receives information at the same time. Because of this advantage, a second wire/frequency is needed and this is a difference between this mode and the previous or




Picture 14 Shows the full-duplex transmission

### 3.3.3 Different way of communication

In this part of the research you are going to see the different ways to communicate. That has been investigated in order to choose the best one for our application. They can be divided by communications using either physical means (wires) or electromagnetic waves (wireless). The different kinds of communication of these subgroups are shown below:

1. Communications with physical means
    a. Normal wire
    b. PLC
2. Communications with electromagnetic waves
    a. RF
    b. Wi-Fi
    c. Bluetooth

You will find a deeper explication of all of them in the following pages.

1a) Normal wire

It consists of joining the devices using a wire. This is the easiest way of communication.

On the one hand, there is no need for any extra device and the quality of the signal could be optimum depending on the wire. It means that electromagnetic interferences (EMI) and noise could be attenuated using shielded wires and the transmission could be excellent. This kind of communications allows a full-duplex communication.

On the other hand, the longer the distance between the devices, the more amount of cable is needed, and therefore the communication could be more expensive than the other options. Another disadvantage by using cables is that it is easy to stop the communication by cutting the wire in two parts.

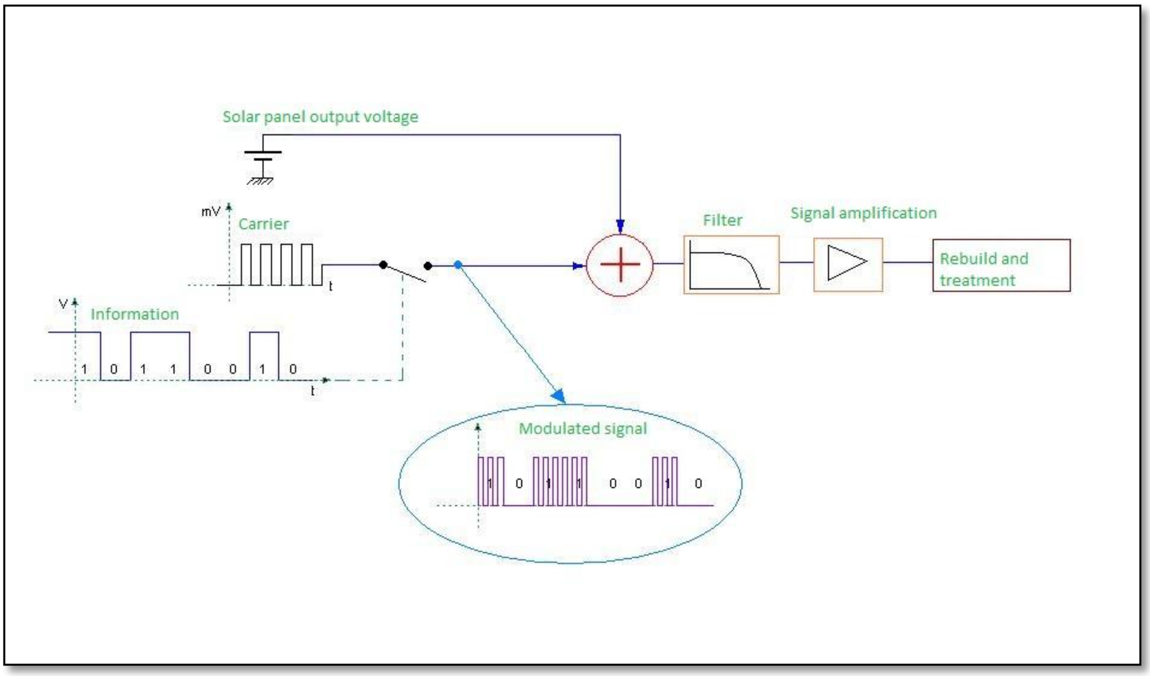
1b) PLC

PLC (Power line communications) is a communication that uses the conventional power lines in order to transfer data signals. In other words, it means that the data is transmitted through the conductor which is also used for electrical power transmission. These systems are based in adding a modulated carrier signal on the wiring system.

This kind of communication is used to reduce the number of cables, and so, the costs of the communication installations, but some devices are needed between the DTEs. It is possible to use the same system in different solar field, but it is difficult to develop a stable system for different solar fields and the signal depends of the power line current and voltage.

The PLC was originally made for AC power in conventional use and its first applications used AC lines. One of the most common applications is home internet using 230 V AC power line.

The current from the solar field to the regulator is not AC but DC, since the solar panel's voltage input is DC. There are some DC power line communication applications such as automotive, audio and of course, green energy management. As told before, whatever the plc is, a modulated signal must be added to the current to be transmitted and received by the receiver. A simple sketch of it would be like the one below.




Picture 15 Describing the power line communication

Developing and studying a system with these functionalities is not easy at all and would take a long time. Some companies have already developed integrated and reliable modules to do it, so they have been taken into account.

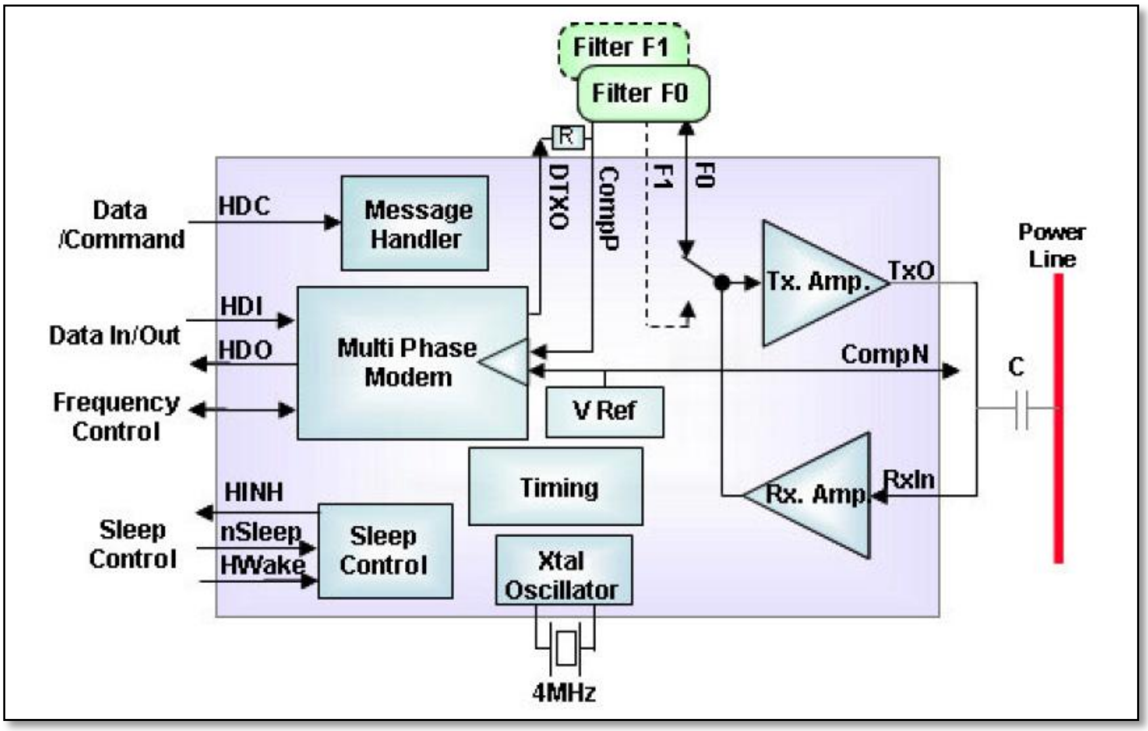
Sig-60 is a module manufactured by YAMAR© thought mainly for applications like ours. The manufacturer's brief explanation of the device is the following:

*"The SIG60 is a second generation transceiver for digital communication over battery powerline. It allows the powerline to be employed for both power and communication, thus eliminating the need for special wires for carrying control and data. The SIG60 uses a unique multiplex digital signaling technology that overcomes the powerline noisy environment. A small footprint integrates most of the external components needed for its operation. A sleep mode puts the device in a power saving mode while it is still capable of sensing the bus for remote wakeup messages from other devices. The communication over powerline reduces harness and connector size, increases reliability, saves node costs and increases the network throughput. The SIG60 operates as an AC/DC Powerline transceiver that replaces the RS232 or LIN transceivers thus eliminating the Data wire"*

**Quote 1.- From YAMAR© http://www.yamar.com/sig60.php**

The main parameters of this device are:

- Storage Temperature -55°C to 150°C
- Input Voltage -0.6V to Vdd+0.3V
- Vdd Supply voltage 3V to 3.6V
- Operating Temperature (Industrial: -40°C to 85°C)
- Idd Supply Current: 40 mA (condition:5.5MHz)
- Idd Supply Current during Tx: 50 mA (condition:5.5MHz)
- Idd Supply Current in Sleep mode: 80 uA Average current consumption
- Bit rate between 9.6 Kbps to 115.2 Kbps (good for the application)




Picture 16 Display the logical box

In the Picture 16, you can find the schematic given in the datasheet. The sig-60 and one capacitor connected to the power line are the only needed components to be able to communicate. This device could be interesting the prototype.

2a) RF

RF (radiofrequency) communications use this zone of the electromagnetic spectrum to connect some DTEs. The range of these electromagnetic waves goes from 3Hz to 300 GHz and inside there are some subgroups that depending on the frequency can be used for radars, radio, and even army communications, for example. Only few spectrum zones are free to be used by anyone. So the whole range is controlled and standardized by governments and laws. In Appendix 4 Different uses of radio frequencies, you can see the whole radio spectrum, subgroups and its allocations.

Although using this way of communication there is no need of extra wire for the DTEs, it is possible to copy the signal and the fact that between the DTEs it is needed to add more components (RF transceiver, RF receiver), so there is more risk of failure. The mode of this kind of communication is simplex between the transmitter and the receiver.

2b) Wi-Fi

Wi-Fi (Picture 17 Wi-Fi logo) is a brand of *Wi-Fi Alliance©,* the commercial organization that checks and certifies equipments to fulfill the LANs (Local Area Networks) 802.11 standards.

**Picture 17 Wi-Fi logo**

They use a 2,4 GHz (actually 5,4 GHz too) electromagnetic wave and the transmission speed is up to 300 Mbps. Wi-Fi technology is thought to be used in short distances, because otherwise it exists a big risk of interferences. One interesting point is the safety. There are many standards to code the information and data, like WPA and WEP. Another good point is the fact that there is an assured compatibility between the huge number of WIFI devices. Using this kind of communication in our project would need to develop a new network every time and add WIFI components between ETDs. It allows half-duplex communications.

2c) Bluetooth

This is a specific kind of RF communication technology as well that uses the same bandwidth that WIFI but in this case it was thought for low power consumption, with a short range (10 meters depending on the version) based on low-cost transceiver microchips in each device. Due to this small range it would not be useful in one project like ours. The transmission rate could be up to 24 Mbps.

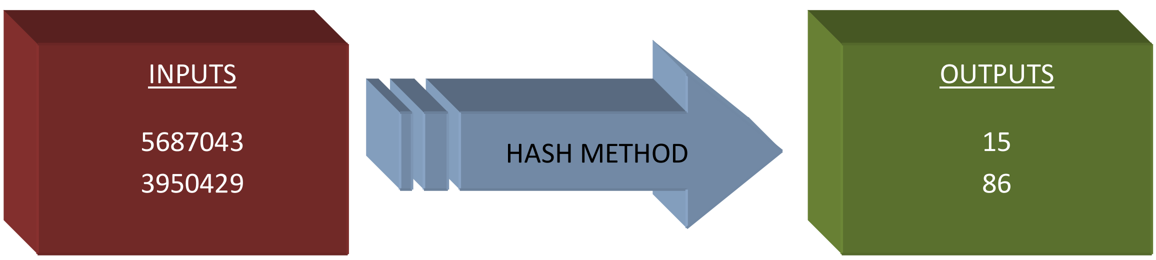
**Picture 18 Bluetooth logo**

### 3.3.4 Different kind of localization

#### *3.3.4.1 The hash method*

A hash method is a mathematical function that converts large, variable size amount of data into a smaller datum. They usually have as an input a long number or string (usually more than 6 characters) and as an output an integer number, the size of 3 characters for example. The values are returned from a hash method are usually called hash values or sums or just hashes.

Here there is an example of hash method that gets as an input a number of 7 characters and an output of 2.




**Picture 19** The hash method

Here is an example of a hash method that receives an input a number of 7 characters and sends an output of 2 numbers:

- The input is 5687043
- Module the number with 10000
- Divide the number with 10000
- So now there are 2 new numbers

  A = 568

  B=7043
- Now add the digits of each number

  A = 19

  B = 14
- **Add the digits again**

  A = 10

  B = 5
- **Add the digits again**

  A = 1

  B = 5
- **Final output**

  F = A*10+5 = 15

It is important to say that this method is only a one-way method which means that you can't convert a hashed number back to the original (in our example there is no way to return to 5687043 from the 15).

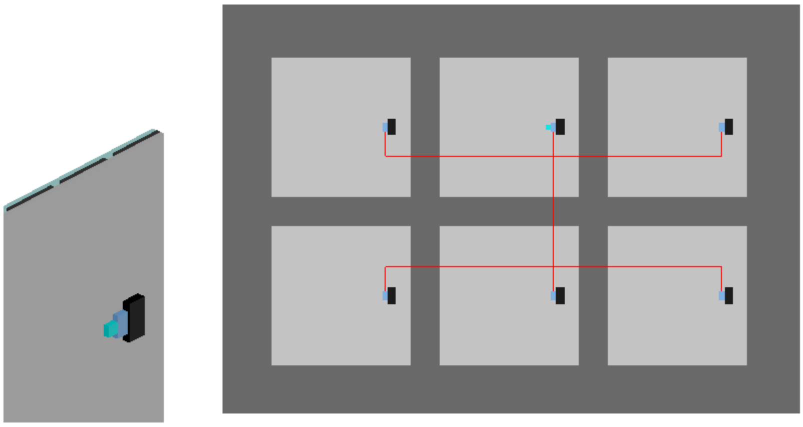
You can also note that a lot of numbers can have the same hash number. (In our example also the number "1000005" has 15 as a hash value).

The hash methods are usually used in the big databases. Are useful to delete the unnecessary parts of a number, order numbers in a short, and storage them in smaller databases. For example hash method can be used to order a list of DNAS and separate them according to specific characteristics. When it's necessary to search one DNA the hash method can be used to speed up the research.

The hash methods are used in big databases to storage and search the contains, fingerprints, randomization functions and cryptographic functions. In the project the hash algorithm is useful to random generate and cryptograph passwords.

### *3.3.4.2 GPS*

The second way to secure the panels is using a GPS which tells the right position of chip with less than 3 meters accuracy. This system contains a few satellites in a determinate orbit around the world and sends signals. Some special chips are able to understand the data sent from the satellites and calculate the position where they are. It means that if we add in our module one GPS receiver, it would have as output the same signal which is the position where it is located.

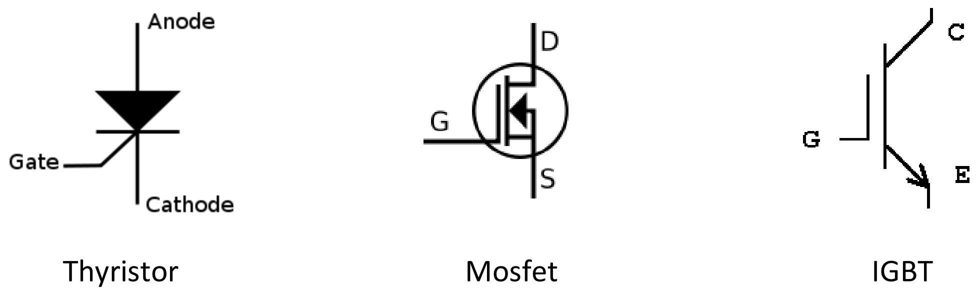


**Picture 20 Shows the GPS on the panel**

As you can see in the Picture 20, one solution against the thieves would be to add, on the backside of the panel array, a GPS that would send the position to the mastercontroller and at the same time to the several slave controllers. If someone would move the panel array, the position signal from the GPS would change and master controller would send a wrong password, and then the slave controllers would receive the wrong code and would not let the solar panel work.

### 3.3.5 Switches

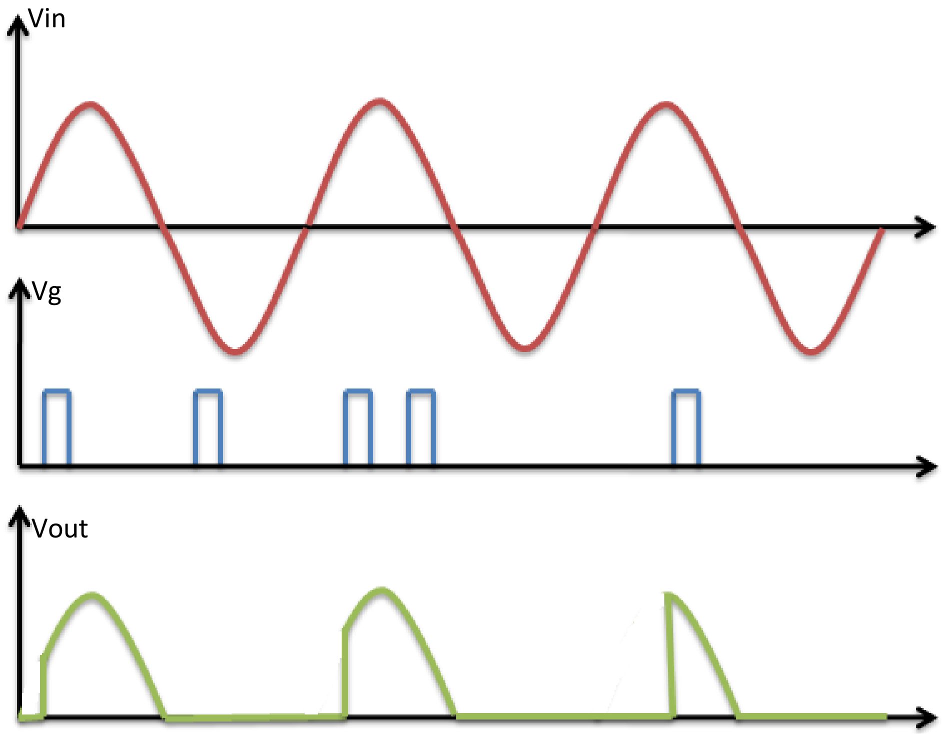
One of the most delicate points is the switch. The switch must be able to support 200 V and 10 A. In the market there are several kinds of switches that could be proper to the system but this document only explains the top three whose are thyristor (SCR), IGBT and Mosfet. Their symbols are showed in the picture below (Picture 21).




Picture 21 Shows the three different switches, Thyristor, Mosfet and IGBT

#### *Thyristor*

The thyristor is the most simple of them. Its operation is like a diode and it needs a signal to activate it. It means that, the thyristor will not work until it has received the signal (Vg). Once it has received the signal, if the voltage is positive over the thyristor, then it will let the current pass until the voltage is negative, at this moment the thyristor is deactivated automatically. This device is used in applications with medium and high voltage because the thyristor can work with high voltage. Moreover, its main disadvantage is that it consumes lots of energy. In Picture 22 the thyristor operation is shown.



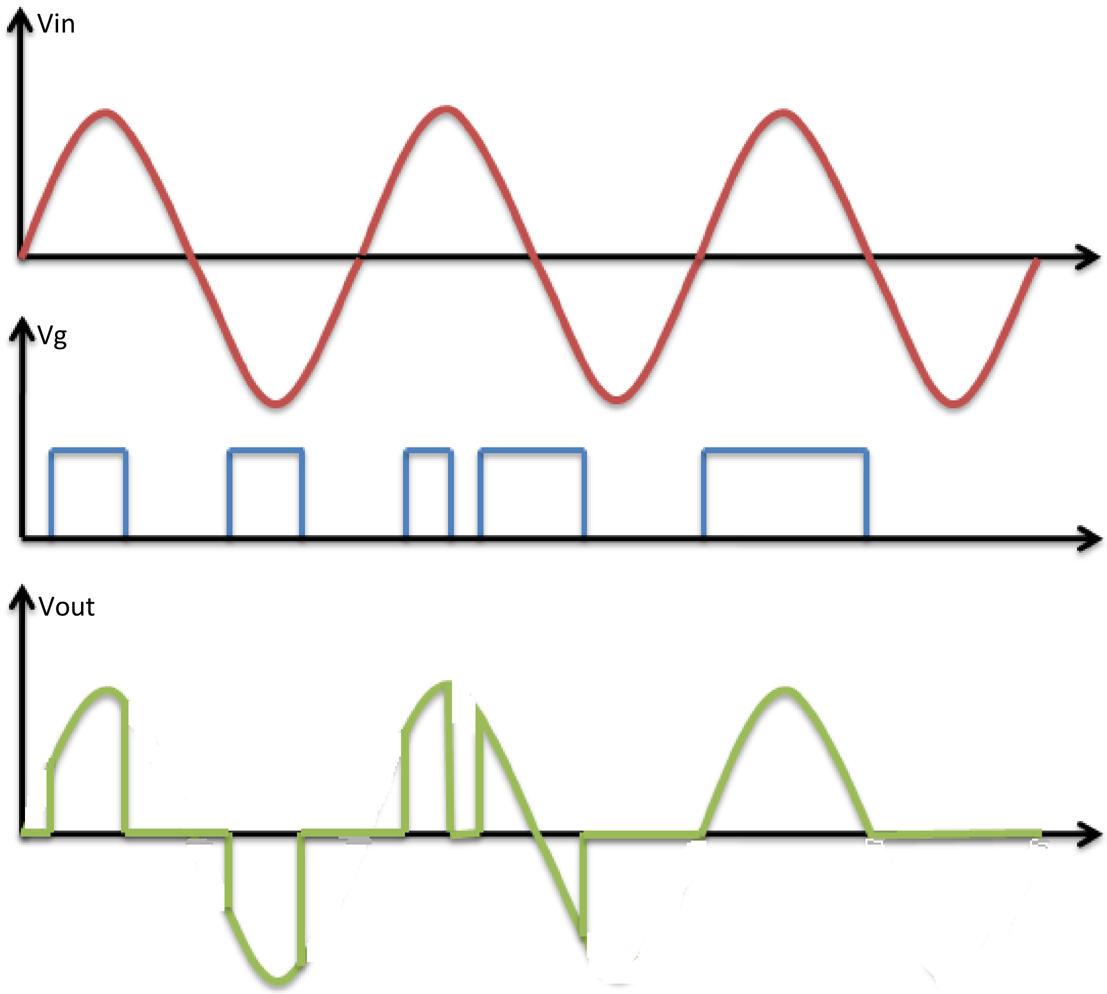

Picture 22 Shows the Thyristor's operation diagram. He X-axel is time

## *Mosfet*

The Mosfet is a field-effect transistor that it is used to switch electronic signals. There are two kinds of Mosfets: the first one is the p-channel Mosfet and the second one is the n-channel Mosfet.

The operation of the p-channel Mosfet is as follows: at first, the Mosfet doesn´t allow the current to pass (open circuit). When it receives a high level signal, at the gate, the Mosfet let the current pass (closed switch).

The operation of an n-channel Mosfet is completely the opposite. The Mosfet's transistor consumes little energy. This is the reason why they are used in applications of low and medium voltage. In the Picture 23 is showing the n-channel Mosfet's operation.




Picture 23 Shows the Mosfet's operation diagram. X-axel is time

### *IGBT*

The IGBT combines the simple control of the Mosfets with the high current and low saturation voltage capability of bipolar transistors in a single device. The IGBT is used in medium to high power applications. The operation is the same as the Mosfet but the IGBT consumes more energy than Mosfet.

### 3.3.6 Converter

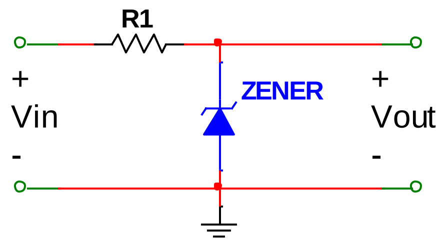
A voltage regulator, a converter, is an electronic device designed to protect electrical and electronic devices of variations in voltage. Below there is a declaration for some methods of voltage regulation.

### *Voltage regulators based on a zener diode.*

A Zener diode is a type of diode that permits current not only in the forward direction like a normal one, but also in the reverse direction. When the diode is working the reverse way, the Zener sets an output voltage called "Zener voltage".

### *Parallel regulator*

The parallel regulator is the easiest system to control the voltage. It only needs a resistance and a zener diode as it is shown in the Picture 24.




**Picture 24 shows a parallel regulator schematic**

The zener sets the maximum output voltage ($V_{out}$) and the resistance (R1) must support the voltage difference. When the current is high, the zener absorbs excess current but if the current is too high, the current will destroy it. Below, it is showed the equations that describe the regulator.

$$V_{in} = V_{R1} + V_{out}$$

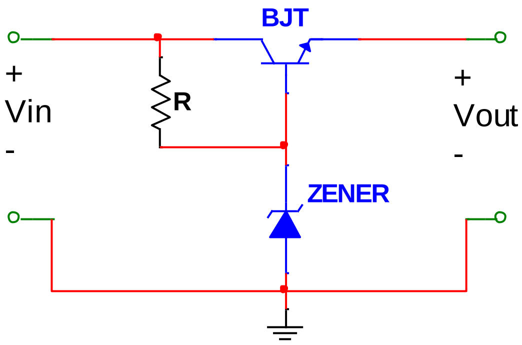
$$I_{in} = I_{zener} + I_{out}$$

### *Transistor regulator*

This regulator is a bit more complicated than the parallel regulator but, in this case, this device regulates the current automatically depending on the load needs. As can be seen in the schematic (Picture 25) this regulator is composed by a zener diode, a resistance and a transistor BJT. The equations that describe the regulator are the following:

$$V_{out} = V_{zener} - V_{BE}$$

$$R_{max} \leq \frac{V_{in} - V_{zener}}{I_{zener\_min} + I_{B\_max}}$$



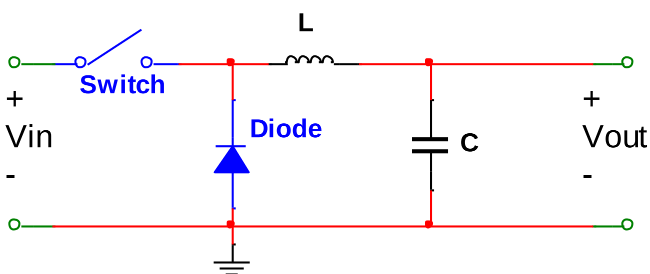

Picture 25 shows the transistor regulator schematic

### *Switching regulators*

The operation of a switching regulator is based on trying to use as little energy as possible and to avoid dissipating energy unnecessarily. To achieve this, these regulators are continually switching the input signal as a function of the energy demand. Below, it is explained one kind of switching regulator to understand better its operation.

#### Buck

Buck is an easy converter for dc-dc that is used to decrease voltage. As can be seen in the schematic (Picture 26) this regulator is composed of a diode, an inductance, a capacitor and a switch.




Picture 26 Shows the buck schematic

When the switch is closed the inductance (L) and the capacitor (C) are recharged to supply enough power to the load. To control the switch is often used a PWM (Pulse

Width Modulation) signal. When the load is constant, the duty cycle is constant but if the load is variable, the duty cycle will be variable. In this case, it is necessary to implement a system control to control the switch.

Switching regulators are the most efficient converters but are also more complex to implement. Nowadays, it is possible to find an integrated circuit that functions as a switching regulator for a low price. An example is integrated circuit MC33063A.

### 3.3.7 Microcontroller

Microcontrollers (µC) are the most important parts of our system since they compute, manage and control the functioning of the system. The main parts of the microcontroller are the memory, the cpu and input/output pins. The code is necessary to make the microcontroller operating and it is saved in the memory. The code describes the microcontroller operation and the cpu executes the code. The cpu only follows the instructions of the code. The input/output pins are necessary for the microcontroller to communicate.

An example of the microcontroller operation is flicker a led. To achieve this, you have to write a code where you describe the operation. For example: switch on the led, in pin 7, for 1 second and then switch off the led, in pin 7, for 1 second and do this repeatedly. When the code is ready, you have to download the code into the microcontroller. To download the code from the computer into the microcontroller the inputs pins are used. Finally, the led, that is connected to pin 7, will start to flicker. Additionally, the cpu can have other modules such as timers to measure time, standard communication port or AD converters to measure analog signals.

Microcontrollers are able to last at least 20-25 years which is required for the solution.

MC9S08QG8 of *Freescale* is the chosen one for the prototype. You can find some important data about the microcontroller in Appendix 5 Characteristics of microcontroller.

### 3.3.8 Table of choice

The table of choice (for the complete table see Appendix 7 Table of choice) is a table made by the students and shows the solution they thought was the best one and recommended it to Solarcom. The table of choice was made in the purpose to make it easier for Solarcom to decide which solution to work on.

The decision was taken using the scores of five different parameters: Cost (price), Consumption (power, watts, efficiency), Installing (Easy or hard to set up), Protection (against thieves) and utility (how easy it can be repaired).

For each solution there is one mark in each parameter, but not all the parameters have the same level of priority. The parameters are from 0.1 to 0.3, cost and protection were scored by 0.1, the lowest score because it is not the most important feasibility: The whole panel structure could cost thousands of Euros, so if our method costs 100 or 200€ makes no difference.

All the solutions have a good level of protection and some are better than others but in the end all of the solutions would protect the solar field against the thieves.

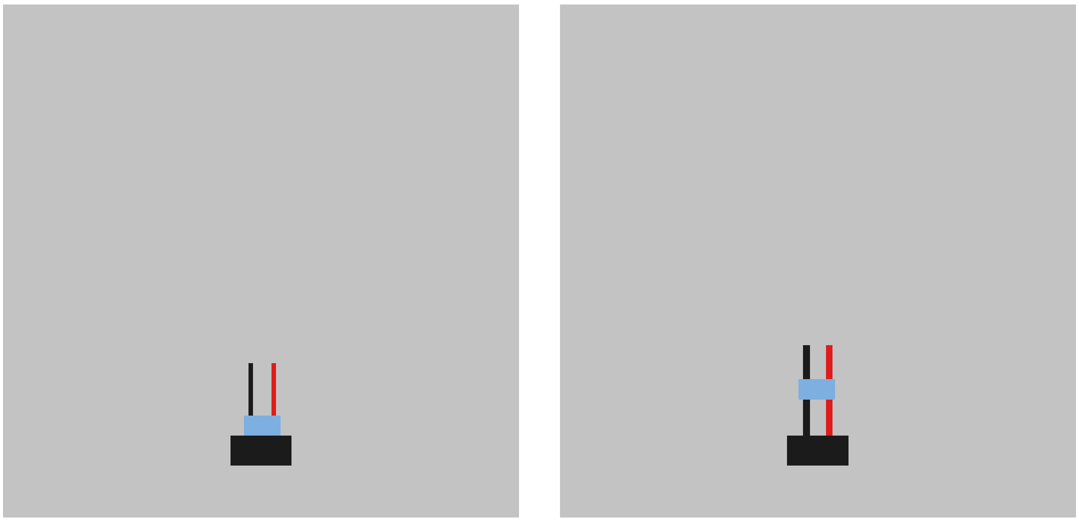
The next parameter was the utility, scored by 0.2, and finally, the most important parameters taken into account for the decision were the consumption and the installing, scored by 0.3, the highest mark. For example, the consumption is very important since the solar field must be as efficient as possible.

Giving values to every solution's parameter, the solution with the highest score was the mastercontroll of the ground using RF.

Solarcom on the other hand thought that PLC would be the best way of communication.

### 3.3.9 Assembly

The anti-thief system is useless if the thieves can bypass it. The module must be handled on the panel by a non-returned way. The panel use a black box which has the simple outputs "+ V, -V" and the thieves can use them if they are able to reach them. So an important part of the project was to make a protecting box for the module and also protect the black box.



Picture 27 Shows the position where to attach the prototype's module on the panel. The right picture shows a wrong method to attach the module. The thieves can reach the V+ and V-

**Solar panel assembly**

We research many ways such as removing the black box and put the module inside the panel but all of them harmed the warranty of the panel.

So the final solution is:

- To make a solid box which protect the module
- Glue it on the panel
- Make sure that there is not a gap between the black box and the module
- Cover the black box too

The material which is best suited for all these applications is the resin that can be specified for our plastics.

## 3.4 Prototype

### 3.4.1 Introduction

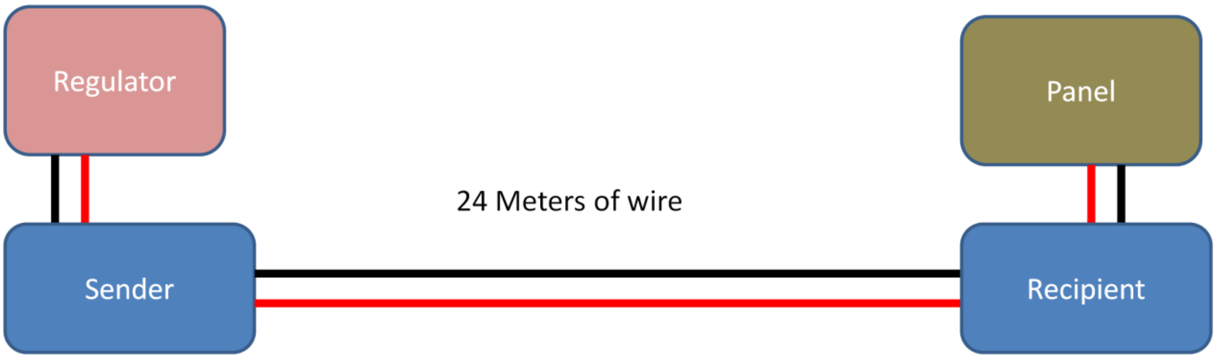
A prototype is an early model of the product that is built to test the functions of the product separately or all together. Depending of the product, usually there are a lot of prototypes, sometimes one per function, before the last product and the start of the production. Many times the prototype does not even look like the end product.

In this section, 3.4 in the report, it describes several parts of the prototype that was built and tested.

The specifications of this prototype are different than the final product too. Electric and communication characteristics were formed to the lab environment for the tests. For example the input voltage of the prototype is established to exactly 12 Voltages but in real life the systems input voltage varies.

### 3.4.2 The general scheme of the prototype

In this section the general scheme is going to be explained and later on in the report there will be a deeper explanation about the different circuits and the different parts of system.




Picture 28 Show the prototype

First of all you have to understand the principle of the prototype. Easy explained, there is a sender and a recipient. The sender is sending the signal and the recipient receives it.

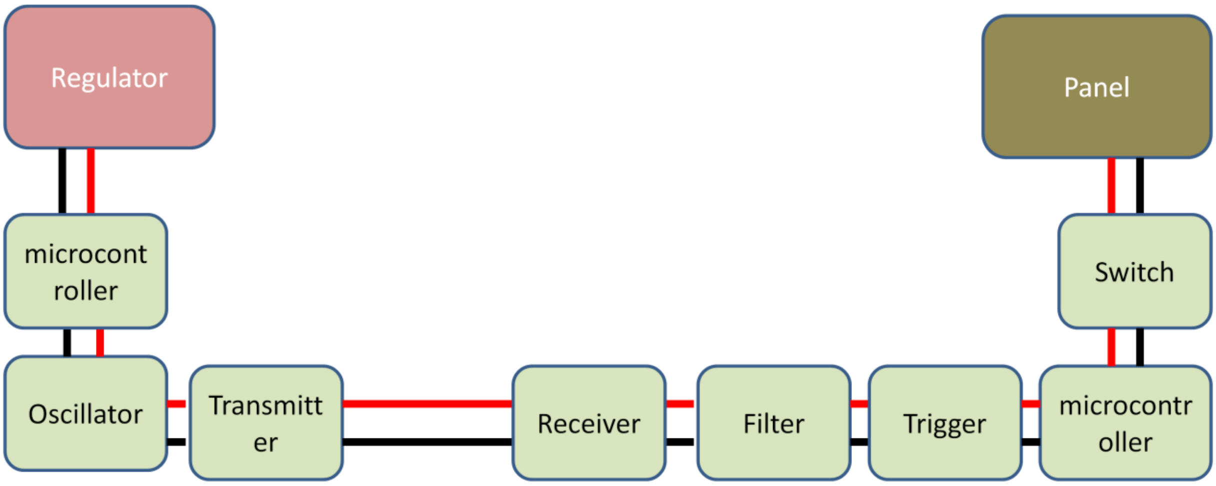
The sender contains:

- Microcontroller
- Oscillator
- Transmitter

The recipient contains:

- Receiver
- Filter
- Trigger
- Microcontroller

- Switch




**Picture 29 General scheme**

The first step in the prototype is the microcontroller. The microcontroller can produce a random number, decrypt it and send it. The sent signal from the microcontroller enables or disables the oscillator. The oscillator's only function is to create a high frequency signal that is induced in to the powerline, where the current is floating, by the transmitter. So, the microcontroller, oscillator and transmitter are the sender in the prototype. The sent signal is now travelling through the wire to the other part of the system, the recipient.

It is time to collect the high frequency signal from the power line. First of all the receiver collects the high frequency signal. After the signal is collected by the receiver, the filter erases the signals with the wrong amplitude or wrong frequency. The, now clear, signal is now collected and understood by the trigger that transforms the high frequency signal into a digital signal that can be understood by the microcontroller on the recipient. The microcontroller on the recipient checks and validates the number that has been sent by the microcontroller on the sender. And as the last happening the switch switches the panel on or off depending of the signal.

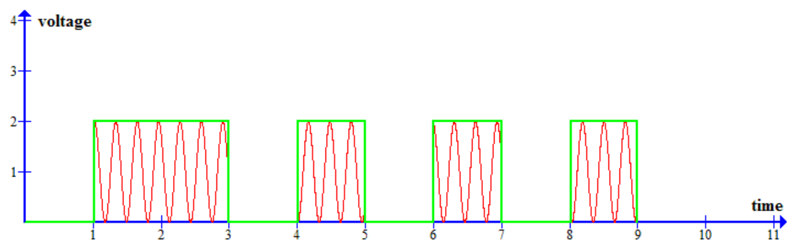
### 3.4.3 Transmitting and receiving

To embed a signal in a power line neither the analog nor the digital waveforms are possible to be used. The analog is impossible to be send over long distances because of the noise and the long wire effect. The digital is difficult to add to the line and to receive it afterwards .To send signal through power line it is necessary to use a combination of a high frequency square waveform and a duration-decoded pulse signal . See appendix for detailed information about digital and analog signal.

The solution is to send a standard square waveform with modulable durations. In this case the signal is able to travel in the wires with low current and low losses. The square's signal form describes why it is possible to receive it after a long distance without a loss, see Appendix 9 Explanation of digital and analog signal . Furthermore the high frequency signal doesn't contain any data (to be modified during the affection) but it is decoded in the duration. The duration of the signal is decoded to represent the bits "0" & "1". By this way to send the signal, the signal has the possibility to arrive to the receiver without losing its strength and also be readable.

To determine the best frequency of the signal it is necessary to count in the fact that higher frequency, the more difficult it has to travel long distances and need more power. On the other hand if the frequency is too low there is a possibility to cause complications to the regulators of a solar filed. Furthermore the low frequencies are more difficult to be filtered later. So the frequency of 200kHZ was chosen as a safe solution because:

- The prototype needs a high frequency signal to be able to transmit in long distances.
- The higher the frequency is the easier it is to filter it.
- Not in the same frequency range than other communications, power signals or switches commutation, such as the regulators which works in 20 kHz frequency.
- The higher the carrier signal frequency is the higher technology is needed, so the more expensive the components are (higher bandwidth).



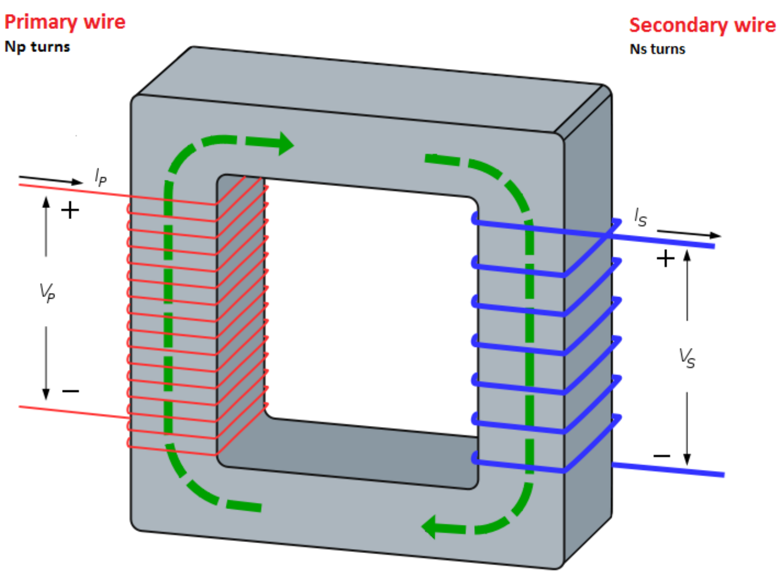

**Picture 30 In these picture the double duration represents the bit "1" and the other the bit "0"**

### 3.4.3.1 Embedding the signal to the lines

To embed the signal we simply used a transformer utilizing the inductive effect.

Electromagnetic induction (or simply electric induction) called the emergence of electricity due to the magnetic field. It is the phenomenon of increasing the voltage tension in a wire that exists inside a changing magnetic field. In other words, disturbance of the magnetic field causes disruption of the electric field. This means that a variable magnetic field produces electricity. Furthermore a wire that carries a current produces an electromagnetic field.

A transformer is based in these basic principles. The primary wire that contains electricity of Vp voltage and Ip current produce an electromagnetic field. The variable magnetic field causes pontentional and electricity difference to the secondary wire. A ferromagnetic material is necessary to be used to ensure that the magnetic lines will travel through the primary and secondary wire.

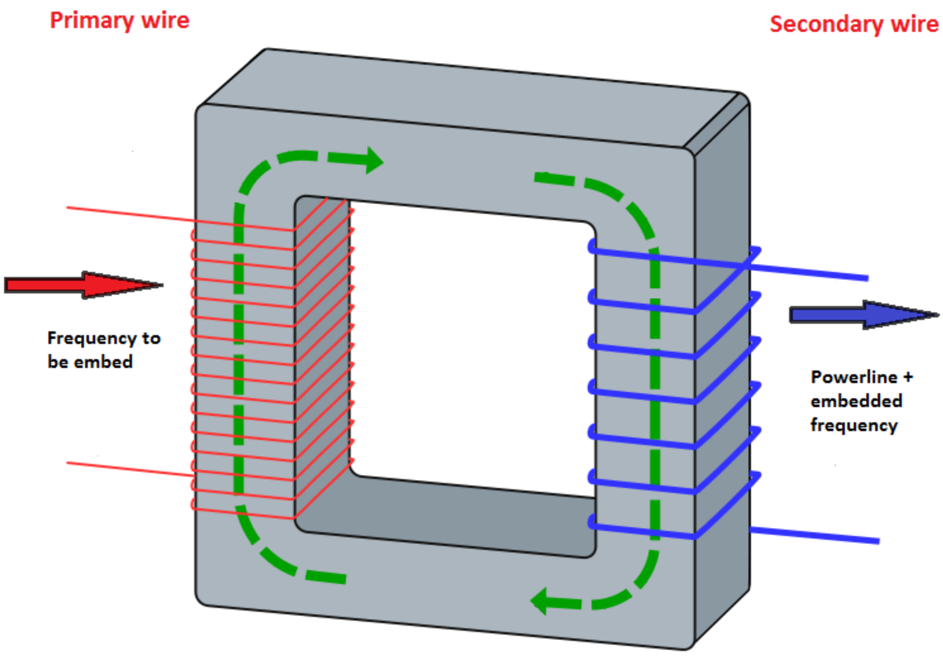



**Picture 31 Np is the number of turns of the primary wire, Ns is the turns of the secondary wire, Vp and Ip the tension and the current of the primary wire, Vs and Is of the secondary. The green line is the flow of the magnetic field.**

The following formula calculates the connection between the voltages and the number of turns.

$$\frac{V_S}{V_P} = \frac{N_S}{N_P}$$

In our cause the high frequency sine wave represents the variably electricity in the primary wire that produce the variable magnetic field. The secondary is the power line wire and because of the inductive effect the signal is embedding to it.




Picture 32 transformer

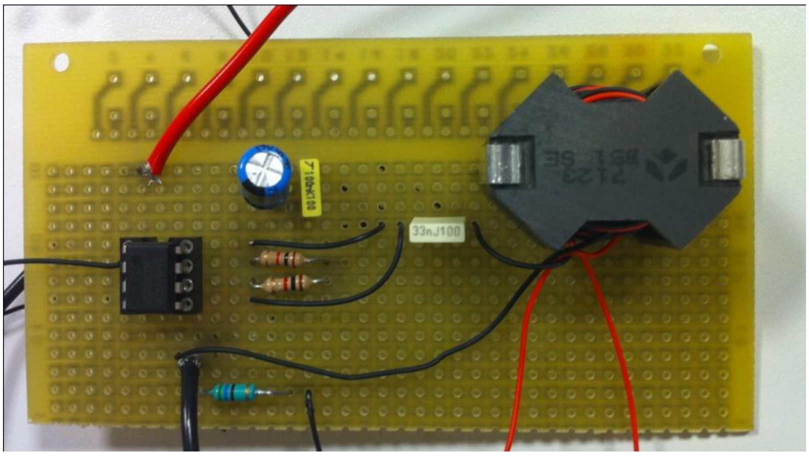
To receive the signal we are using the same technique but the other way around. The power line is now the primary wire. The embedded sine signal creates the magnetic field and the receiver is connected to the output of the secondary wire that produces the same signal which was initially embedded.

In our prototype, the primary and the secondary wire have the same number of turns in the transistor. The number was calculated to 19 turns together with Dixnuef. This leads to the fact that the input voltage and the output voltage have the same value.

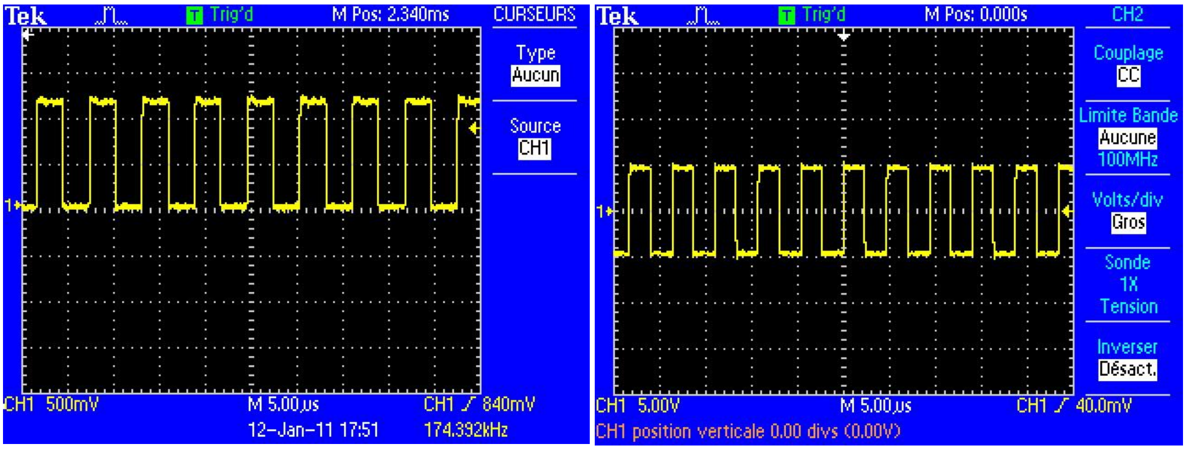
The transmitter cannot send the signals directly after it receives it from the microcontroller's oscillator. It has to modify the signal before inducting it to the power wire.

First of all the oscillator produce a palm of 200 kHZ between zero and 12 voltage. This signal must be amplified to 12V to be strong enough. After that a capacitor is used to offset the signal to be between -6V and +6V.

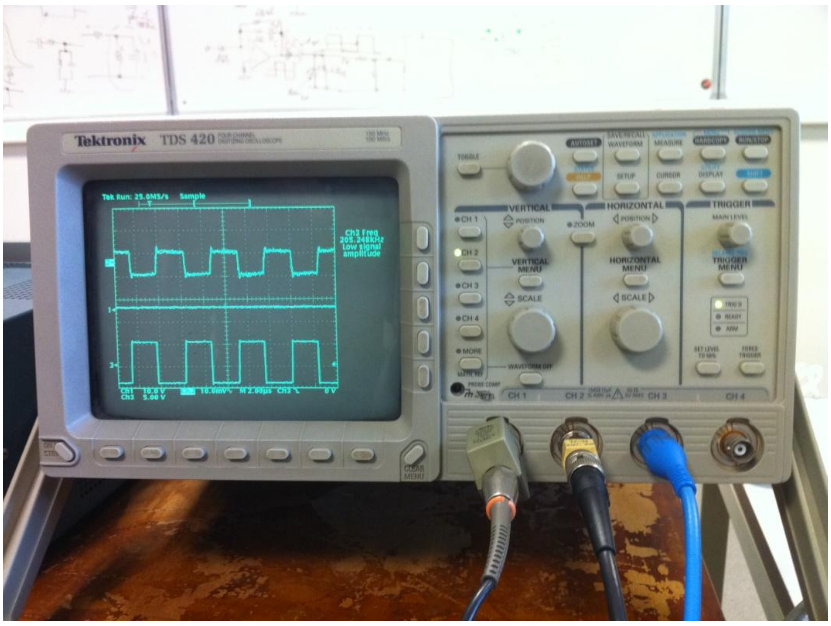
See Appendix 14 Schematic and pictures for the detailed schematic.



**Picture 33 In this picture you can see the transmitter. The amplifier (black with 8 pins) receives the input from the microcontroller and it is connected with the capacitor and the transformer.**




**Picture 34 The signal before and after the offset application**

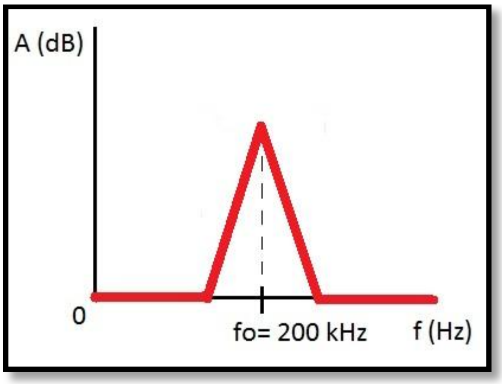



Picture 35 Sending and receiving a waveform of 170khz

### 3.4.3.2 Receiving the signal

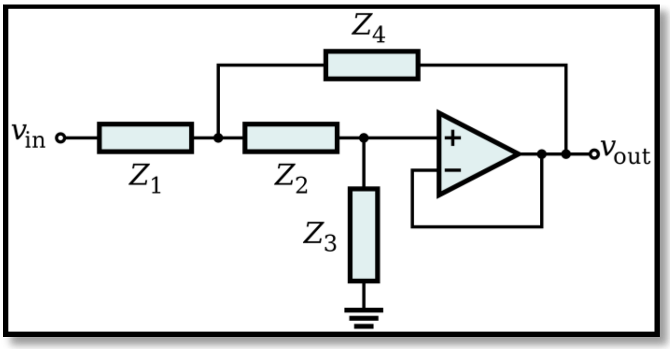
***Filter***

Once the signal is sent by the transmitter, it must be treated in order to get the correct frequency when it reaches the receiver. The distance between the transceiver and the receiver could be long, so the signal could have interferences and undesirable noise. In addition, in the prototype there are only two DTEs (Data Terminal Equipments), one for the transmitter and one for the receiver, but on a solar field there would be many transmitter and receivers which leads to many signals. Due to this reason, a selective filter is needed in the input of the receiver in order to only let the desired signals in. In other words, the filter allows just a small range of frequency signals to pass to the receiver. In this case, the transmission works with a 200 kHz frequency signal, so the filter must accept this frequency and some frequencies close to 200 kHz and delete or attenuate the rest. The picture below shows the ideal filter:



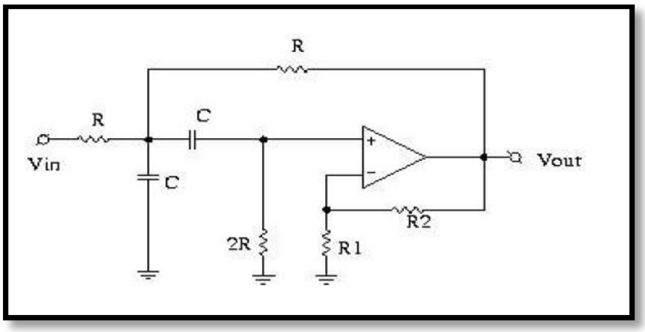

Picture 36 Ideal 200khz filter

As seen Picture 36, a selective filter must be chosen for this application. There are many types of filters, and many topologies. The chosen one for this prototype is a selective filter using Sallen-Key topology. With this topology, it is possible to create all kind of filters by changing the impedances using differences resistors or capacitors. These kinds of filters are commonly used due to their simplicity since they use very few components. In addition, Sallen-key filters are active. It means that they use not only resistors and capacitors but also operational amplifiers. The generic one is the next:



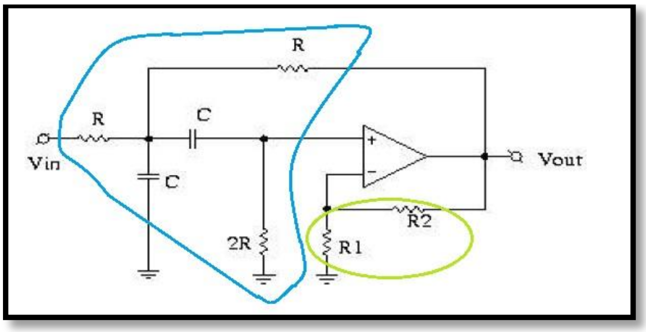

Picture 37 Generic Sallen–Key topology

As told in the previous paragraph, to create a selective filter, it is needed to change the impedances like Picture 37:




Picture 38 Selective Sallen-Key filter topology

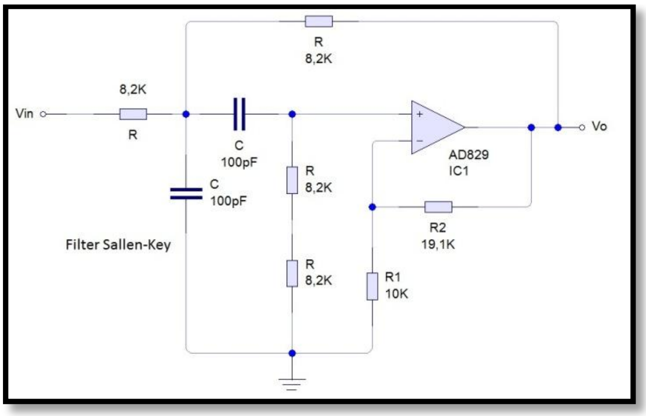
The mentioned circuit could be divided in two parts: The first determines the center frequency (fo=200 kHz), and the second the Q factor (also called quality factor and it characterizes the filter's bandwidth relative to its center frequency, so it gives information about how much selective the filter is and the higher this value is, the better the filter is and the harder to build is as well), the gain of the operational amplifier (G), and the gain in the center frequency (Am). The first part is surrounded by the blue line and the second one by the green as it shows the Picture 39.




Picture 39 Different parts of the Sallen-Key selective filter

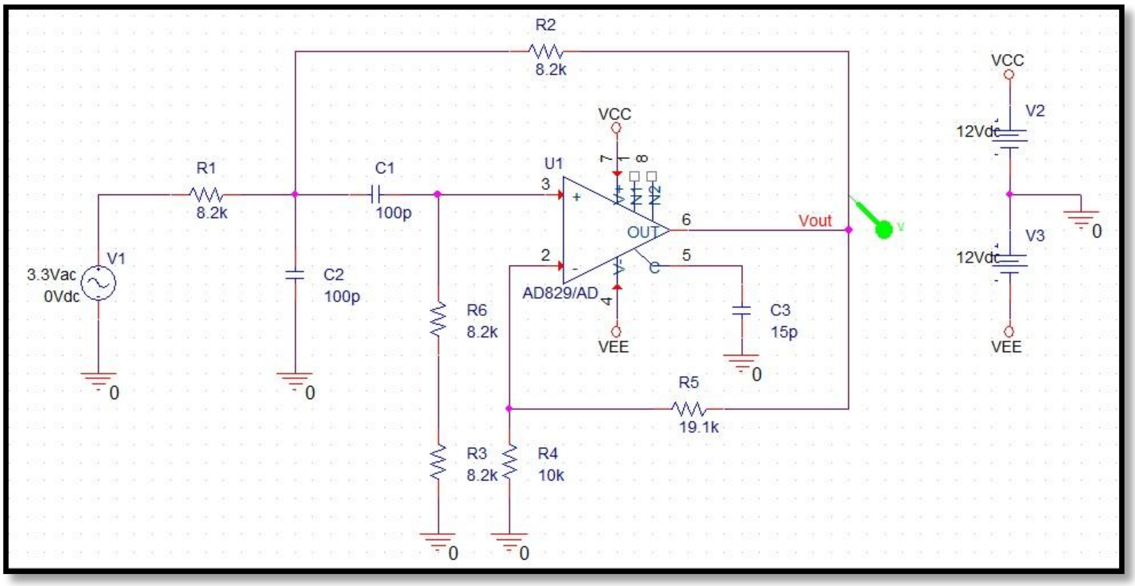
Apart from these components, there is also an active device, the operational amplifier. The chosen one is AD829 because it can work with high frequencies such as ours. Actually it is a video amplifier.

Substituting the real components for the generic ones in the Picture 40, the result is the following:




**Picture 40 Filter's schematic with standardized components**

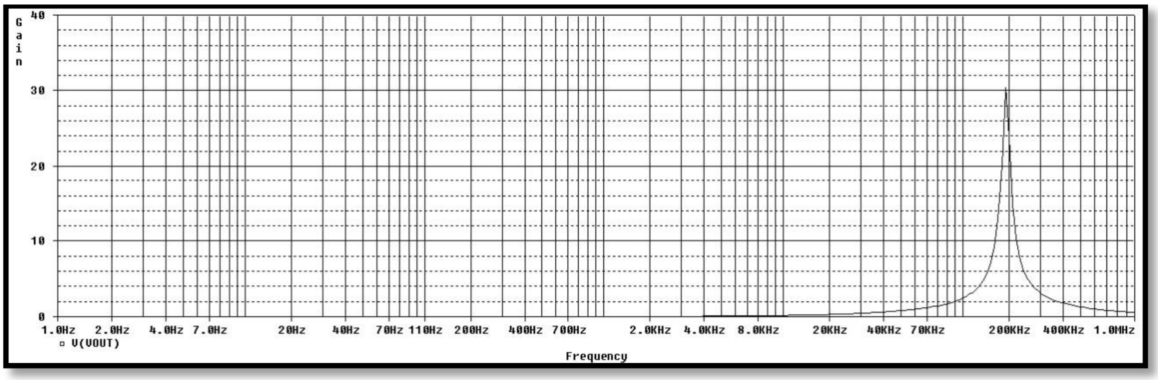
Once all the components are known, it is time to simulate the filter. PSpice© is an electronic program and it is used to simulate circuits since its simulations are close to the reality. This program has been used to do the simulations of the filter:




**Picture 41 Schematic filter for PSpice simulation**

In the previous picture you could see the schematic of all the parts needed to do the simulation. A sinusoidal signal source of 3,3 Volts peak has been added to simulate the input of the circuit. Although the signal sent by the transmitter is square, due to the long distance it becomes sinusoidal in the input of the receiver and it belongs to the first harmonic of the square signal. In the previous schematic it has also been added the ±12V supply and the capacitor given for the datasheet for the integrated AD829.

The Bode diagram shows the system's frequency response, so it gives information about the filter. In other words, this diagram shows the different gains for each frequency. Using PSpice it is possible to have this diagram, and the result is the next:




Picture 42 Bode's diagram of the filter

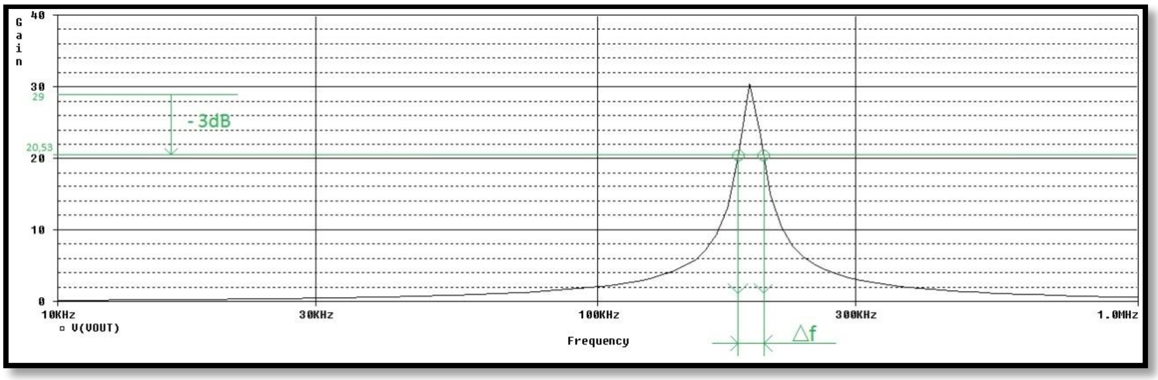
As seen in Picture 42, the filter amplifies the 200 kHz signals and frequencies close to this and attenuates the rest. The peak in the center frequency has a gain (Am) approximately 29 as it is shown in the previous calculations. The center frequency is not exactly 200 kHz since the calculated components are not exactly the standardized ones. In addition, the Picture shows how selective the filter is. Apart from that, there is another way to find the quality of the filter: the bandwidth. The bandwidth is the difference between the frequencies whose their attenuations through the filter are less than 3 decibels (dB, it is a logarithmic scale used in amplifications) compared with the central frequency peak. In the filters, the smaller the bandwidth is, the better it is. The calculations to know the approximated -3dB are the followings:

Knowing that the gain in the peak is 29 (Am=29):

$$-3dB = 20\cdot\log\left(\frac{x}{29}\right) \Rightarrow -3dB = 20\cdot\log x - 20\cdot\log 29 \Rightarrow -3 = 20\cdot\log x - 29{,}247 \Rightarrow$$
$$\Rightarrow 26{,}247 = 20\cdot\log x \Rightarrow 1{,}312 = \log x \Rightarrow x = 10^{1{,}312} = 20{,}53$$

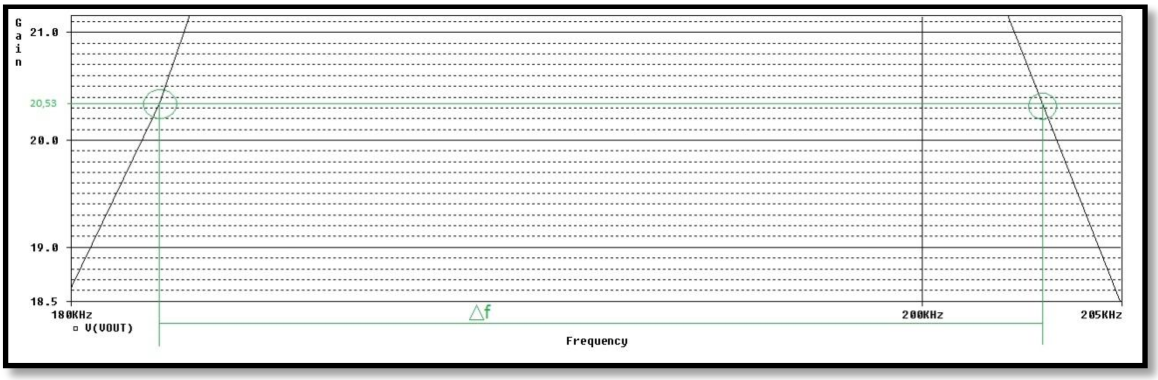
Equation 1 Calculations to get the -3dB

There is one step left to get the approximated value of the simulation's bandwidth. As you could see, in the next Picture 43 there are some draws that help to explain this bandwidth. Knowing the -3dB gain value of the filter, therefore we have to find the 2 points which cross with the filter's response curve in this value. These points belong to a determined frequency and they are the limits of the bandwidth.




Picture 43 Simulation's bandwidth

Taking a zoom of these points it is possible to see better the bandwidth:



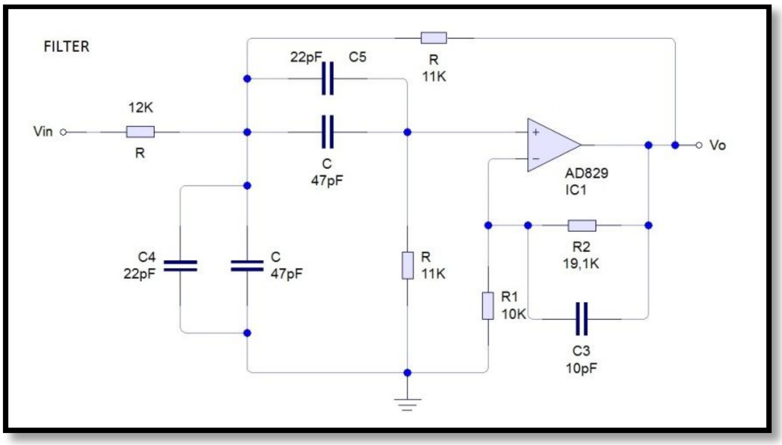

Picture 44 Zoom of the simulation's bandwidth

As seen, more or less the bandwidth (BW) of the simulation is:

$$\Delta f \approx 205 - 180 = 25kHz = BW$$

The result of this BW is ideal and in the reality it is almost impossible to have this short bandwidth for the reasons explained in the next paragraph.

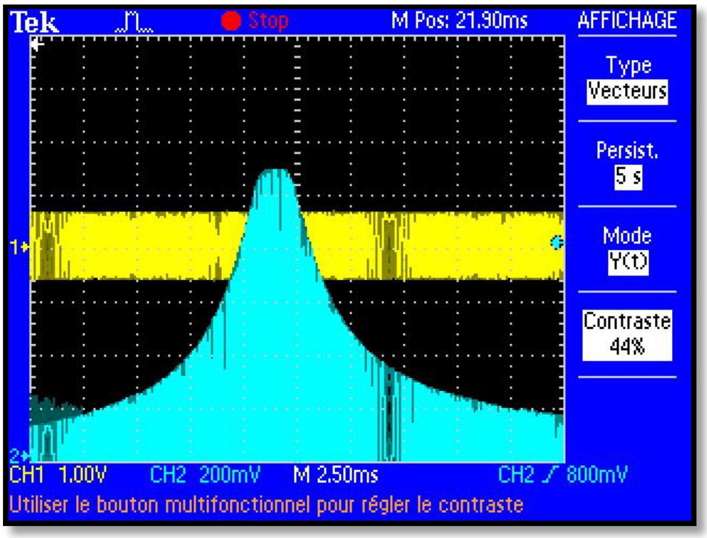
Once the filter is designed and simulated it is time to build in an evaluation board and test it. Due to the fact that the components are not ideal, they have tolerances, and parasite elements, the results after building the circuit were not the expected. Thanks to Mr. Dixneuf we could manage it and eventually we had a good implementation and results. The circuit was slightly modified the final schematic was the following:




Picture 45 Final filter schematic


Picture 46 Filter implementation in an evaluation board

When the filter was built, the last part consists of checking it using an oscilloscope and a signal generator. The test was about a sweep in a big range of frequencies using the generator and the oscilloscope had to read the filter's output.



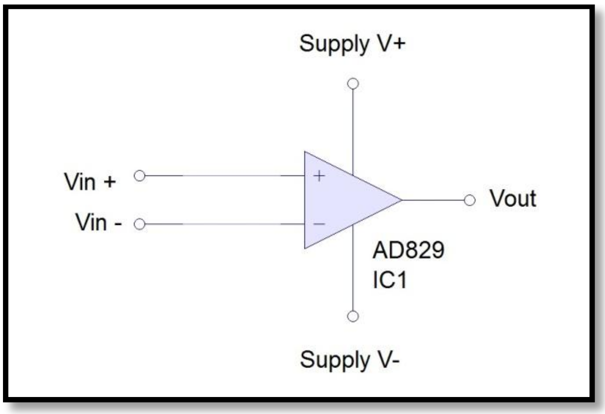

**Picture 47 Filter's curve using the oscilloscope**

As it is shown in Picture 47, the blue signal belongs to a selective filter output. The 200 kHz signal is amplified but the rest are attenuated, so it proves the good status of the filter.

### *Trigger*

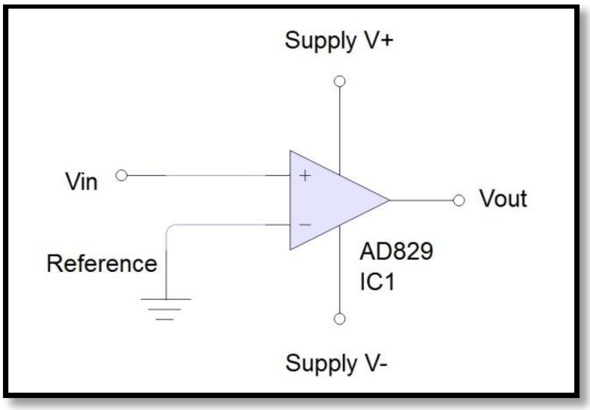
Once the receiver gets the selective and correct signal, a circuit is need to adapt it to the microcontroller. The input signal of the microcontroller must be as perfectly square as possible. The output of the filter is a 200 kHz signal, but not a square shaped.

First of all, it is needed to talk about the operational amplifiers. In Picture 48 you could see the basic schematic about an operational amplifier. The important fact for our application to be known is that if the positive input (Vin+) is higher than the negative one (Vin-), the output signal (Vout) is approximately the same as the positive supply (Supply V+). Otherwise, if the positive input (Vin+) is lower than the negative one (Vin-), the output signal (Vout) is approximately the same as the negative supply (Supply V-).



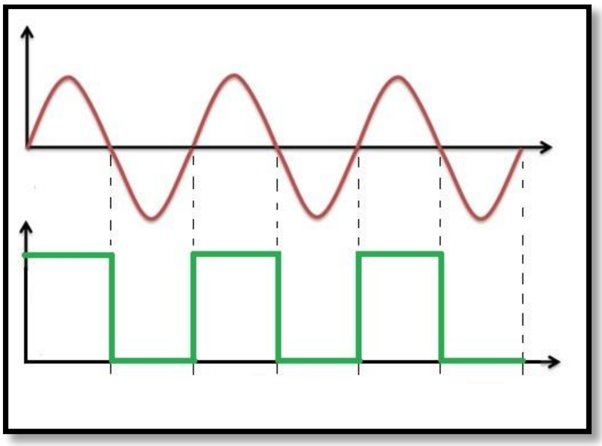

Picture 48 Operational amplifier’s schematic

The negative input in the amplifier (Vin- ) usually is the ground, so it is the reference:




Picture 49 Operational amplifier’ schematic

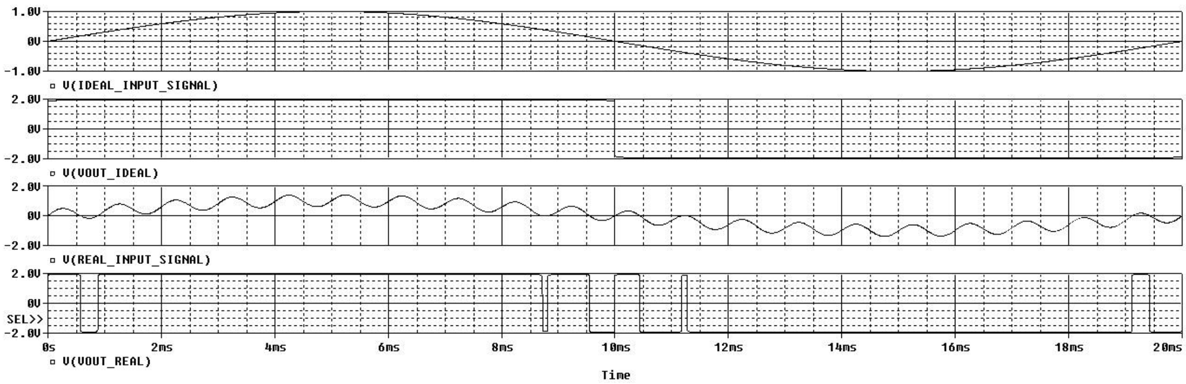
So when the Vin signal of the Picture 49 (Vin+) is positive, the output is positive and it has almost the same value as the positive supply, and when Vin is negative, the output is negative and it has almost the same value as the negative supply. When this happens, the operational amplifier is a comparator and the way it works is called saturation mode. In Picture 50 the input is an AC signal (sinusoidal) as explained earlier, when the input is positive the output saturates positively and when it is negative, negatively.



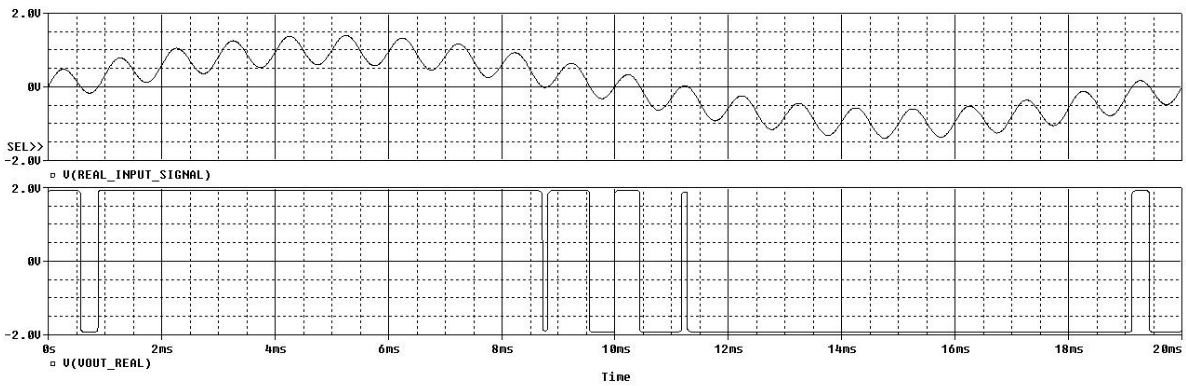
Picture 50 Possible trigger input signal and desired output signal

It is also seen in this previous picture that whatever the input signal is, the output is square because it can be only Vsupply+ and Vsupply-, so it is the desired wave signal for the microcontroller.

The problem comes when the signal is not clean enough and there is some ripple added in the signal, and when it passes by 0, some unaccepted values appear:

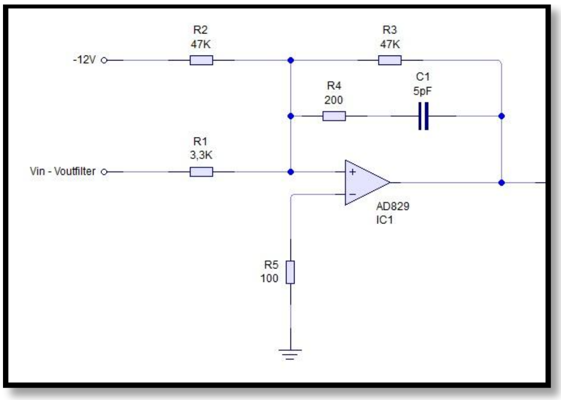



Picture 51 Ideal input, ideal output, rippled input, unacceptable output



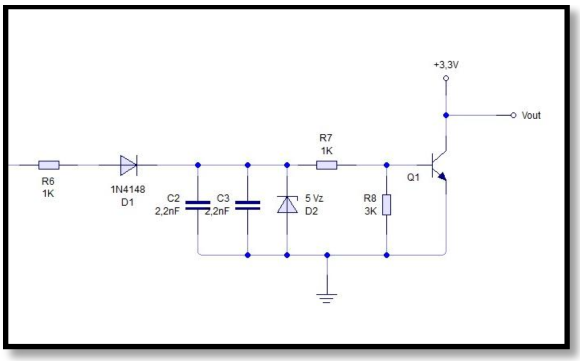

Picture 52 Undesired crosses to reference input and output signals

As shown in the previous pictures, the ripple signal could add some undesired squares in the output that could be understandable for the microcontroller, so the operational amplifier in saturated mode must be adapt to avoid this problem. The way to solve this problem is seen in the next picture, just adding passive components such as resistors and capacitors:




Picture 53 Ripple solving schematic

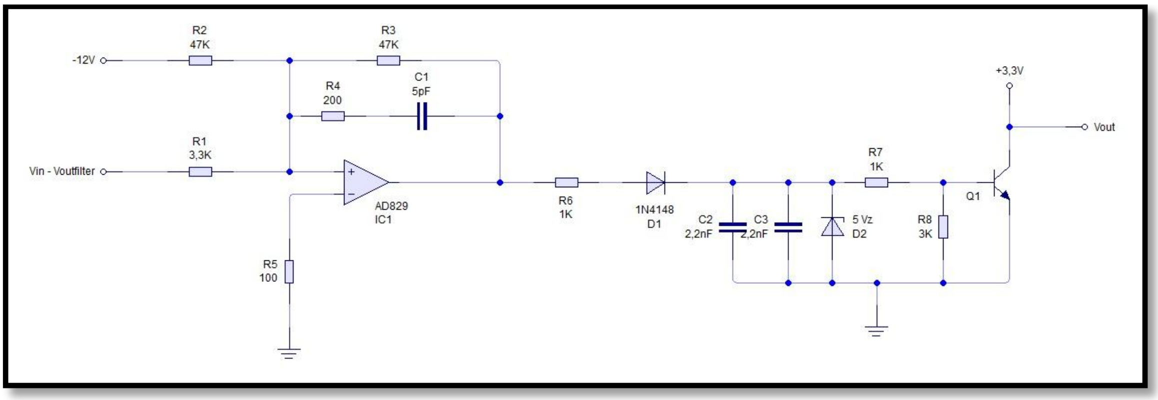
The receiver can only read signals that are 500 µs and 1ms long, and with 200 kHz the signal is much shorter, so the last part is to adapt the signal to be understood by the slavecontroller and it is possible using this circuit:




Picture 54 Circuit for the signal adaptation to the microcontroller

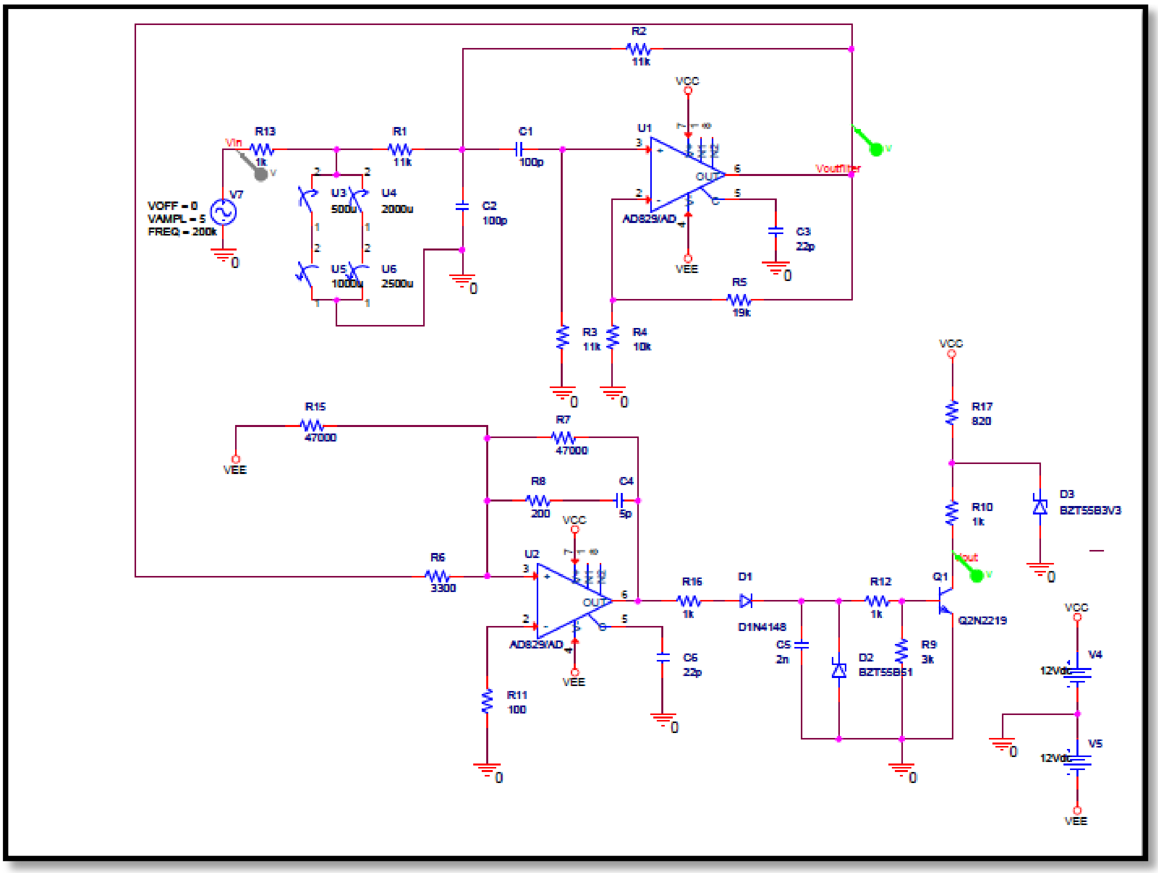
The diode rectifies the signal, the capacitors are signal followers, the zener diode avoids the high voltage, the resistors form a voltage divider and the transistor is a switch that turns on when the tension in the base is more than 0,6V or turns off when this tension is less. Thus the output of the circuit is almost 0 or 3,3V.

The circuit which carries out all these tasks is called trigger.



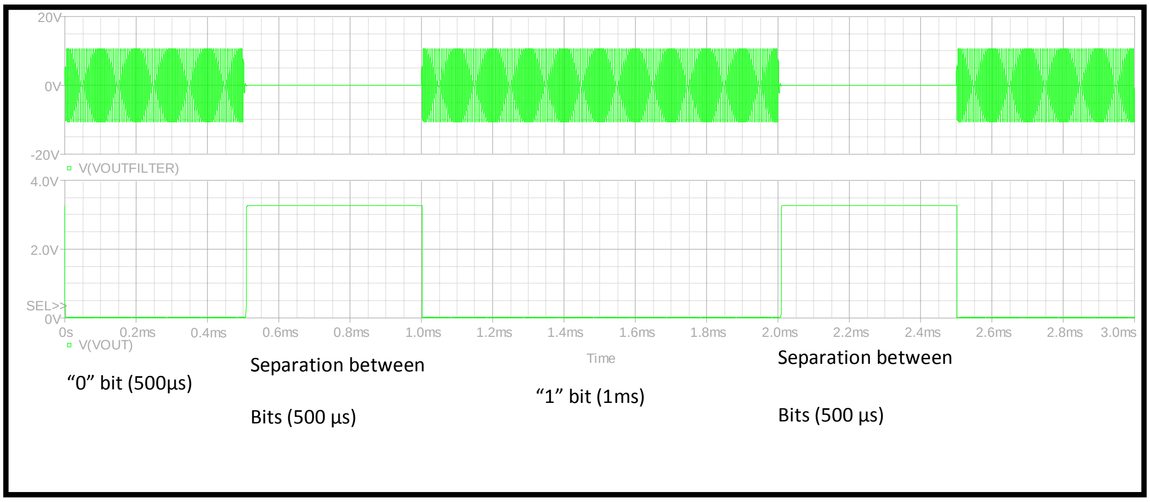

Picture 55 Trigger schematic

Afterwards, some simulations with PSpice© were done, and the results are the following pictures. The trigger is simulated with the filter, all together:



Picture 56 PSpice filter and trigger schematic

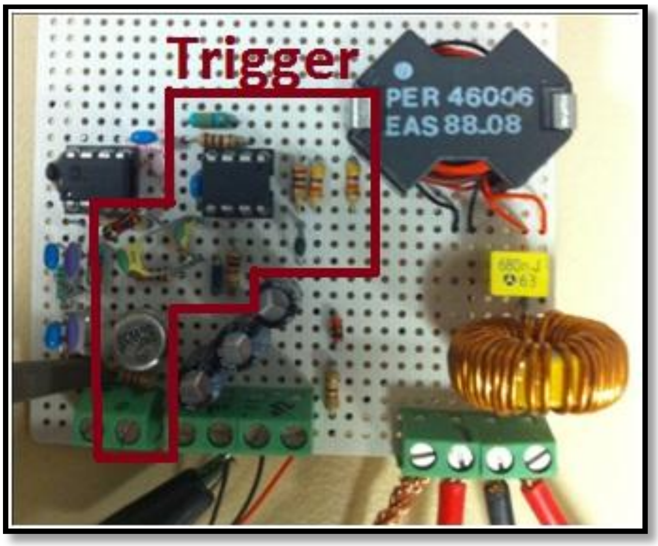
In the Picture 56, there are some switches that simulate the mastercontroller sending “0” during 500µs (at t=0), then a separation between bit during 500µs as well (at t=500µs), after that a “1” during 1ms (at t=1,5ms) and once again the separation (at t=2ms).




**Picture 57 Simulation’s result: Signal trigger IN & OUT**

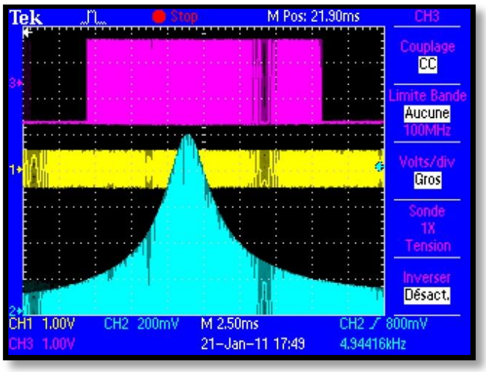
On the one hand, as you could see in the last picture, in the output of the filter (input of the trigger) there is a 200 kHz square signal when the transmitter sends information (“0” during 500µs or “1” during 1ms) or nothing (during 500µs) when the transmitter sends the separation between information bits. On the other hand, in the output of the trigger there are only the clean, rectified and adapted square signals with no high frequency.

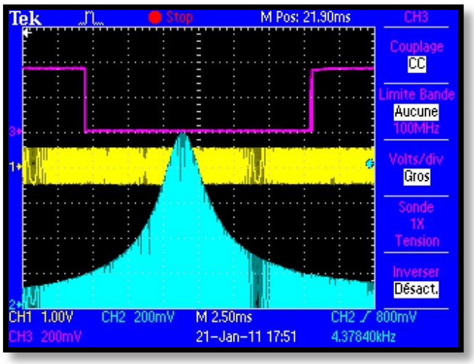
The implementation of this circuit in an evaluation board is the following:



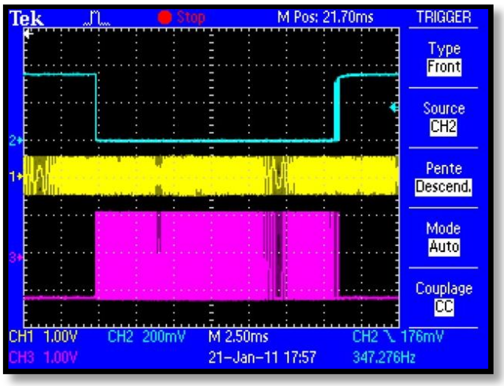

Picture 58 Trigger implementation.

The final part consists of testing the built circuit. It also contains the filter signals. The screen savings of the used oscilloscope are the next:



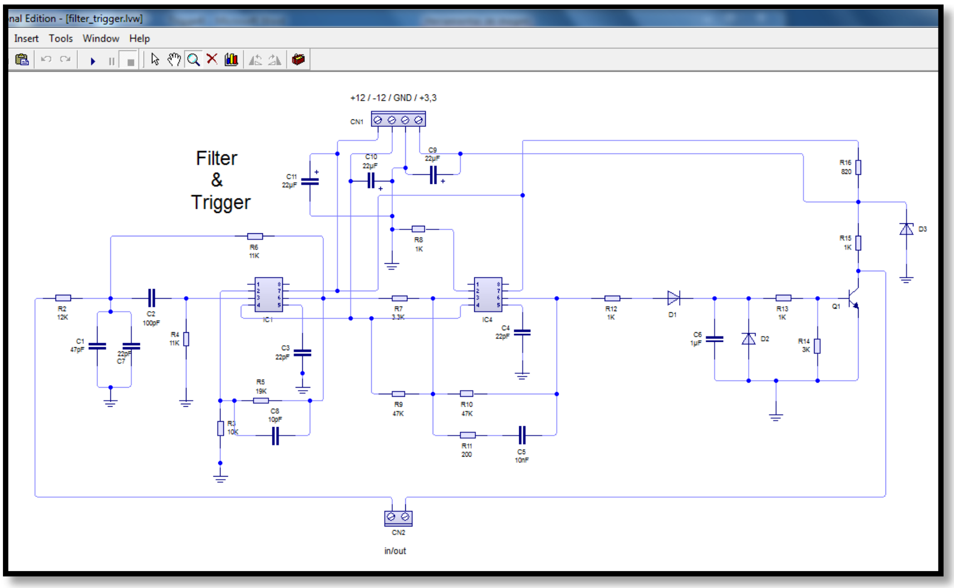




Picture 59 200 kHz Filter's output signal (pink) and filter's bode(blue) Picture 60 Output trigger's signal (pink)

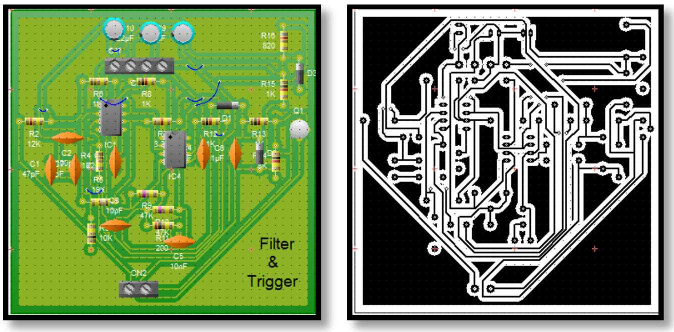



Picture 61 Both filter's output (Pink) and trigger's output(blue) signals

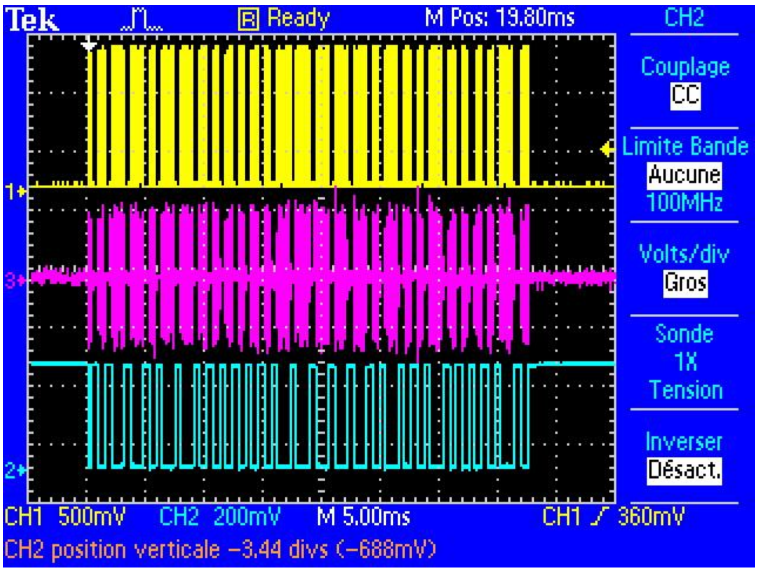
Using Livewire© and PCBWizard© we could have the layout to do a PCB of this part:




Picture 62 Livewire schematic



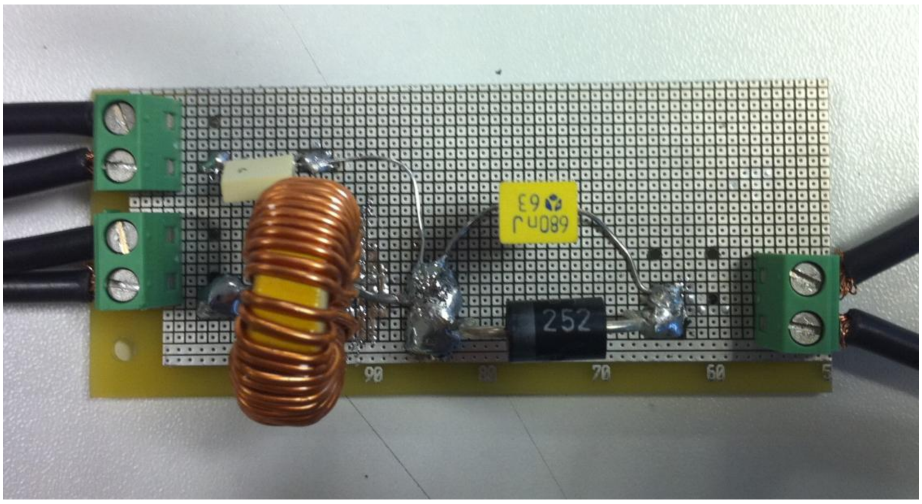

Picture 63 2D layout and artwork layout



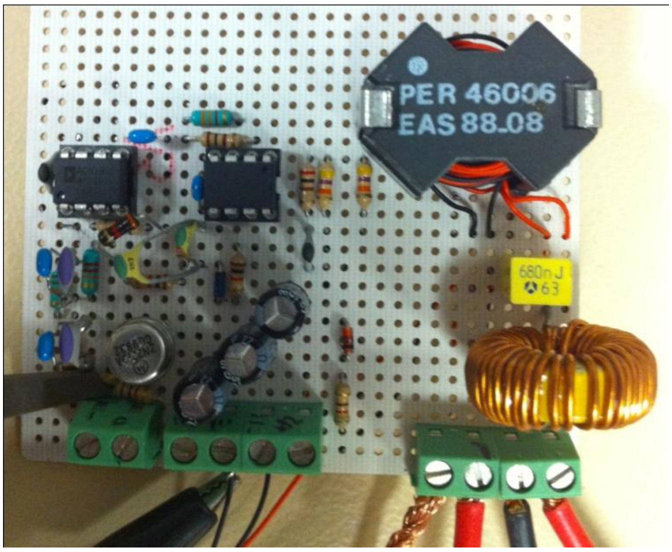

Picture 64 In this picture the yellow waveform is the sending signal of 170kHZ decoded by time. The pink colored is the receiving signal and the blue is the bit signal after the filtering.

#### 3.4.3.3 Protecting the solar filed

To protect that solar structure another simply circuit is also necessary. These circuits erase the high frequency signal. The high frequency signal is not desired in the panel or in the regulator. The signal is only desired between the panel and regulator so after transmitting and receiving, the signal is erased from the power line.




Picture 65 In this circuit an inductor is used to avoid high frequency to reach the regulator and a diode in case of reversed electricity.

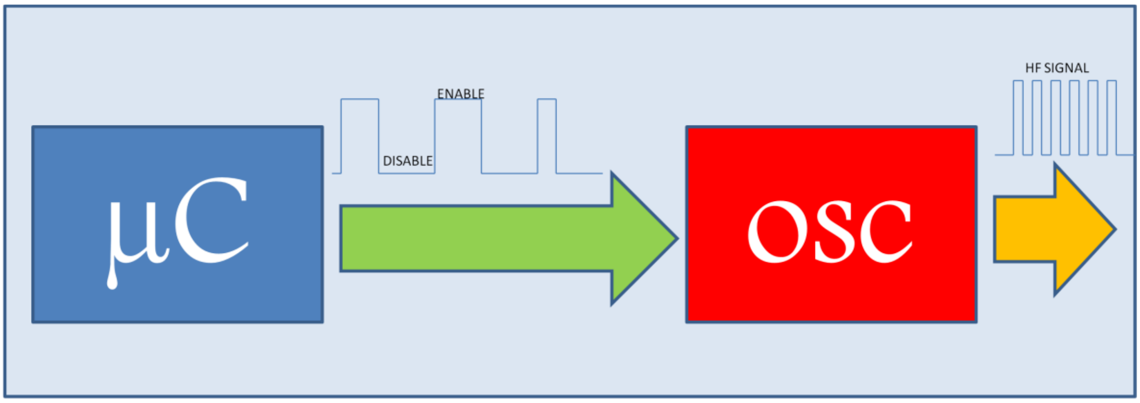



Picture 66 A similar inductor is used to protect the solar panel as you can see in this photo from the receiver.

Appendix 14 Schematic and pictures.

### 3.4.4 Master controller

The master controller is composed of two blocks. The first one is the microcontroller whose controls the transmission of data. The second one is the oscillator whose generates high frequency signal. In the Picture 67 is shown the transmitter block diagram.




Picture 67 The transmitter block diagram

When the microcontroller wants to send a bit, enables the oscillator for a time. The time that the oscillator is activated depends on what kind of bit is sent. Below is shown a table (**¡Error! No se encuentra el origen de la referencia.**) with the different bits and their times.

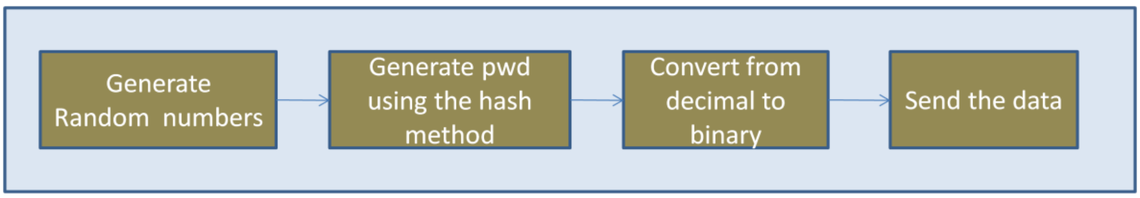
| | |
|---|---|
| Start bit | 340 µs |
| Bit=0 | 500 µs |
| Bit=1 | 1000 µs |

Table 1 Different bits and their times

The separation between bits it's always the same, 500 µs.

#### 3.4.4.1 Microcontroller

The microcontroller has to perform the following functions: generate six random numbers between 0 and 7, generate the password using the hash method (see page 63), converting from decimal to binary number and send the data (enable or disable the oscillator). Each function is executed by the code program as shown in Picture 68.



Picture 68 Sequence of functions execution

#### 3.4.4.2 Communication protocol

A communication protocol is used to send data to the receiver who can understand the meaning of each bit. The protocol is composed of one start bit, 27 data bits and one parity bit as it's shown in Picture 69.

| | | |
|---|---|---|
| Start bit | Data (27 bits) | Parity bit |

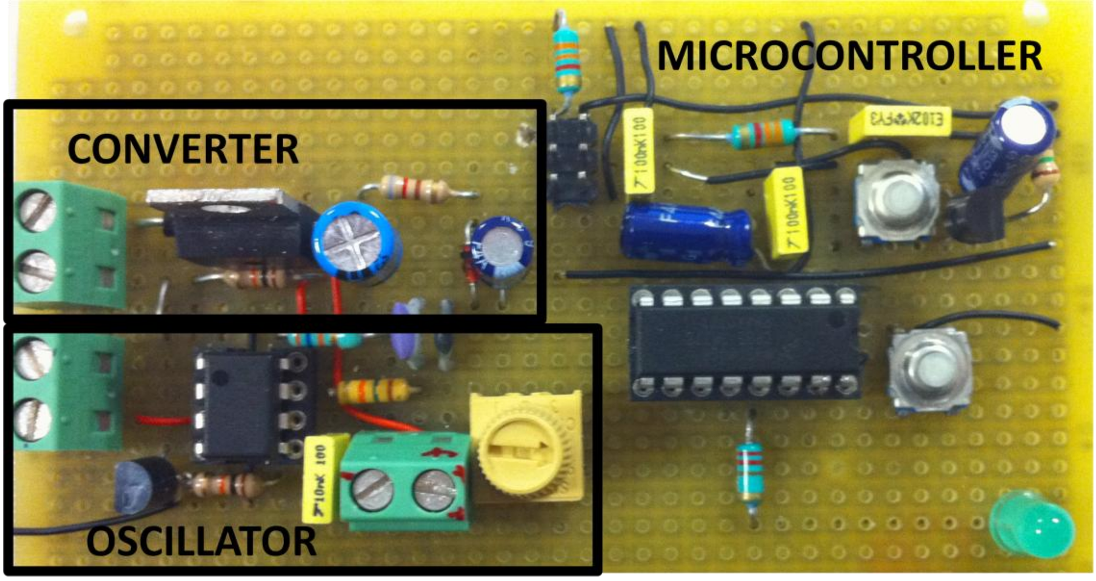
Picture 69 Data frame

The parity bit is a bit to check if the communication has been successful. To calculate the parity bit it is necessary to do the sum of all bits of data. The parity bit will be zero if the result of the sum is an even number and it will be one if the number is an odd number.

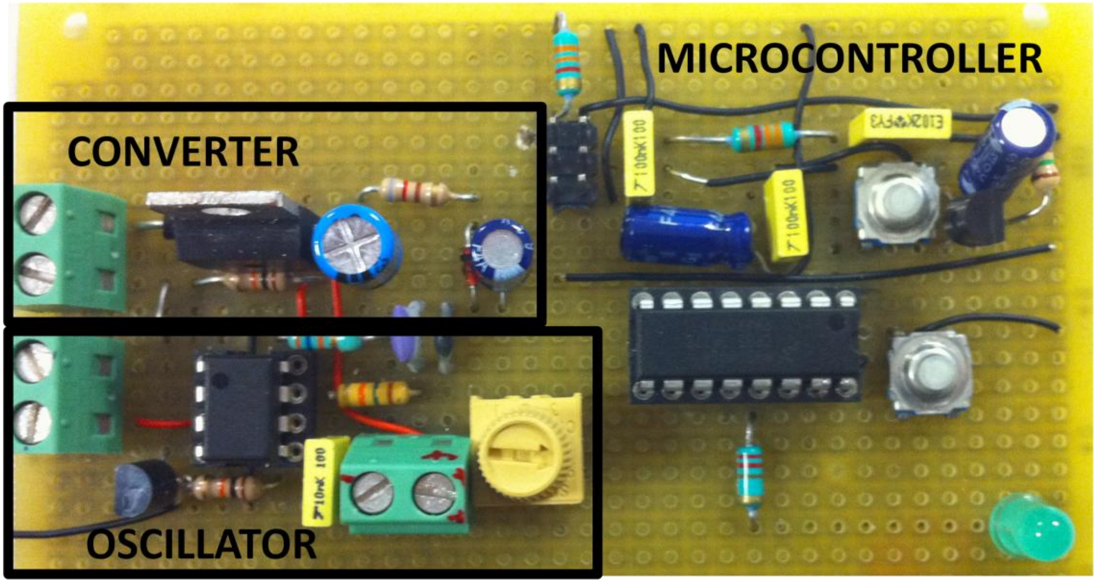
#### 3.4.4.3 Oscillator

The function of the oscillator is to generate a high frequency (HF) signal. This signal is sent when the ENABLE signal is activated. The frequency of HF signal is 200 kHz. To reach this value a 555 chip from Texas Instruments used and it is required to make some calculations to set the proper value. These calculations can be read in the 6.10 appendix "oscillator calculations".

### 3.4.4.4 Prototype and experimental results

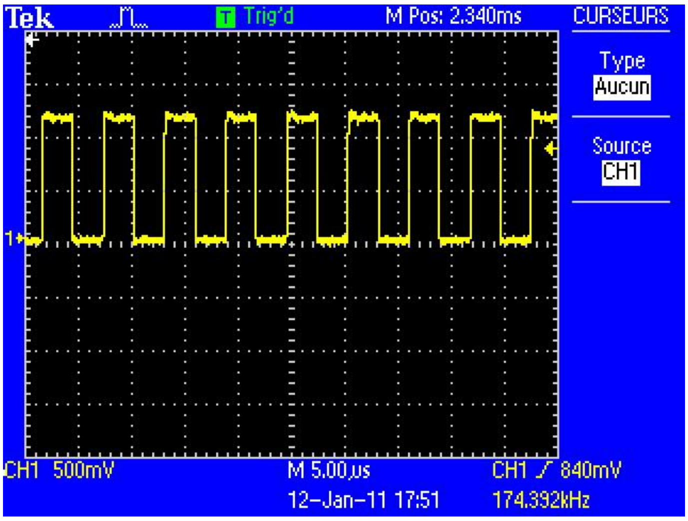



Picture 70 you can see the transmitter prototype. In this board is possible to distinguish three parts. The first one is the converter area whose mission is to adapt the voltage to supply the different electronic device. The second one is the microcontroller area and the last one is the oscillator area.



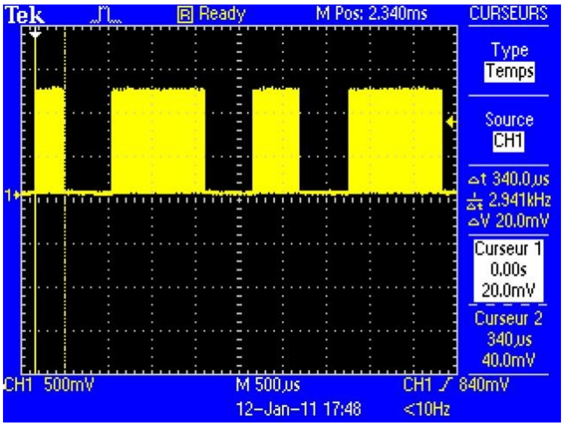

Picture 70 Transmitter prototype

The following picture (Picture 71) shows the high frequency signal. As you can see the signal is perfectly symmetrical and its frequency is around 200 kHz.

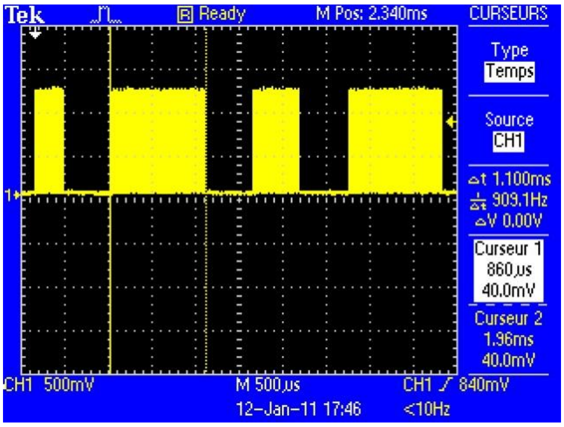



Picture 71 High Frequency signal

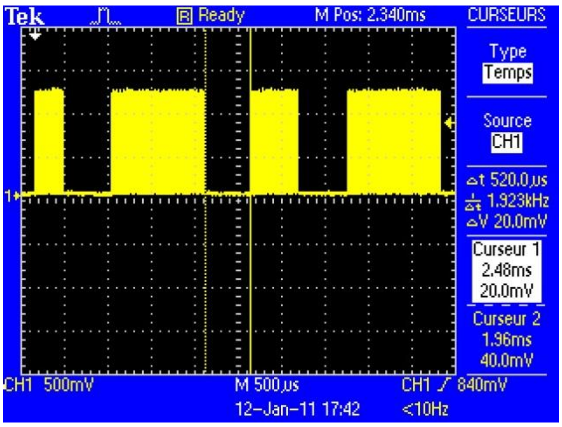
From Picture 72 to Picture 75, it can be seen as experimental results are the same as the theoretical results in table 1.



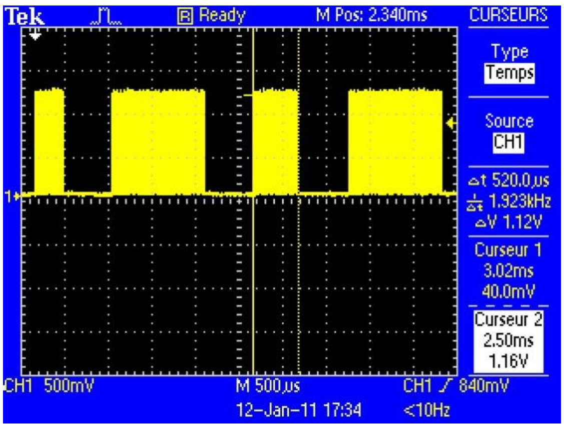

Picture 72 Start bit measure




Picture 73 Bit 1 measure



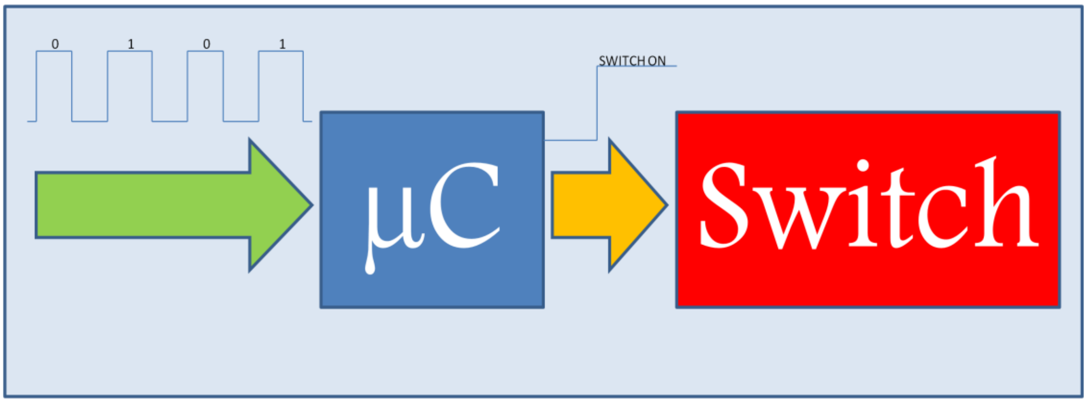

Picture 74 Separation measure




Picture 75 Bit 0 measure

### 3.4.5 The slave controller

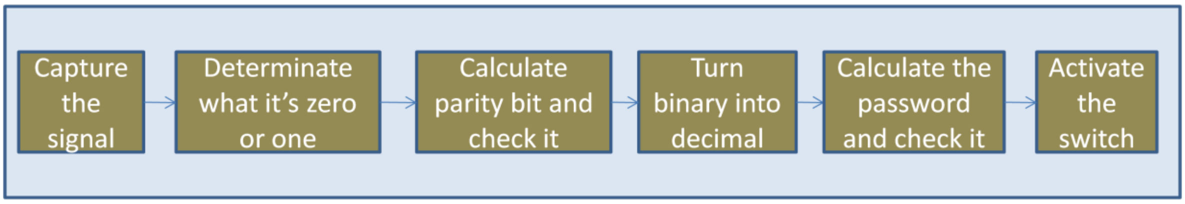
The slave controller is composed of two blocks, just like the master controller. The first one is the microcontroller whose capture the signal and check it. The second one is the switch which is activated by microcontroller. In the Picture 76 is shown the receiver block diagram.




Picture 76 The receiver block diagram

#### 3.4.5.1 Microcontroller

The microcontroller has to perform the following functions: capture the input signal and measure the time between events, determinate what it's a zero or one, calculate the parity bit and check the parity, turn binary numbers into decimal numbers, calculate the password and check it, activate the switch. Each function is executed by the code program as shown in Picture 77.

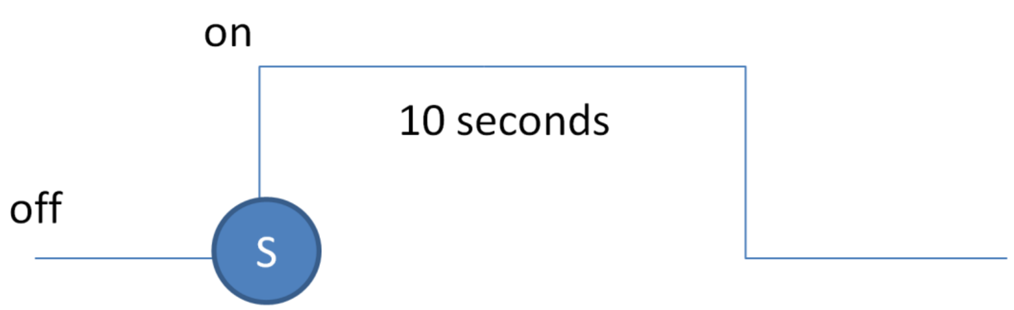



Picture 77 Sequence of functions execution in the slave controller

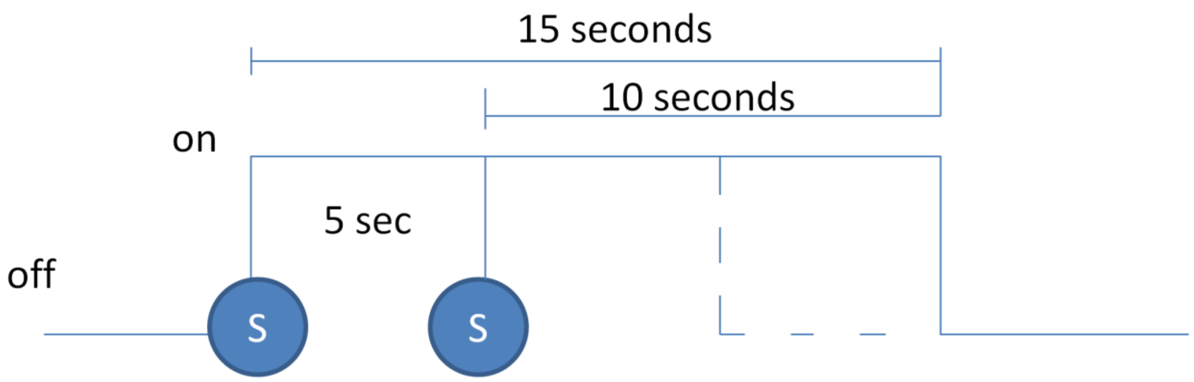
### 3.4.6 Checking function

One problem would be if the thieves cut the wires and continuously have light on the panel. The Mosfet that control the current would always be switched on. The reason for this is because the microcontroller attached to the panel would continuously have an electrical supply.

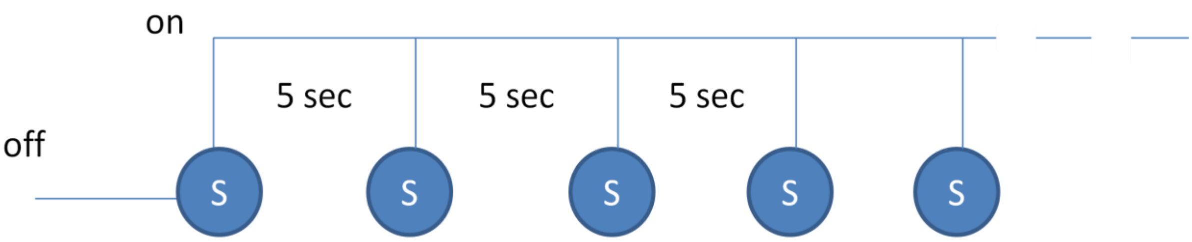
To avoid this unlikely but possible situation there is a function inside the microcontrollers. The controller on the recipient needs a hasched signal to make the panel start operating. The microcontroller on the recipient needs a new signal in the time lap of ten seconds after the first signal is sent otherwise it switches off the panel. The function on the microcontroller on the sender is continually sending a new signal every 5 seconds.




**Picture 78 show the Mosfet who controlling if the panel is operating or not. And this time there is only one signal and the Mosfet switches off after ten seconds. The big S is a sign for signal.**




**Picture 79 show the Mosfet who controlling if the panel is operating or not. And this time there is two sent signals. The second one is sent 5 seconds after the first one. The big S is a sign for signal.**




**Picture 80 show the Mosfet who controlling if the panel is operating or not. This time there is a signal every 5 second from the transceiver and this is how the function in the prototype is operating. The big S is a sign for signal.**

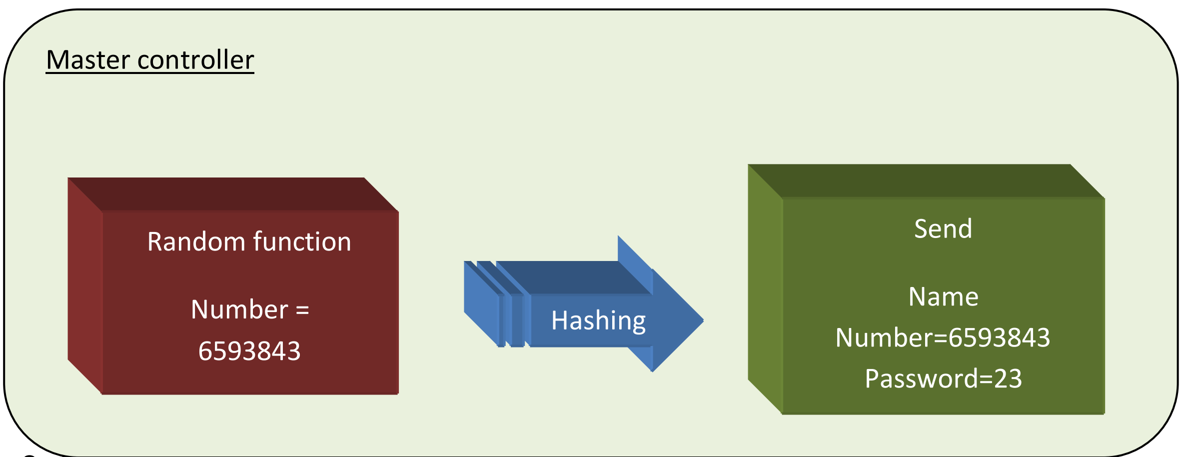
The microcontroller on the recipient unit has two counters. One of the counters is needed to receive the signal and hasch the signal. The other counter is counting the time from when the first signal is received and if the counter reaches 10 seconds the Mosfet controlling the panel is switched off.

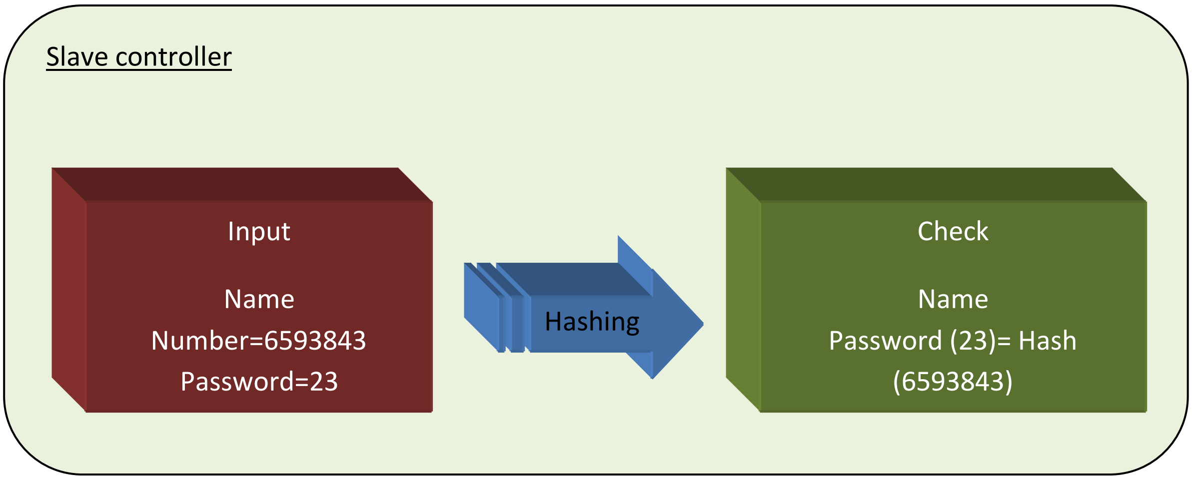
By adding this function in the system the panel is useless after maximum 10 seconds after stolen.

### 3.4.7 The hash method

For the needs of these projects a hash method will be used to generate and cryptograph the signal sent from the master controller to the slave controller.

Random number function generates a number which is the input of our hash method. The hash method uses this number to generate a new password. After that the master controller sends a package of information to the slave that contains the name of the controller, the number and the password. The slave checks the name and after that use the same hash method to generate again the number. If the output is the same with the password it operates the panel.






**Hash method of the prototype**

Using this method we achieved:

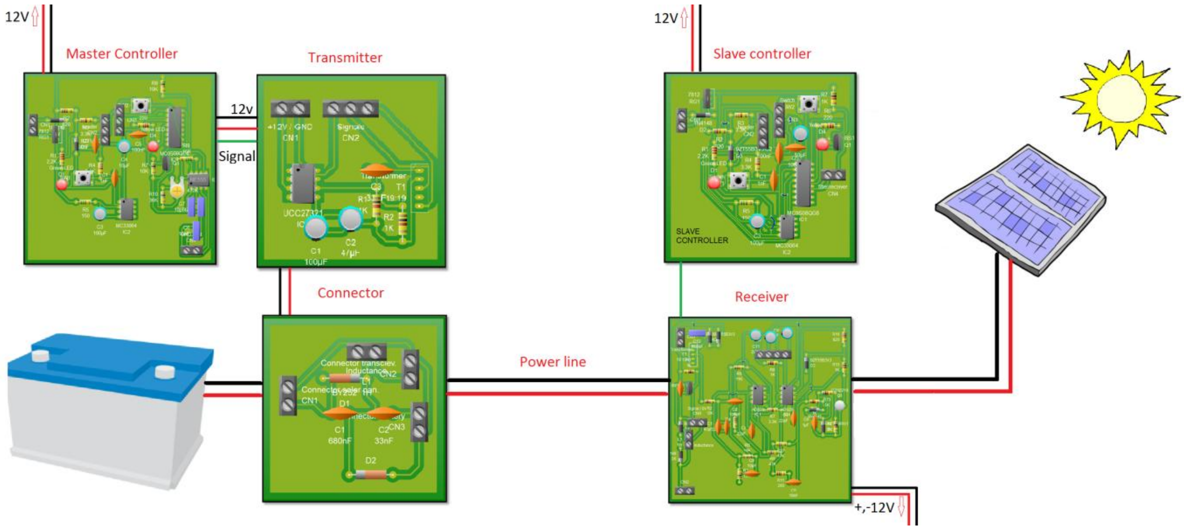
- ✓ A generated new password every time.
- ✓ A Cryptographed signal that differences every time.

By this way the anti-thief system actually checks only if the master control has the same hash method with the slave control. The password is used to send only the results of the hash which are enough to check if the method is the same but not enough to copy the method. So is impossible to copy this system.

### 3.4.8 How are the circuits connected

All the circuits are finally connected by the way shown in Picture 81 to be tested. The 12 voltage supply was created by a DC power source, see Appendix 12 Used equipment. The length between the connecter (Protector) and receiver was 24 meters.

Picture 81




Picture 81 Show how the circuits were connected during the tests

# 4. Further development

In this section you will find some components which are necessary components that need to be added to the prototype before it is completely ready to be manufactured.

The prototype uses a switch but it is not the optimal one so there is one suggestion later in the report under 4.1.

The input voltage to the prototype is 12 voltages but when the solution is operating at a solar field the input would be varying. So the final solution need a converter to continuously make sure that the module has the right voltage. Under 4.2 there is a suggestion of a converter.

The resin that will be necessary to attach the module on the panel is not finally decided. You can read more about it under 4.3.

## 4.1 Switch

The mission of switch is enabling or disabling the solar panel. So that the switch can perform its function well, it must be able to withstand 200 v 10 A. The chosen solution is the Mosfet because it is able to meet the requirements and it consumes little power. The model of Mosfet is SUP57N20-33 and its main characteristics are showed in the Table 2.

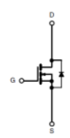
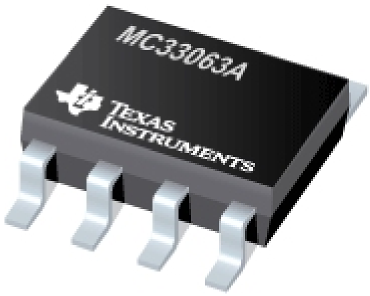
<table>
<tr><th colspan="4">Mosfet characteristics</th></tr>
<tr><td rowspan="5"></td><td>$V_{ds\_max}$</td><td>200 V</td><td rowspan="5"></td></tr>
<tr><td>$I_{D\_max}$</td><td>57 A</td></tr>
<tr><td>$R_{ds(on)}$</td><td>0.033 Ω</td></tr>
<tr><td>$V_{GS(th)}$</td><td>2 to 4 V</td></tr>
<tr><td>Temperature Range</td><td>- 55 to 175 ºC</td></tr>
<tr><td>SUP57N20-33</td><td colspan="2"></td><td>N-channel Mosfet</td></tr>
</table>

**Table 2.-** Mosfet characteristics

## 4.2 Converter

The energy, for the different devices, is suplied by the MC33063A (Picture 82). This device allows to obtain 3.3 volts at the output with a input voltage between 3 and 40 volts.




Picture 82 Shows converter MC33063A

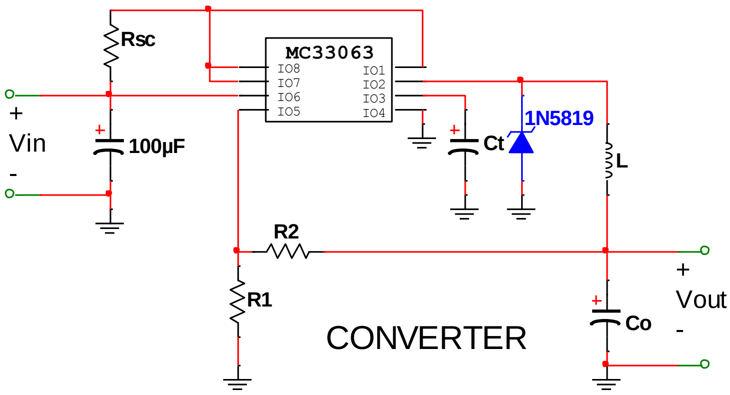
On the other hand, it is necessary to calculate some parameters for the MC33063A to make it work properly. The ecuations to calculate the different parameters are showed in Appendix 8 Converter's parameters.

Acording to ecuations of Table 4 in Appendix 8 Converter's parameters, it is easy to calculate the parameters for differents voltages values. The follow table (Table 3) shows the parameters for diferent input voltage.

| $V_{in}$(V) | d | $T_{off}$ ( µs) | $T_{on}$ ( µs) | $C_T$ (pF) | $R_{sc}$(Ω) | $L_{min}$ (µH) | $C_o$(µF) | R1/R2 | R2(kΩ) |
|---|---|---|---|---|---|---|---|---|---|
| 12 | 0.54 | 19.7 | 10.6 | 425 | 0.3 | 81.7 | 37.9 | 1,64 | 1,804 |
| 15 | 0.39 | 21.8 | 8.47 | 339 | 0.3 | 90.6 | 37.9 | 1,64 | 1,804 |
| 20 | 0.26 | 24 | 6.34 | 253 | 0.3 | 99.5 | 37.9 | 1,64 | 1,804 |
| 25 | 0.20 | 25.2 | 5.06 | 202 | 0.3 | 104.76 | 37.9 | 1,64 | 1,804 |
| 30 | 0.16 | 26.1 | 4.22 | 169 | 0.3 | 108.27 | 37.9 | 1,64 | 1,804 |
| 35 | 0.14 | 26.7 | 3.60 | 144 | 0.3 | 110.78 | 37.9 | 1,64 | 1,804 |
| 40 | 0.12 | 27.1 | 3.16 | 126 | 0.3 | 112.66 | 37.9 | 1,64 | 1,804 |
| *Constants* | $V_{out}$=3.3V; $f$=33kHz ; $V_{ripple}$=0.1V ; R1=1.1 kΩ ; $I_{out}$=0.5A; $V_{sat}$=1 V; $V_F$= 0.85V | | | | | | | | |

Table 3 Parameters of converter

To conclude this section, the Picture 83 displays the schematic of converter.




Picture 83 Shows the schematic of the converter

## 4.3 The resin

Resin will be used to protect the system. It was decided to use resin from the 3M Company that seems to be specified in these applications. The resin was a huge catalog of version depends of the materials and the applications. After research we finally defined that the possible materials are:

- Ethylene vinyl acetate - EVA
- White poly-vinyl-ester
- Tedlar – Polyvinyl fluoride - PVF

We tried to get a sample of the back sheet of the panel in and test in order to find out which of these materials it is made of. Unfortunately we could not do a planned glass temperature test in order to decide the material. The reason was that we did not have any samples to test. Another reason was because of the time problems that were faced, see management (New planning), the group decided to focus on the electrical aspects.

# 5. Conclusion

The students have been working on an EPS project in ENIT, Tarbes France, during 4 months. They goals of the EPS were to learn how to manage a project and develop further experience in electronics.

## 5.1 Management

The students have gained experience in management. They learned how to start, visualize, plan, work in and close a project. It is one thing to read about management but it is very hard to apply the know-how in reality. This is something the students have experienced during the project. One example is when the planning was so incorrect and a complete new planning was made so the students completed the new planning much faster than before because of the new experience and the knowledge in MS project. This is a proof of greater understanding in management.

In the beginning it is hard to understand why it’s necessary to use management and it is more an obstacle than a helping hand. After a while you can see the benefits of having good management like keeping track of time, optimize the work and understand the current situation.

The things that could be better for the next project is to get the important document sign earlier by the supervisors, even though it had to be changes in the documents many times. The requirement for example was never signed. When the final decision of how to work and final requirement we started working straight away and to get the requirement document signed after the prototype was half finished seemed a bit unnecessary.

The students know a lot more about the purpose of management and the tools used in management so they think that the goals of EPS concerning experience in management is achieved.

## 5.2 Electronics

The students have different backgrounds and different engineering skills, thus the level of electronic knowledge was and still is very varied.

The electronic circuits in the prototype are advanced and the code that is controlling the prototype is very complex.

The different circuits

- Master controller
- Slave controller
- Filter
- Oscillator
- Cleaner (remove high frequency signal)

Because of the complexity, everyone in the group had a chance to develop their electronic skills.

One of the objectives was to build a working prototype as a deliverable. Under the management chapter you can find the specifics about the deliverables.

The goal for this particular deliverable was to

- At the end of the project we are going to deliver a prototype.
    - ➢ The minimum is to develop a prototype functional for a standard panel with a short distance between the master and the slave microcontroller.
    - ➢ The maximum is to develop a prototype functional for all different panels in a huge range of distances between the master and slave microcontroller.

Our prototype is functional for one input of 12 voltages and simulates a standard type of panel. The distance between the transmitter and receiver is 24 meters and the prototype has all the requirements that Solarcom has asked for. Therefore we consider that the prototype live up to more than the minimum level and therefore a success.

The one thing that was planned but never accomplished was the choice of resin and testing the resin. We never received any samples of the back sheet of the panel or an answer from Solarcom concerning the material on the back sheet. With this in mind we still think the results are good and the project landed stable and with good results. It will be very easy for

Solarcom to get in contact with a company, for example 3M, and ask for the best resin for this application.

The goal of the project was to earn deeper understanding in electronics therefore is the goal of the project in a technical point of view achieved.

## 5.3 What can be reused

The way of sending information can be used in all applications where you have to send information through power line.

The length of tasks and the actual length are good to save to another project. To be a good project manager you need to have experience and be able to recognize similarities from previous task, and in that way, now how long a task will take.

The problems we faced and the solution of the problems is always good to store. If a similar problem is faced it might help you to come to a solution faster and more important, chose the right one.

# 6. Appendix

## 6.1 Appendix 1 Rules of behavior

§ If you're late you pay the cake! If you're late you'll need to buy something to the next meeting. If there is more than one person late then the latest person needs to pay!

§ Respect! Respect everyone and their opinion.

§ Round discussion! Before we make a discussion everyone must speak and tell their opinion and thoughts about the discussion.

§ Keep the timetable! If we plan to do a task, do it on time!

§ Honesty! If there is a problem in the group don't be afraid to speak up.

# General rules

§ We shall switch the head of the meeting after every meeting. But if a person doesn't like to be the head he doesn't need to be the head of a meeting.

§ Before a meeting we start to define the topics that would like to discuss. That will be our agenda. Then we have a meeting about the topics.

____________________ ____________________

Agis Kothalis Anders Ågren

____________________ ____________________

Ivan Tellez Vispe Carlos Andrés López-Peiáez

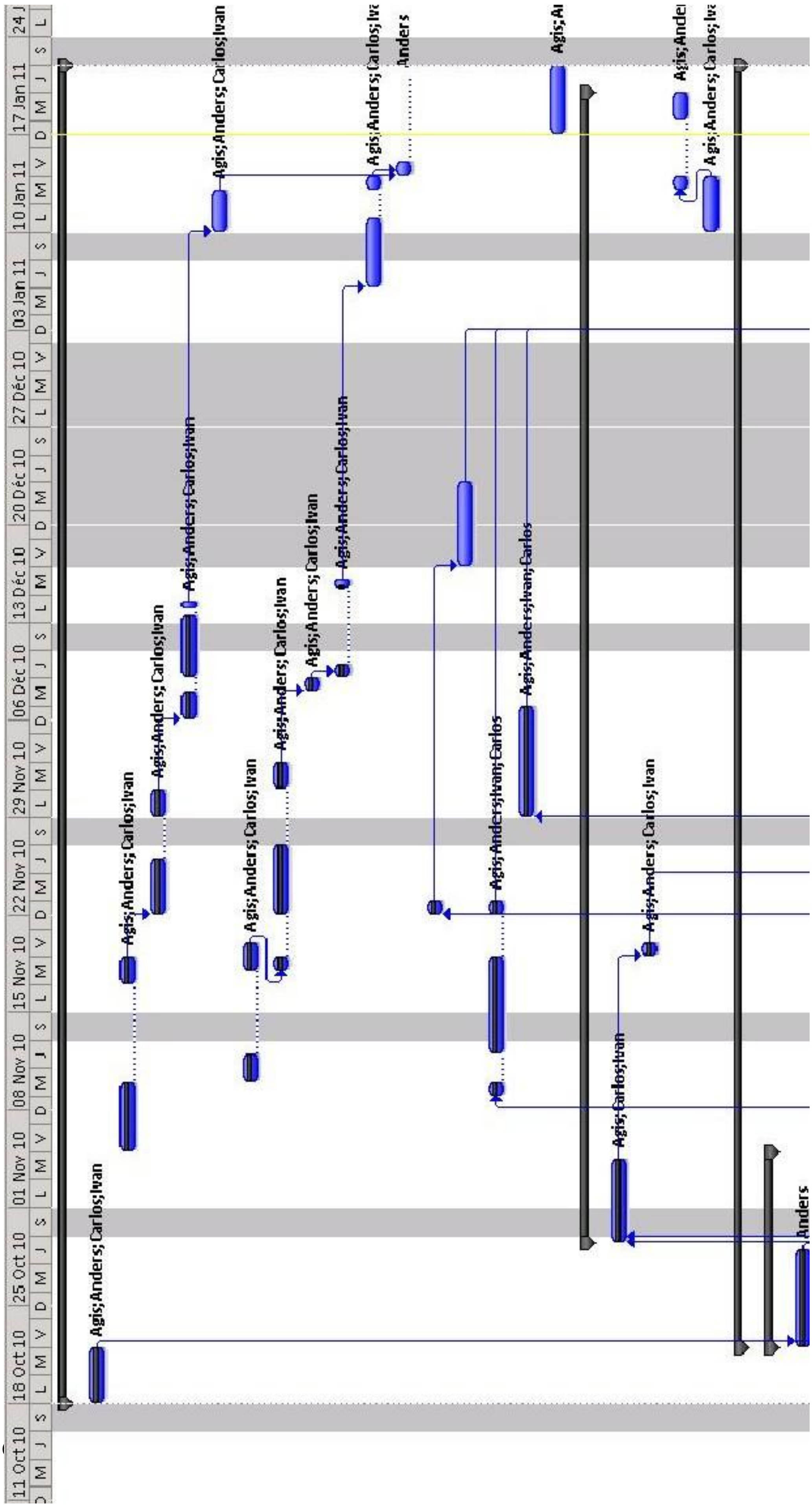
Friday, November 26, 2010

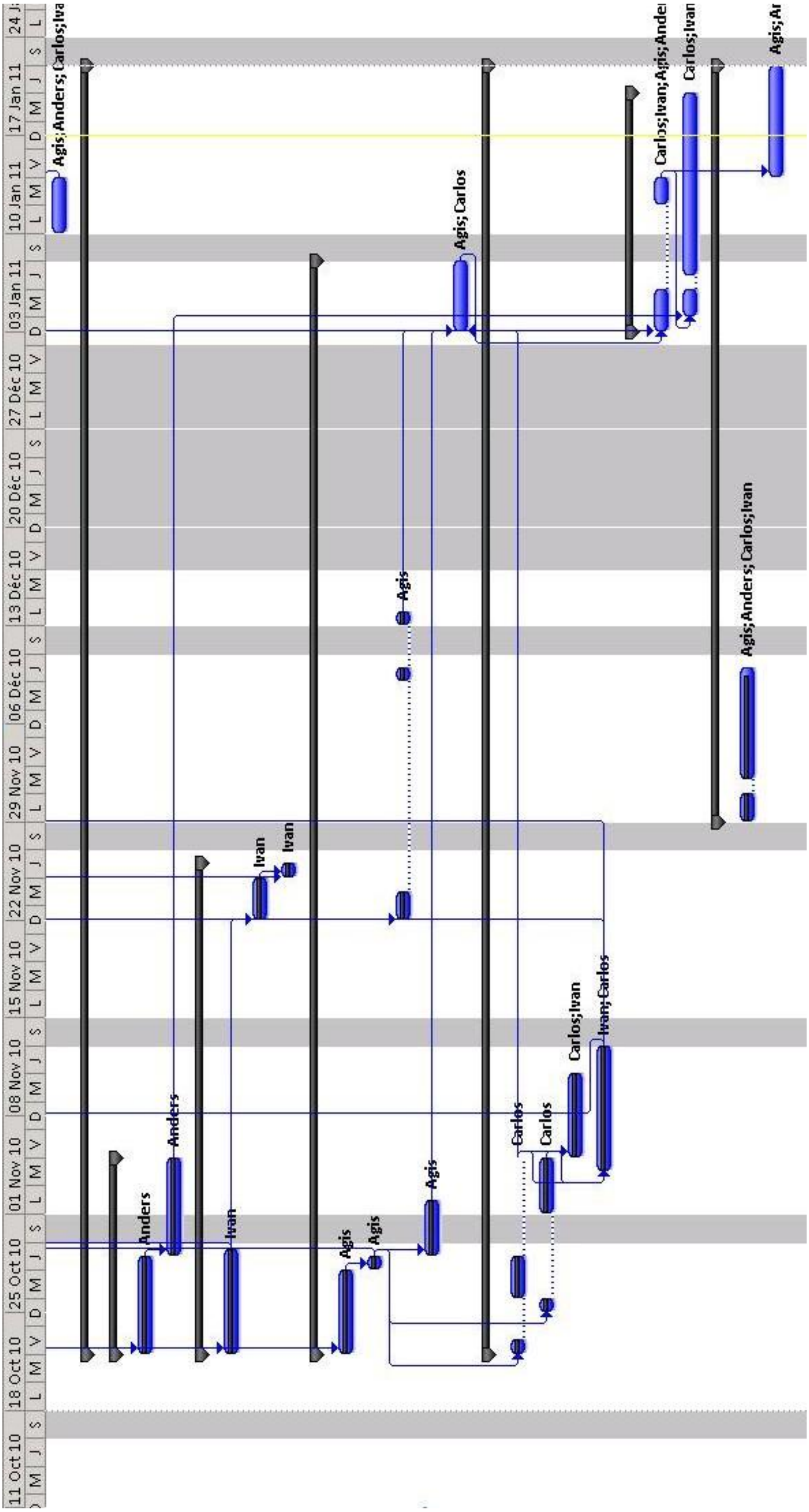
## 6.2 Appendix 2 List of tasks

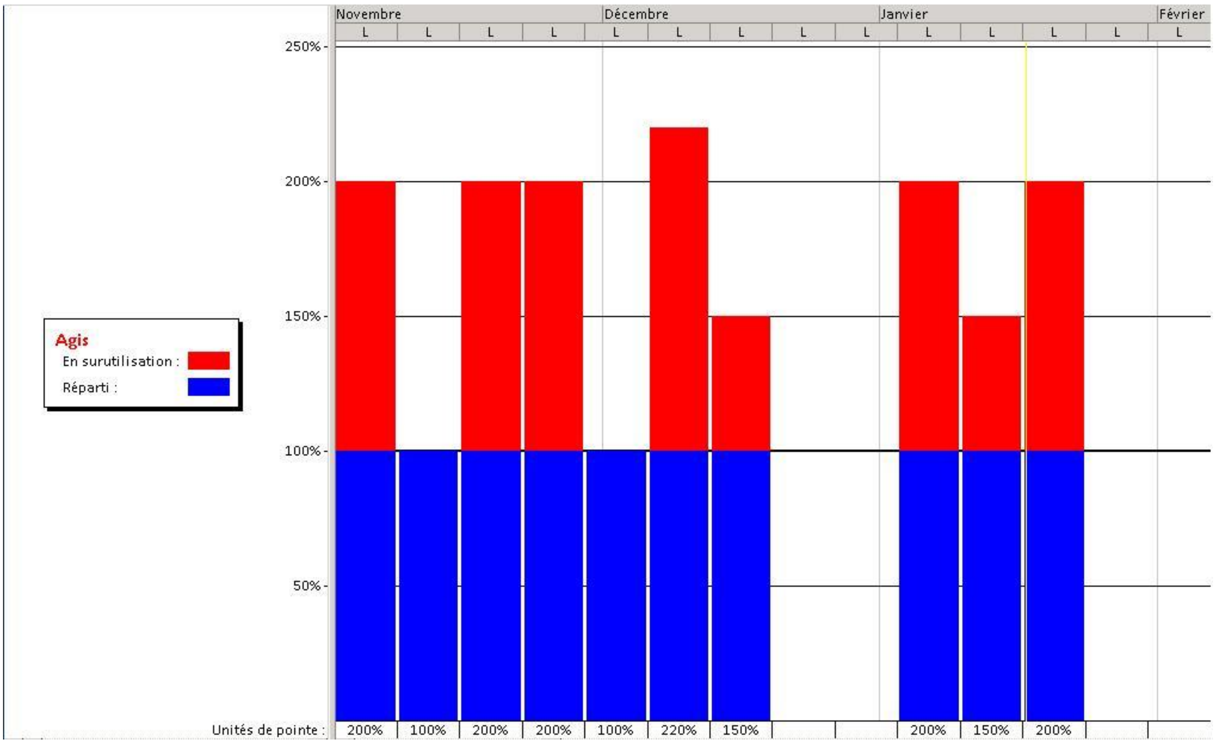
| ID | TASKS | FIXED WORK | RESOURCES | Responsibilities |
|---|---|---|---|---|
| A | Reports | | | |
| A.1 | Intermediate report | 6H | Agis & Anders & Carlos & Ivan | |
| A.2 | End project report | 18H | Agis & Anders & Carlos & Ivan | |
| C | Solarcom report | | | |
| C.1 | Matrix of solutions | 4H | Agis 100% Carlos & Ivan 50% | Agis |
| C.2 | Meeting | 3H | Agis & Anders & Carlos & Ivan | Ivan |
| C.3 | Prepare final docs | 30H | Agis & Anders & Carlos & Ivan | Anders |
| C.4 | Further research | | Agis & Anders & Carlos & Ivan | Carlos |
| D | Management | | | |
| D1 | Launch of project | 18H | Agis & Anders & Carlos & Ivan | Agis |
| D2 | Close the project | 12H | Agis & Anders & Carlos & Ivan | Anders |
| D3 | Management report 1 | 6H | Ivan & Carlos 33% Agis & anders 100% | Agis |
| D4 | Management report 2 | 6H | Agis & anders 33% Ivan & Carlos 100% | Anders |
| D5 | Management report 3 | 6H | Ivan & Carlos 33% Agis & anders 100% | Carlos |
| D6 | Management report 4 | 6H | Agis & anders 33% Ivan & Carlos 100% | Ivan |
| D7 | Technical report 1 | 6H | Ivan & Carlos 33% Agis & anders 100% | Agis |
| D8 | Technical report 2 | 6H | Agis & anders 33% Ivan & Carlos 100% | Carlos |
| D9 | Technical report 3 | 6H | Ivan & Carlos 33% Agis & anders 100% | Ivan |
| D10 | Technical report 4 | 6H | Agis & anders 33% Ivan & Carlos 100% | Agis |
| D11 | Technical report 5 | 6H | Ivan & Carlos 33% Agis & anders 100% | Carlos |

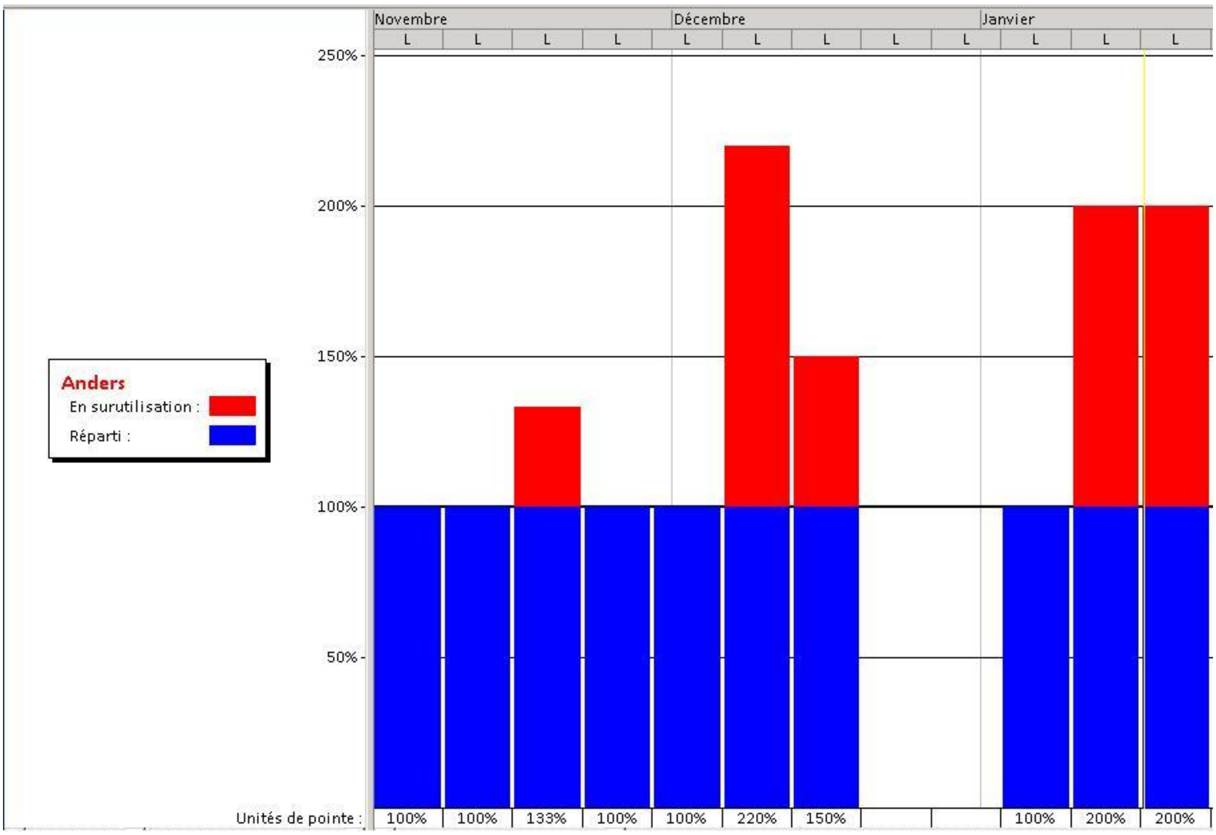
| | | | | |
|---|---|---|---|---|
| D12 | Assembling | 4H | Anders | Anders |
| D13 | Milestone 1 | 0 | | |
| D14 | Milestone 2 | 0 | | |
| D15 | Milestone 3 | 0 | | |
| D16 | Supply | 2 WEEKS | | |
| E | Prototype | | | |
| E.1 | Test | | | Carlos |
| E.1.1 | Asssembly | 30H | Agis & Anders & Carlos & Ivan | Carlos |
| E.1.2 | Simulate | 18H | Agis & Anders & Carlos & Ivan | Carlos |
| E.1.3 | Validation | | | Carlos |
| E.2 | Mech. Part | | | Anders |
| E.2.1 | Research | 18H | Anders | Anders |
| E.2.2 | Choose resin | 36H | Anders | Anders |
| E.3 | Commmunication | | | Ivan |
| E.3.1 | Research ways of communication | 30H | Ivan | Ivan |
| E.3.2 | Transcievers | 6H | Ivan | Ivan |
| E.3.3 | Recievers | 6H | Ivan | Ivan |
| E.4 | Digital signature | | | Agis |
| E.4.1 | Choose master | 18H | Agis | Agis |
| E.4.2 | Choose slave | 18H | Agis | Agis |
| E.4.3 | GPS / Password | 1H | Agis | Agis |
| E.4.4 | Algorithm | 6H | Agis | Agis |
| E.4.5 | Develop the code | 18H | Agis and carlos | Agis |
| E.5 | Circuit | | | Carlos |
| E.5.1 | Switch | 24H | Carlos | Carlos |
| E.5.2 | Converter | 24H | Carlos | Carlos |
| E.5.3 | Additional parts | 12H | Carlos & Ivan | Carlos |
| E.5.4 | Schematic | 6H | Carlos & Ivan | Carlos |

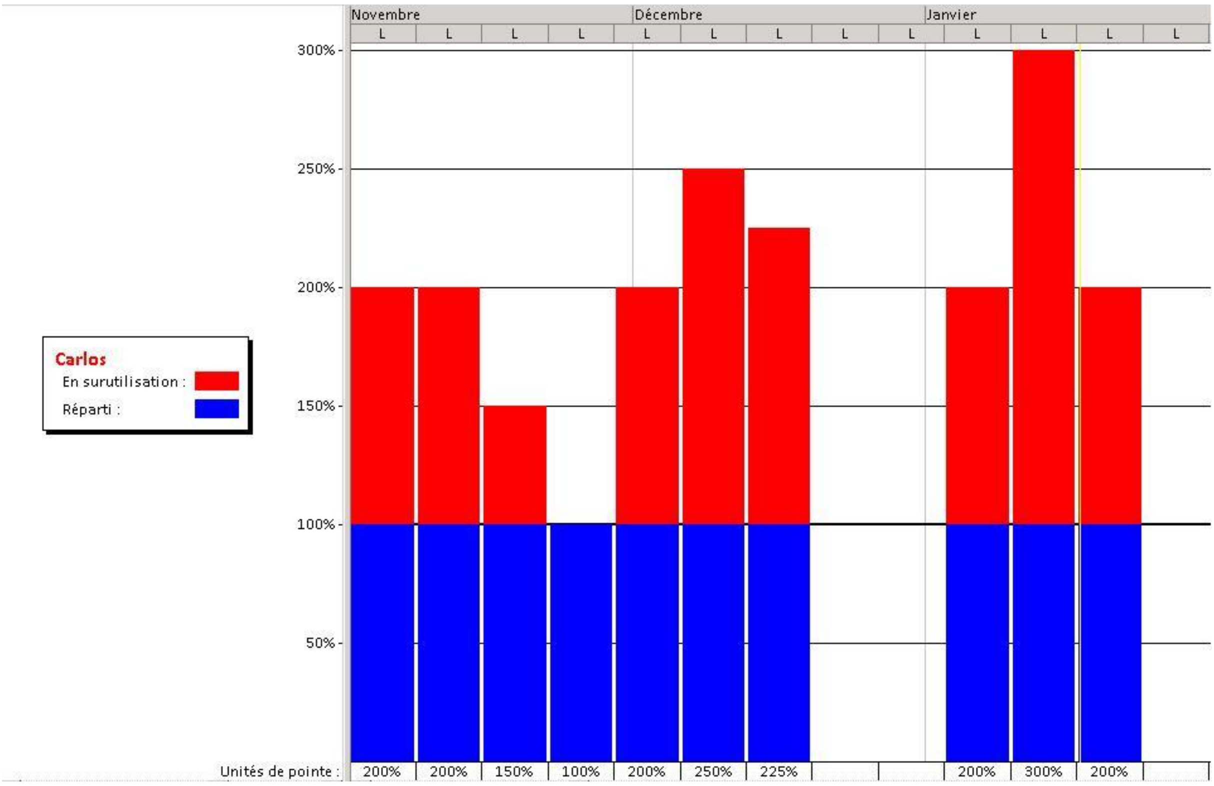
## 6.3 Appendix 3 Gantt chart and resources usage

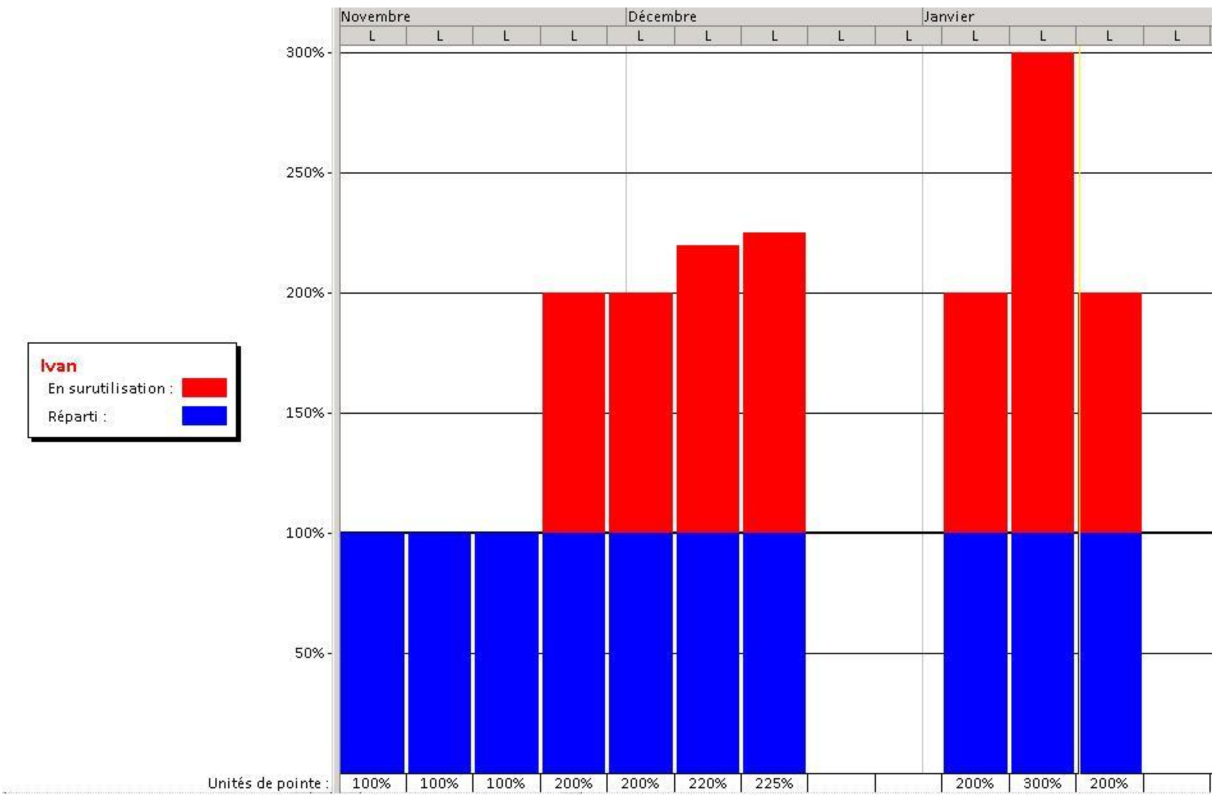



11 Oct 10
18 Oct 10
25 Oct 10
01 Nov 10
08 Nov 10
15 Nov 10
22 Nov 10
29 Nov 10
06 Déc 10
13 Déc 10
20 Déc 10
27 Déc 10
03 Jan 11
10 Jan 11
17 Jan 11
Anders
Ivan
Agis
Carlos
Carlos;Ivan
Ivan;Carlos
Agis;Carlos
Agis;Anders;Carlos;Ivan


Novembre
Décembre
Janvier
Février
L L L L L L L L L L L L L L
250%
200%
150%
100%
50%
Agis
En surutilisation :
Réparti :
Unités de pointe :
200% 100% 200% 200% 100% 220% 150% 200% 150% 200%


Novembre
Décembre
Janvier
L L L L L L L L L L L L
250%
200%
150%
100%
50%
Anders
En surutilisation :
Réparti :
Unités de pointe :
100% 100% 133% 100% 100% 220% 150% 100% 200% 200%


Novembre
Décembre
Janvier
L L L L L L L L L L L L L
300%
250%
200%
150%
100%
50%
Carlos
En surutilisation :
Réparti :
Unités de pointe :
200% 200% 150% 100% 200% 250% 225% 200% 300% 200%


Novembre
Décembre
Janvier
L L L L L L L L L L L L L
300%
250%
200%
150%
100%
50%
Ivan
En surutilisation :
Réparti :
Unités de pointe :
100% 100% 100% 200% 200% 220% 225% 200% 300% 200%

## 6.4 Appendix 4 Different uses of radio frequencies

| Frequencies in kHz | Allocated Purposes |
|---|---|
| 490–510 | Distress (telegraph) |
| 510–535 | Government |
| 535–1605 | AM radio |
| 1605–1750 | Land/mobile public safety |
| 1800–2000 | Amateur radio |
| **Frequencies in MHz** | **Allocated Purposes** |
| 26.96–27.23, 462.525–467.475 | Citizen band radios |
| 30.56–32, 33–34, 35–38, 39–40, 40.02–40.98, 41.015–46.6, 47–49.6, 72–73, 74.6–74.8, 75.2–76, 150.05–156.2475, 157.1875–161.575, 162.0125–173.4 | Private mobil radio (taxis, trucks, buses, railroads) |
| 220–222, 421–430, 451–454, 456–459, 460–512 | |
| 746–824, 851–869, 896–901, 935–940 | |
| 74.8–75.2, 108–137, 328.6–335.4, 960–1215, 1427–1525, 220–2290, 2310–2320, 2345–2390 | Aviation (communication and radar) |
| 162.0125–173.2 | Vehicle recovery (LoJack) |
| 50–54, 144–148, 216–220, 222–225, 420–450, 902–928, 1240–1300, 2300–2305, 2390–2450 | Amateur radio |
| 72–73, 75.2–76, 218–219 | Radio control (personal) |
| 54–72, 76–88, 174–216, 470–608 | Television broadcasting VHF and UHF |
| 88–99, 100–108 | FM radio broadcasting |
| 824–849 | Cellular telephones |
| 1850–1990 | Personal communications |
| 1910–1930, 2390–2400 | Personal comm. (unlicensed) |
| 1215–1240, 1350–1400, 1559–1610 | Global Positioning Systems (GPS) |
| **Frequencies in GHz** | **Allocated Purposes** |
| 0.216–0.220, 0.235–0.267, 0.4061–0.45, 0.902–0.928, 0.960–1.215, 1.215–2.229, 2.320–2.345, 2.360–2.390, 2.7–3.1, 3.1–3.7, 5.0–5.47, 5.6–5.925, 8.5–10, 10.0–10.45, 10.5–10.55, 13.25–13.75, 14–14.2, 15.4–16.6, 17.2–17.7, 24.05–24.45, 33.4–36, 45–46.9, 59–64, 66–71, 76–77, 92–100 | Radar, all types |
| 2.390–2.400 | LANs (unlicensed) |
| 2.40–2.4835 | Microwave ovens |
| 45.5–46.9, 76–77, 95–100, 134–142 | Vehicle, anticollision, navigation |
| 10.5–10.55, 24.05–24.25 | Police speed radar |
| 0.902–0.928, 2.4–2.5, 5.85–5.925 | Radio frequency identification (RFID) |
| 3.7–4.2, 11.7–12.2, 14.2–14.5, 17.7–18.8, 27.5–29.1, 29.25–30, 40.5–41.5, 49.2–50.2 | Geostationary satellites with fixed earth receivers |

Figure 1 RF Spectrum and allocated purposes. Source by:

http://zone.ni.com/devzone/cda/tut/p/id/3541#toc1

## 6.5 Appendix 5 Characteristics of microcontroller

| Rating | Symbol | Value | Unit |
|---|---|---|---|
| Supply voltage | $V_{DD}$ | –0.3 to +3.8 | V |
| Maximum current into $V_{DD}$ | $I_{DD}$ | 120 | mA |
| Digital input voltage | $V_{In}$ | –0.3 to $V_{DD}$ + 0.3 | V |
| Instantaneous maximum current<br>Single pin limit (applies to all port pins)[1, 2, 3] | $I_D$ | ± 25 | mA |
| Storage temperature range | $T_{stg}$ | –55 to 150 | °C |

Figure 2 shows the character of the microcontroller

| Rating | Symbol | Value | Unit |
|---|---|---|---|
| Operating temperature range (packaged) | $T_A$ | $T_L$ to $T_H$ | °C |
| C | | –40 to 85 | |
| M | | –40 to 125 | |

### Memory Options

- FLASH read/program/erase over full operating voltage and temperature
- MC9S08QG8 — 8 Kbytes FLASH, 512 bytes RAM

Figure 3 Memory options

### Input/Output

- 12 general-purpose input/output (I/O) pins, one input-only pin and one output-only pin; outputs 10 mA each, 60 mA max for package
- Software selectable pullups on ports when used as input
- Software selectable slew rate control and drive strength on ports when used as output
- Internal pullup on $\overline{\text{RESET}}$ and $\overline{\text{IRQ}}$ pins to reduce customer system cost

Figure 4 Inputs/Outputs

### Clock Source Options

- **ICS** — Internal clock source module containing a frequency-locked-loop (FLL) controlled by internal or external reference; precision trimming of internal reference allows 0.2% resolution and 2% deviation over temperature and voltage; supports bus frequencies from 1 MHz to 10 MHz
- **XOSC** — Low-power oscillator module with software selectable crystal or ceramic resonator range, 31.25 kHz to 38.4 kHz or 1 MHz to 16 MHz, and supports external clock source input up to 20 MHz

Figure 5 Clock source options

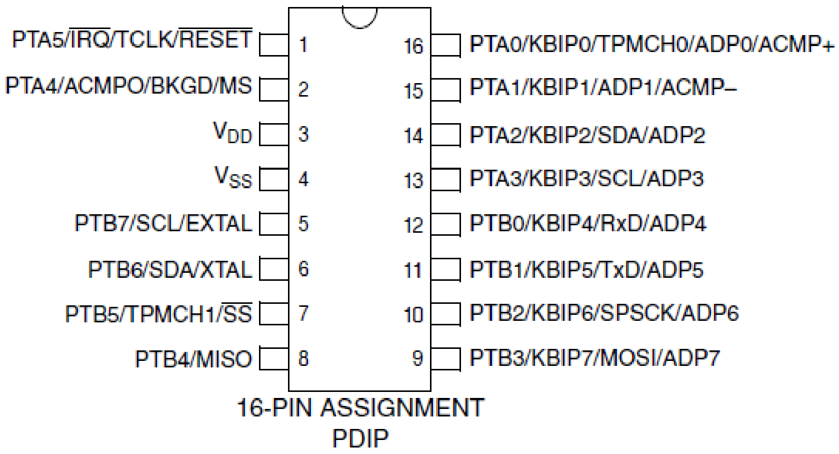



Figure 6 Device Pin Assignment

## 6.6 Appendix 6 Comparing Table

| Comparing chart | | | | | | |
|---|---|---|---|---|---|---|
| | advantages | disadvantages | Cost ($) | Install | Protection | Consumption |
| **Gps** | | | | | | |
| **On the panel** | Even if the robber found the gps there is no way to use it<br>Fewer components minimize the chances of failure | need extra wire<br>Communication with the satelite | 15 per panel | med | **** | 28,36 w |
| **Off the panel - RF** | Even if the robber found the gps there is no way to use it<br>no need of extra wire | It is possible to copy the signal<br>Simillar to master controller solution<br>more components more chances of failure | 14,5 per panel | easy + | **** | 45,45 w |
| **Master controll** | | | | | | |
| **In the ground** | less expensive<br>Fewer components minimize the chances of failure<br>It is possible to use the same system in different solar field | Unpossible to change the master controll if there is a failure<br>need more extra wires | 12 per panel | med | **** | 31,92 w |
| **Off the ground - RF** | It is possible to use the same system in different solar field<br>no need of extra wire | more components more chances of failure<br>Ii is possible to copy the signal | 14,3 per panel | med | **** | 45,45 w |
| **Off the ground -WIFI** | Safer communication | Need to develop a new network everytime<br>system restart<br>Possible communication issues | - | diff+ | ***** | 37,69 w |
| **Off the ground -PLC** | It is possible to use the same system in different solar field | Difficult to develop a stable system for different solar<br>The signal depends of the powerline current and voltage | - | diff++ | **** | 44,72 w |

## 6.7 Appendix 7 Table of choice

| Method | cost | consumption | Instaling | Protection | Utility | Final |
|---|---|---|---|---|---|---|
| weight pointer | 0,1 | 0,3 | 0,3 | 0,1 | 0,2 | |
| **Master controll Off the ground – RF** | 8 | 7 | 10 | 8 | 10 | 8,7 |
| **GPS On the panel** | 8 | 10 | 7 | 9 | 9 | 8,6 |
| **Master controll In the groud** | 10 | 9 | 5 | 8 | 4 | 6,8 |
| **Master controll Off the ground –PLC** | 8 | 7 | 7 | 8 | 8 | 7,4 |
| **GPS Off the panel - RF** | 8 | 7 | 8 | 9 | 9 | 8 |
| **Master controll Off the ground –WIFI** | 7 | 8 | 4 | 10 | 7 | 6,7 |

| Method | Cost | Cost (6 panel per structure 10 structure = 60 panel) | p/p | Instaling | Protection | consumption |
|---|---|---|---|---|---|---|
| **Master controll Off the ground – RF** | 1 mastercontroller (5$)+ 60 slavescontrollers (10$)+1 transsievers(5$)+ 60 receivers (4$) | 860 | 14.3 | easy | **** | 45,806 w |
| **GPS On the panel** | 10 GPS (30$)+60 slavecontrollers(10$) | 900 + wires (less) | 15 | med | **** | 28,36 w |
| **Master controll In the groud** | 10 mastercontrollers (5 $)+ 60 slavescontrollers (10$)+10 signaboosters (5$) | 700 + wires | 12 | diff | **** | 31,92 w |
| **Master controll Off the ground –PLC** | 10 mastercontrollers(5$)+ 10 transciever (SIG60)+60 receivers(SIG61)+60 slavecontrollers (10$) | 650 + trans/reiver | | diff+ | **** | 44,72 w |
| **GPS Off the panel - RF** | 1 GPS (30$)+ 60 recievers (5$) + 1 transcievers(5$)+60 ucontrollers(10$) | 875 | 14.5 | easy+ | **** | 45,45 w |
| **Master controll Off the ground –WIFI** | 1 mastercontroller(5$)+ 61 transcievers WIFI+60 slavecontrollers (10$) | 600 + 61 transcievers | | diff++ | ***** | 37,69 w |

## 6.8 Appendix 8 Converter's parameters

| PARAMETER | ECUATION |
|---|---|
| $d$ | $\frac{V_{out}+V_F}{V_{in}-V_{sat}-V_{out}}$ |
| $T$ | $\frac{1}{f}$ |
| $t_{off}$ | $\frac{T}{d+1}$ |
| $t_{on}$ | $T-t_{off}$ |
| $C_T$ | $4\times10^{-5}\cdot t_{on}$ |
| $I_{pk}$ | $2\cdot I_{out}$ |
| $R_{SC}$ | $\frac{0.3}{I_{pk}}$ |
| $L_{min}$ | $\left(\frac{V_{in}-V_{sat}-V_{out}}{I_{pk}}\right)\cdot t_{on}$ |
| $C_o$ | $\frac{I_{pk}\cdot T}{8\cdot V_{ripple}}$ |
| $V_{out}$ | $\left(1+\frac{R2}{R1}\right)\cdot 1.25$ |

Table 4 Parameter ecuations

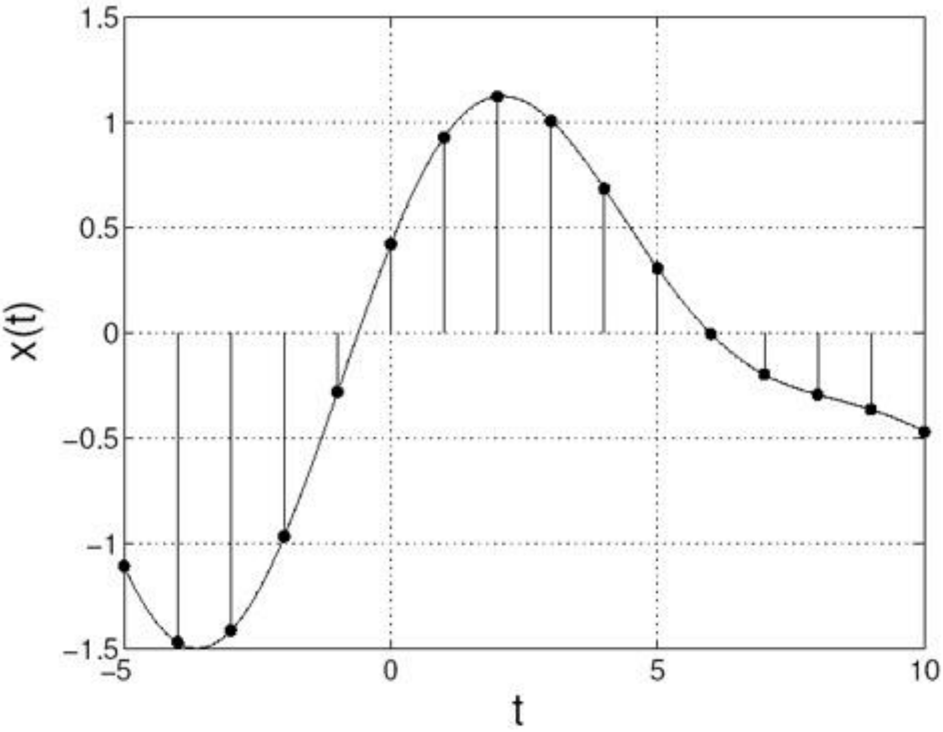
Description of the paremeters of the Table 4:

- The parameters $t_{on}$, $t_{off}$ ,T and d describe the PWM signal witch is used to control the output voltage. The MC33063A generate this signal the internal way.

- The converter allows to work with frecuency values ($f$) up to 100 kHz but the typical value is 33 kHz.
- The $V_F$ value is determinated by el external zener (1N5819). The typical value is 0.85 V.
- The $V_{sat}$ is the saturation voltage of the internal transistors. When they are in Darlinton connection the typical value is 1 V.
- $V_{ripple}$ is the maximum ripple voltage desired in the output.
- $V_{in}$ is the input voltage. The input votage should be between 3 and 40 V.
- $V_{out}$ is the output voltage desired.
- $I_{out}$ is the output Current desired.
- The parameters $C_T$, $C_o$ ,$L_{min}$ and $R_{sc}$ are values of the electronic components that are necessary to make it work.

## 6.9 Appendix 9 Explanation of digital and analog signal

An analog signal is any signal that time variable features (voltage tension) represent a value. An analog signal uses some property of the medium to convey the signal's information. For example, an aneroid barometer uses rotary position as the signal to convey pressure information. Electrically, the property most commonly used is voltage followed closely by frequency, current, and charge.




Picture 84

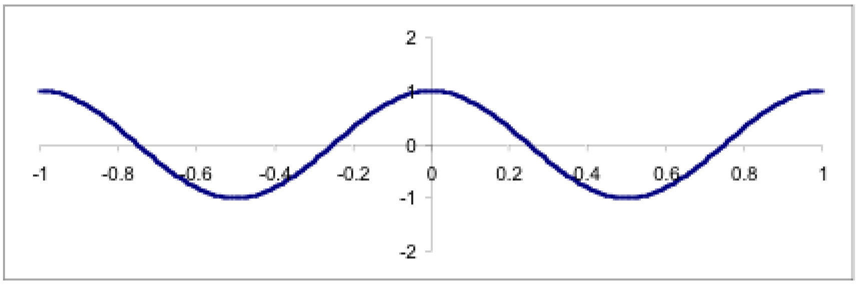
As you can see in the graph there is a value that response to a physical change and varies according to the time.

Any information may be conveyed by an analog signal. Often such a signal is a measured response to changes in physical phenomena, such as sound, light, temperature, position, or pressure, and is achieved using a transducer.

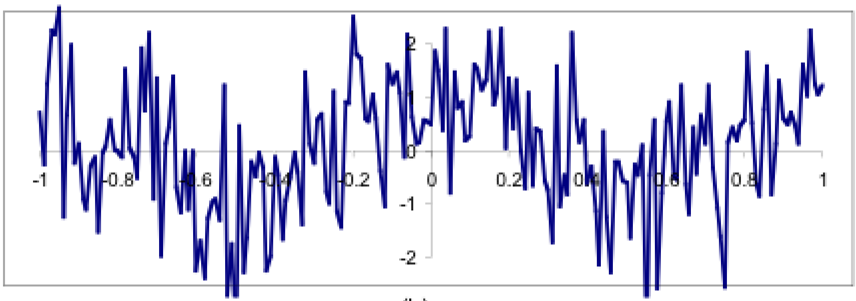
The analog signal can be received and processed more easily than the digital as it is directly connected to a physical change. For an example a pressure sensor automatically produce a different voltage level with a different pressure and this can be detected directly to use this as a digital signal you must first translate to a square waveform that represents the bits.

But the main advanced is the fine definition of it compare to the digital one.

On the other side the analog signals are very sensitive to the noise and the loss. This makes it difficult to send this signal long distance.




(a)

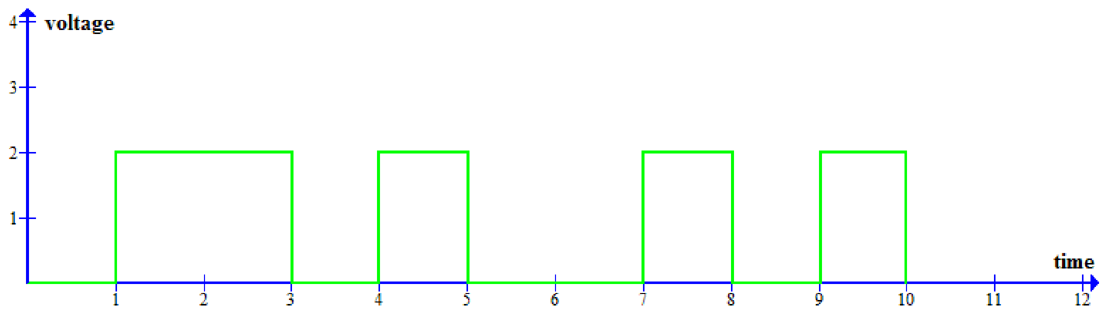



(b)

Picture 85 The sending and the receiving signal

## Digital

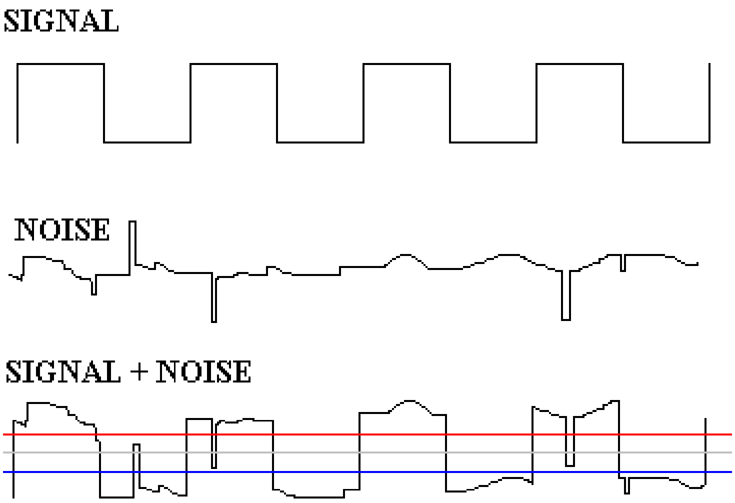
The digital signal is a waveform that switches between voltages values (usually 0V-5V) representing the two status of a bit (0 & 1). The most of the devices are interpret the signal as low or high that response to 0 & 1. Using the digital signal the information is necessary to be converted to a string of bits to be sent.



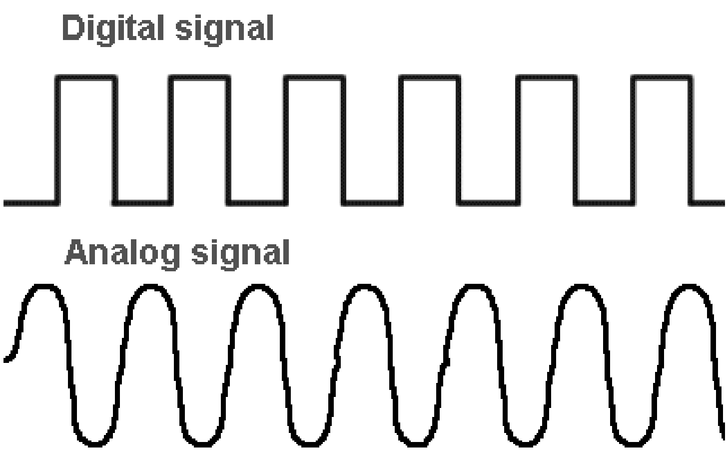

Picture 86

If we try to read the signal in above graph we will read the following bits. {0, 1, 1, 0, 1, 0, 0, 1, 0, 1}

The digital signal is easier to send long distances without noise. A digital waveform is possible to be read even if it was affected by noise as it is only necessary to realize the low and the high status. And this is the main advance against to the analog.




Picture 87 it is visible that even with the noise is easy to understand where there are low and where the high levels that is enough to read the information




Picture 88 An analog and a digital signal

## 6.10 Appendix 10 Oscillator calculations

The equations, to calculate the different parameters of the oscillator circuit, are showed below:

$$t_1 = 0.693 \cdot R_A \cdot C$$

$$t_2 = \frac{R_A \cdot R_B}{R_A + R_B} \cdot C \cdot Ln\left(\frac{R_B - 2R_A}{2R_B - R_A}\right)$$

$$T = t_1 + t_2$$

$$f = \frac{1}{T}$$

$$R_B < 0.5 \cdot R_A$$

The frequency necessary for this application is 170 KHz and to simplify the calculations is been set the value of $R_B$ (15KΩ). Thus, $R_A$ is equal a 36 KΩ and the capacitor value is 120 pF.

However, experimentally it was necessary to use a capacitor of 57 pF instead of 120 pF

## 6.11 Appendix 11 Filter calculations

There are some direct equations whose unknown factors are the values of the components that are wanted to know:

**Equation 2** $$f_o = \frac{1}{2 \cdot \pi \cdot R \cdot C};$$

**Equation 3** $$Q = \frac{1}{3-G};$$

**Equation 4** $$G = 1 + \frac{R_2}{R_1};$$

**Equation 5** $$A_m = \frac{G}{3-G};$$

Mr. Dixneuf told that a good filter should have a Q factor of 10, so first by changing the Q by 10 in the Equation 3, the gain G got:

$$Q = \frac{1}{3-G} \Rightarrow 10 = \frac{1}{3-G} \Rightarrow 30 - 10G = 1 \Rightarrow -10G = -29 \Rightarrow G = 2{,}9$$

**This gain in our system is not a critic value. After knowing it, it is substituted in**

Equation 4 and $R_1$ and $R_2$ will be known:

$$G = 1 + \frac{R_2}{R_1} \Rightarrow 2{,}9 = 1 + \frac{R_2}{R_1} \Rightarrow \frac{R_2}{R_1} = 1{,}9$$

There is a ratio between $R_1$ and $R_2$. If the value of $R_1$ is given, such as $\boxed{R_1 = 10\,k\Omega}$, then $R_2 = R_1 \cdot 1{,}9 = \boxed{19\,k\Omega = R_2}$.

To know the gain of the filter in its maximum peak (center frequency) it is only needed to use the Equation 5:

$$A_m = \frac{G}{3-G} = \frac{2{,}9}{3-2{,}9} = \frac{2{,}9}{0{,}1} = 29;$$

There is an equation left, and with this, all the components will be defined.

$$f_o = 200\,kHz = \frac{1}{2\cdot\pi\cdot R\cdot C} \Rightarrow R\cdot C = \frac{1}{2\cdot\pi\cdot 200\cdot 10^{3}} = 7{,}95\cdot 10^{-7}$$

There is a ratio between $C$ and $R$. First of all, the value of $C$ the must be given for us, such as $\boxed{C = 100\,pF}$. There are not infinite values for all the passive components (resistors, capacitors and inductances), so in the market you could buy only the standard ones. Each standard has its own serie, and each serie has its own values and tolerances. For this capacitor, it should be one value in the standard E-12 because all the capacitors have a high tolerance of 20% (it belongs to E-12). Resistors are quite different because there is a wide range of values since they belong from E-12 (20%) to E-192 (0,1%), depending on the tolerance. The higher the serie is, the more standard values are and the less tolerance is.

Getting $C$, then the $R$ value is calculated:

$$R = \frac{7{,}95\cdot 10^{-7}}{C} = \frac{7{,}95\cdot 10^{-7}}{100\cdot 10^{-12}} = \boxed{7{,}95\,k\Omega = R}$$

All the theoretical components value results must be adapted to standard ones. They are shown in the table below:

| **Theorical components /Serie** | **Serie E-12 (20% tolerance)** | **Serie E24 (5% tolerance)** | **Serie E96 (1% tolerance)** |
|---|---|---|---|
| $R$ **=7,95 kΩ** | | 8,2 kΩ | |
| $C$ **=100 pF** | 100pF | | |
| $R_1$ **= 10 kΩ** | | 10kΩ | |
| $R_2$ **= 19 kΩ** | | | 19,1KΩ |

**Table 5 Chart of standardized components**

## 6.12 Appendix 12 Used equipment

### Oscilloscope (Tekronix TDS 2014B)

An oscilloscope is a measurement instrument and measures the voltage over time or the differences between two different voltages. The voltage is shown to the left of the display and continues to the right side of the display. When the voltage point has travelled all the way to the right side it starts over again. There is some delay in the oscilloscope and the eye and therefore the display shows the different positions as a stream of light.


Picture 89 Oscilloscope Tekronix TDS 2014 B

### Signal generator (Wavetek 185)

A signal generator is used in order to create a signal. You can for example decide different frequency, amplitude and different type of signal. This unit was used in order to simulate the high frequency current that was going to be sent through the power line.


Picture 90 Signal generator Wavetek 185

## DC power supply (ELC AL 991S)

A power supply was used to provide the correct voltage and current. This unit served for simulating the input from the panel.


Picture 91 DC power supply ELC AL 991S

## Freescale Code Warrior software

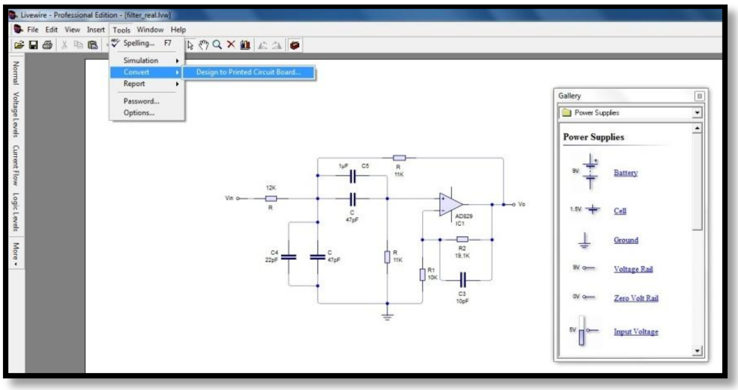
The software that was used for programming the microcontrollers was Freescale Code Warrior IDE version 5.7.0 Build 2015 by Freescale Semiconductor. INC. The program is programming in the language C++.

## PSpice

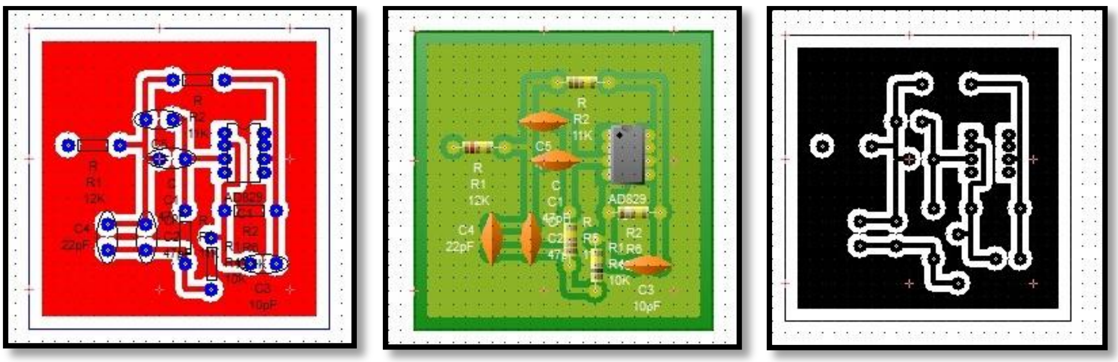
PSpice© is one of the best electronic programs and it is used to simulate circuits since its simulations are close to the reality.  This program has been used to do the simulations of the filter:

## PCBWizard software

PCBWizard © is another electronic program thought to build PCB (Printed Circuit Board) and it is a complement of Livewire©. The design made with Livewire© is imported to PCBWizard and it generates automatically the printed circuit board layout.




Picture 92 Conversion Livewire© circuit to PCB with PCBWizard©



Picture 93 Normal, real2D and artwork views.

## The Google documents

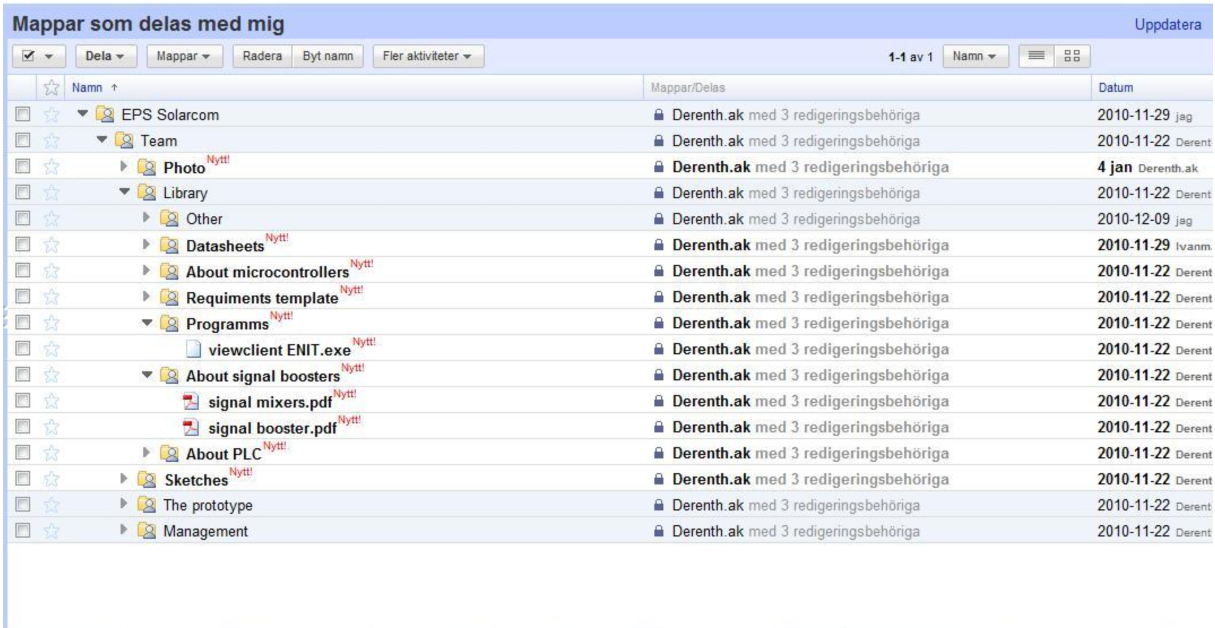
To make it easy for everyone to take part in the different documents and datasheets, Google documents was used.

Google documents made it easy to share the documents. Another group used a different type of file sharing program named Drop Box which was used for the same purpose as

Google documents.

### How it works

Someone make a digital folder on Google's server and then invite people to join this specific folder. If the person who is invited accepts, he can read all the files in that folder. After getting access to the folder, everyone can upload documents, replace documents and work in the documents.

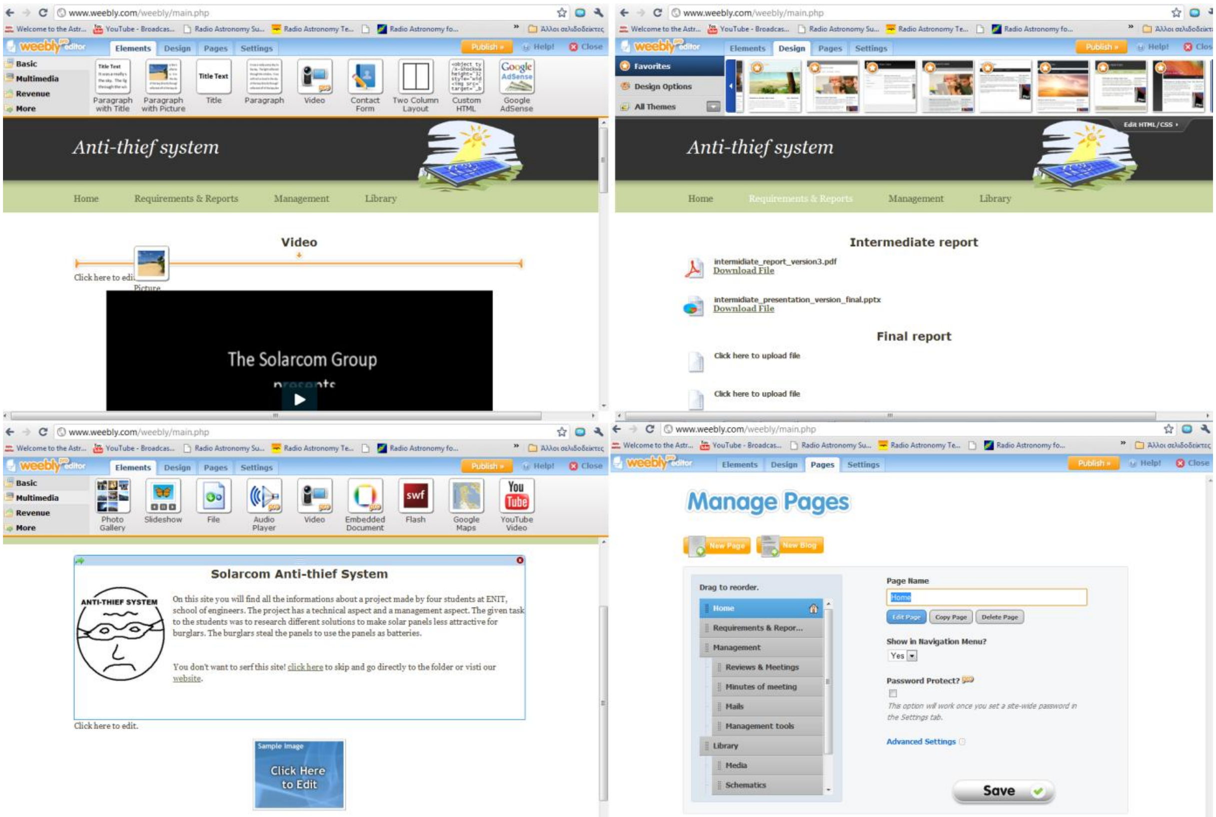



Picture 94 The Google documents

## The website

In the beginning of the project, when the deliverables were chosen, the students decided to make a digital folder containing minutes of meeting, power points etc. (see the WBS) To make it easy to find the right information for the person using the CD a site will appear. The site looks exactly like a website and the website with the exactly same appearance can be found on the internet (http://anti-thief.weebly.com/). This makes it very easy to quickly find exactly the information needed.

Weebly (www.weebly.com) is an easy drag & drop style website editor that also provides you host names and host for free. It makes it easy to add photos, video, text and other functions in your websites. It is a complete interface that proposes you to use already existing design template, automatically develop your menu and adding elements on the website. But it also gives you the access to the HTML/CSS code to design your own template or add even more things.



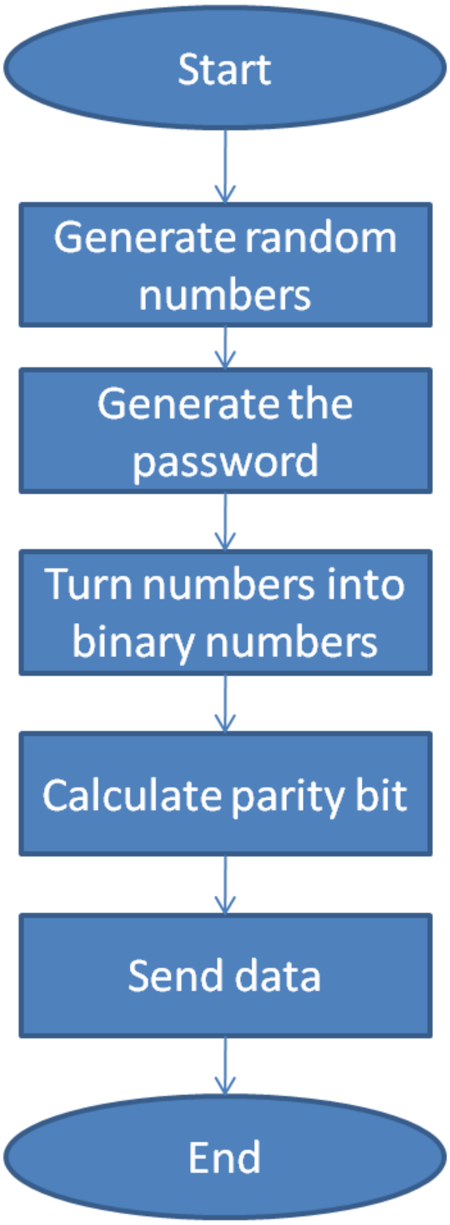

**Picture 95 The weebly interface**

## 6.13 Appendix 13 The C++ code

### MASTER CONTROLLER

#### *Flow chart*

In the Picture 96 is showed the master controller flow chart. This flow chart describes the general functions .




Picture 96 Mastercontroller flow chart

### *Master controller Code*

```
/*******************************************************************************
*******************************************************************************
************MASTER CONTROLLER CODE*****************************
*******************************************************************************
**********************************************************************
***Date: 21/01/2011                                                    ***
***Version: 1.0                                                        ***
***Software: CodeWarrior 5.7.0 Build 2015 by Freescale Semiconductor. INC  ***
***Programmer: Solarcom- Enit group                                    ***
*******************************************************************************
**********************************************************************

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++GENERAL DESCRIPTION+++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+This code has the mission to perform the below functions:              +
+1-Generate 4 random numbers between 0 and 7                             +
+2-Calculate the password with the 4 random numbers                      +
+3-Turn decimal numbers into binary numbers                              +
+4- Calculate the parity bit                                             +
+5- Send all the bits                                                    +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++/*

#include <hidef.h> /* for EnableInterrupts macro */
#include "derivative.h" /* include peripheral declarations */
#define kback  0x04 //erase the interrupt keyboard flag
#define ch0f 0x80
#define decze 7 //size of the vector that contain all the numbers in decimal format
#define binze 21 //decze*3 size of the vector that contain all the numbers in binary format
#define rdmze 4 // number of random numbers
```

```
////////////////////////  Global variables   ///////////////////////////////////////////////////////

int tabTrans[decze]; //register of decimal numbers
int bina[binze]; // register of binary numbers
int buf; // buffer
int flag_sending=0; // 1=sending data.
int flag_send_now=0; //when you press the button the data are sent
int cycle=0; //it's necessary to maintain the separation between bits

///////////////////////////////////MCU_INIT FUNCTION///////////////////////////////////////////

void MCU_init(void); /* Device initialization function declaration */

/////////////////////////////////TEMPO FUNCTION////////////////////////////////////////////////

void tempo(void)    // This function is just to waste  time
{ int x=0;
while (x<=200) x++;
}

///////////////////////////INTERRUPT INTER_BP////////////////////////////////////////////////
// When someone press the button the program goes to this interruption

interrupt void Inter_BP(void)
{
  tempo();
  flag_send_now=1;//activate the flag send now
  if( PTAD_PTAD2==0)
  {
    PTAD_PTAD3=1- PTAD_PTAD3; //toggles the led that is connected to this pin
  }
  KBISC|=kback;  //erase the flag
}

///////////////////////////INTERRUPT INIT_CHRONOMETER////////////////////////////////////
```

```
void init_chronometer(void)
{
  MTIMMOD=65;  //Set the limit counter 65=500us
  MTIMSC_TSTP=1;     //0= start the counter
}

///////////////////////////INTERRUPT INTER_CHRONO///////////////////////////////////////////

interrupt void Inter_chrono(void)
{
  PTAD_PTAD1=1;    //deactivate output => 1=disable oscillator, 0= enable oscillator
  MTIMSC_TSTP=1;     //0= start the counter
  MTIMSC_TRST=1;     //Reset the counter
  if(cycle==0)  //check it has left the separation between bits
  {
    cycle=1;
    MTIMMOD=65;                  //separation between bits
    MTIMSC_TSTP=0;     //0= start the counter
  }
  else
  { cycle=0;
    flag_sending=0;     //deactivate flag sending
  }

   MTIMSC&=~0b10000000;    //deactivate the flag
}

/////////////////////////// DECIMAL_TO_BINARY FUNCTION////////////////////////////////////
// This function performs the conversion from decimal number to binary number

void decimal_to_binary(int number[])

{  int i,x=0,n;
   n=decze-1;
```

```
   for(i=decze;i>=0;i--)
 {
   x=(i+1)*3-1;
    bina[x]=number[i]%2;
    buf=number[i]/2;
    bina[x-1]=buf%2;
    buf=buf/2;
    bina[x-2]=buf%2;
 }
}

/////////////////////////// TRANSMIT FUNCTION//////////////////////////////////////
//This function calculates the parity bit and sends the start bit, the data and the parity bit

 void transmit(int data[])
{
 int i;
 int addition=0;
 int parity=0;
 flag_sending=1;

 //*********send start bit*******************

  MTIMSC_TSTP=1;    //stop counter
  MTIMMOD=180;      //change the limit counter (1300us)
  PTAD_PTAD1=0;     //activate output
  MTIMSC_TSTP=0;    //start counter
  while(flag_sending==1)
   {
   }

 //*********send datas*********************

  for(i=0;i<binze;i++)
 {
```

```
 // this condition avoids to erase the register prematurely
   if ((bina[i]==1)&&(flag_sending==0))
   {
        flag_sending=1;
     MTIMSC_TSTP=1;   //stop counter
     MTIMMOD=130;      //change the limit counter
     PTAD_PTAD1=0;    //activate output
     MTIMSC_TSTP=0;   //start counter
    }
// this condition avoids to erase the register prematurely
   if ((bina[i]==0)&&(flag_sending==0))
    {
     flag_sending=1;
     MTIMSC_TSTP=1;    //stop counter
     MTIMMOD=100;      //change the limit counter
     PTAD_PTAD1=0;     //activate output
     MTIMSC_TSTP=0;   //start counter
    }
    while(flag_sending==1) // wait for the bit is sent
    {
    }
  }
  //**************calculate the parity*********

  addition=0;
  for(i=0;i<binze;i++)
  {
   addition=addition + bina[i];
  }
  parity=addition%2; //if remaider is 0 addition is a odd number

  //*******send parity*********************

   if (parity==1) //check if the parity bit is a odd number
   {
```

```
      flag_sending=1;
      MTIMSC_TSTP=1;   //stop counter
      MTIMMOD=130;      //change the limit counter
      PTAD_PTAD1=0;    //activate output
      MTIMSC_TSTP=0;   //start counter
    }
   if (parity==0)  //check if the parity bit is a even number
     {
      flag_sending=1;
      MTIMSC_TSTP=1;    //stop counter
      MTIMMOD=100;      //change the limit counter
      PTAD_PTAD1=0;     //activate output
      MTIMSC_TSTP=0;   //start counter
     }
    while(flag_sending==1)
    {
    }
}

////////////////////////// MAIN FUNCTION////////////////////////////////////

void main(void)
{

//*********Inicialization function*************

  MCU_init(); /* call Device Initialization */
  init_chronometer();
  PTAD_PTAD1=1;
  EnableInterrupts;

  //*********loop forever*******************

  while(1)
  {
```

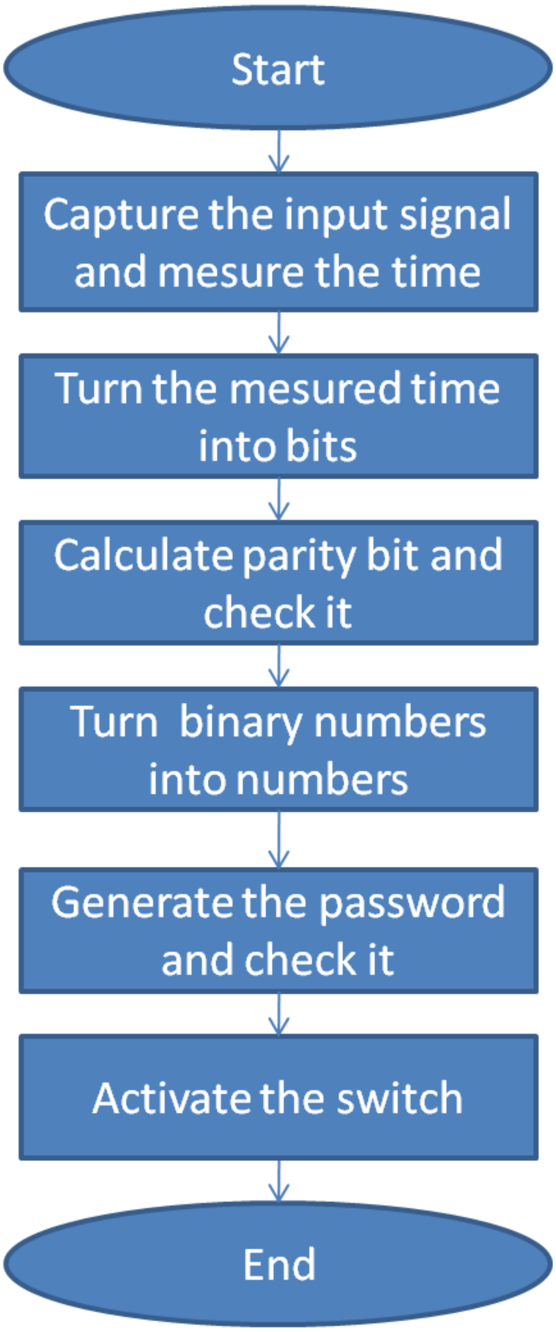
```
    while(flag_send_now==1)
    {

  //***generate 4 random number***

      for(i=0;i<rdmze;i++)
      {
       tabTrans[i]=rand()%8;
      }
       x=0;
  //***************************
 //calculate the password. It's possible to change it for other algorithm more complex

      for(i=rdmze;i<decze;i++)
      {
       tabTrans[i]=(tabTrans[x]+1)%8;
       x++;
      }
 //***************************
      decimal_to_binary(tabTrans);  convert the numbers to binary
      transmit(bina);// send the data
      flag_send_now=0; //deactive the flag.
    }
 }
}
       /*UNASSIGNED_ISR,          Int.no. 23 Vrti (at FFD0)              Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 22 Reserved2 (at FFD2)         Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 21 Reserved3 (at FFD4)         Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 20 Vacmp (at FFD6)             Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 19 Vadc (at FFD8)              Unassigned */
       /*isrVkeyboard,                      Int.no. 18 Vkeyboard (at FFDA)         Used */
       /*UNASSIGNED_ISR,          Int.no. 17 Viic (at FFDC)              Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 16 Vscitx (at FFDE)            Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 15 Vscirx (at FFE0)            Unassigned */
       /*UNASSIGNED_ISR,          Int.no. 14 Vscierr (at FFE2)           Unassigned */
```

```
/*UNASSIGNED_ISR,    Int.no. 13 Vspi (at FFE4)          Unassigned */
/*UNASSIGNED_ISR,    Int.no. 12 Vmtim (at FFE6)         Unassigned */
/*UNASSIGNED_ISR,    Int.no. 11 Reserved13 (at FFE8)    Unassigned */
/*UNASSIGNED_ISR,    Int.no. 10 Reserved14 (at FFEA)    Unassigned */
/*UNASSIGNED_ISR,    Int.no.  9 Reserved15 (at FFEC)    Unassigned */
/*UNASSIGNED_ISR,    Int.no.  8 Reserved16 (at FFEE)    Unassigned */
/*isrVtpmovf,        Int.no.  7 Vtpmovf (at FFF0)       Used */
/*UNASSIGNED_ISR,    Int.no.  6 Vtpmch1 (at FFF2)       Unassigned */
/*UNASSIGNED_ISR,    Int.no.  5 Vtpmch0 (at FFF4)       Unassigned */
/*UNASSIGNED_ISR,    Int.no.  4 Reserved20 (at FFF6)    Unassigned */
/*UNASSIGNED_ISR,    Int.no.  3 Vlvd (at FFF8)          Unassigned */
/*UNASSIGNED_ISR,    Int.no.  2 Virq (at FFFA)          Unassigned */
/*UNASSIGNED_ISR,    Int.no.  1 Vswi (at FFFC)          Unassigned */
```

## SLAVE CONTROLLER

### *Flow chart*

In the Picture 97 is showed the master controller flow chart. This flow chart describes the general functions (to obtain more information read the code (Slave controller flow)).




**Picture 97: slave controller flow chart**

### *Slave controller flow*

```
/*********************************************************************************
*********************************************************************************
************SLAVE CONTROLLER CODE*******************************
*********************************************************************************
***********************************************************************
***Date: 21/01/2011                                                            ***
***Version: 1.0                                                                ***
***Software: CodeWarrior 5.7.0 Build 2015 by Freescale Semiconductor. INC      ***
***Programmer: Solarcom- Enit group                                            ***
*********************************************************************************
***********************************************************************

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++GENERAL DESCRIPTION+++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+This code has the mission to perform the below functions:                  +
+1-Capture the input signal and measure the time between events             +
+2- Calculate the parity bit and check the parity                           +
+3-Turn binary numbers into decimal numbers                                 +
+4 -Calculate the password and check it                                     +
+5- Activate the switch                                                     +
+5- Limit the time that the switch is working                               +
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++/*

#include <hidef.h> /* for EnableInterrupts macro */
#include "derivative.h" /* include peripheral declarations */
#define kback  0x04
#define ch0f 0x80
#define decze 7
#define binze 21   //decze*3
```

```
#define binpaze 22 //decze*3 +parity bit
#define rdmze 4 // number of random numbers
#define pwdze 3  //size of password always =decze-rdmze
#define allnumbers 46  //(startbit+databit(binze)+paritybit)*2

/////////////////////// Global variables  ////////////////////////////////////////////////////

int tabTrans[decze]; //register for decimal numbers
int pwd[pwdze]; //register to save the password that is generated by the code
int unsigned reciver[allnumbers]=0; //register  where is saved the counter value in each event
int unsigned times[allnumbers]=0; //register  where is saved the time between events
short unsigned bitss[binpaze]=0;//register for binary numbers
int r=0;
int error=0;
int unsigned buf;
int unsigned timerr=0;
int flag_receiving=0;
int flag_translate=0;

/////////////////////// MCU_INIT FUNCTION ///////////////////////////////////////////////

void MCU_init(void); /* Device initialization function declaration */

////////////////////////////////TEMPO FUNCTION///////////////////////////////////////////////

void tempo(void)
{ int x=0;
while (x<=1000) x++;
}

/////////////////////////INTERRUPT INTER_BP/////////////////////////////////////////////////
// When someone press the button the program goes to this interruption

interrupt void Inter_BP(void)
{
```

```
  tempo();
  if(PTAD_PTAD2==0)
  {
     PTAD_PTAD3=1- PTAD_PTAD3; //toggles the led that is connected to this pin
     PTAD_PTAD0=1- PTAD_PTAD0; //toggles the switch that is connected to this pin
  }
  KBISC|=kback;
}
//////////////////////////INTERRUPT INTER_INPUT///////////////////////////////////////////
//This interrupt capture the input signal

interrupt void Inter_input(void)
{
if (flag_translate==0) //this condition is to avoid erase the register prematurely
{
    error=0;
   reciver[r]=TPMC1V; // copy the counter value



  if (r>0)  //requirement to calculate the time
   {
    buf=reciver[r] -reciver[r-1]; // Calculate the time

    if (flag_receiving==1)    //copy all the times


        { flag_receiving=1;
          times[r-1]=buf;
        }
    if ((2700<buf)&&(buf<3000))   //bit start received ???   1290us-1433us
        { flag_receiving=1;      // ready to receive
          times[r-1]=buf;
        }
    }

  if ((flag_receiving==0)&&(r==1)) // bit start no received
```

```
      {
        // reset values
        r=0;
        reciver[0]=0;
        reciver[1]=0;
        buf=0;
      }
    else
        r++;
    if(r>=allnumbers)  // all bits received???
    {
     // reset values
     flag_receiving=0;
     flag_translate=1;
     reciver[0]=0;
     reciver[1]=0;
     r=0;
    }
}
  TPMC1SC&=0b01111111;     //deactivate the flag
}

////////////////////////// INIT_CHRONOMETER FUNCTION//////////////////////////////////////

void init_chronometer(void)
{
//configuration for 7.828 ms
  MTIMCLK=8;
  MTIMMOD=0;
  MTIMSC_TSTP=1;      //0= start the counter
}

//////////////////////////INTERRUPT INTER_CHRONO//////////////////////////////////////////

interrupt void Inter_chrono(void)
```

```
{
 timerr++;
 if (timerr>1277) //Does the switch turned on 10 seconds?
 {
    PTAD_PTAD3=0; //deactivate the led
    PTAD_PTAD0=1;  //deactivate the switch
    timerr=0;// reset the counter
    MTIMSC_TSTP=1;     //stop the timer
 }
  MTIMSC&=~0b10000000;    //deactivate the flag
}

/////////////////////////// TIMES_TO_BIT FUNCTION /////////////////////////////////////////////
//this translate the times to bits

void times_to_bit(void)
{
int v=0,i=0,d=0;
for (v=0;v<allnumbers;v++)
 {
   if ((2700<times[v])&&(times[v]<3000))   //bit start ?   1290us-1433us
     {
       if (v!=0)// is it in a different position than the 0 position??
       error=1;     //the start bit isn´t in its position
     }
    if ((1000<times[v])&&(times[v]<1300))   //separation?    477us-621us
     {
      d=v%2;
      if (d!=1) is the separation in a odd position??
       error=2;     //the separation is in a wrong position
     }
   if ((1400<times[v])&&(times[v]<1850))   //bit  0?    669us-884us
     {
      bitss[i]=0;
      i++;
```

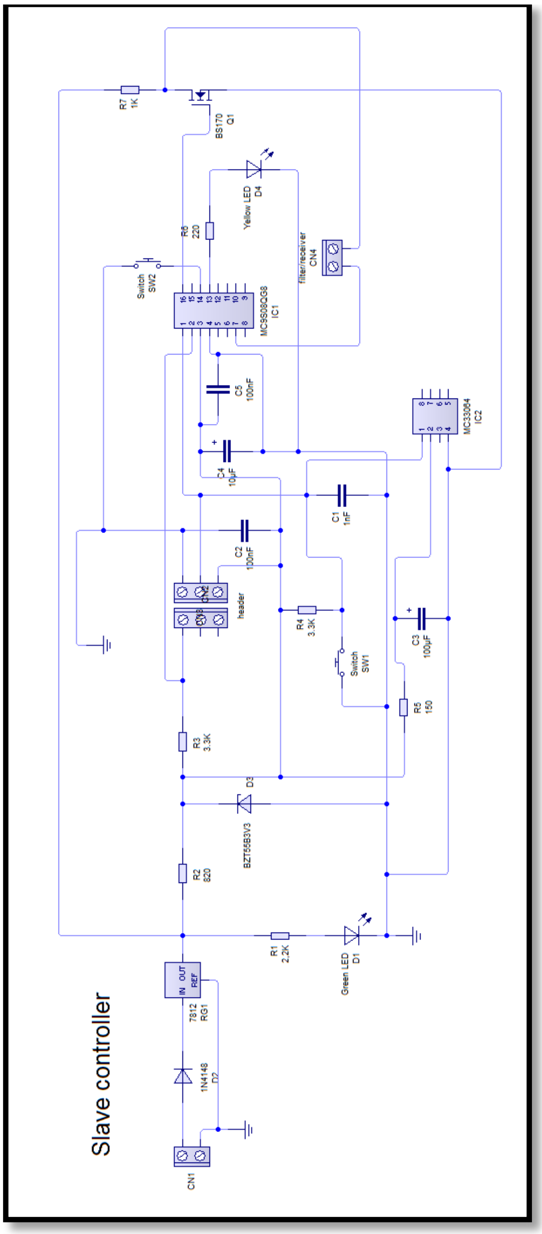
```
        }
      if ((1900<times[v])&&(times[v]<2600))   //bit 1?  908us-1242us
        {
          bitss[i]=1;
          i++;
        }
  }
}

/////////////////////////// BINARY_TO_DECIMAL FUNCTION//////////////////////////////////////
// This function performs the conversion from binary number to decimal number

void binary_to_decimal(void)
{
 int j=0,m=0,n=0;
 for(j=0;j<decze;j++)
  {
    m=4*bitss[n];
    n++;
    m=m+2*bitss[n];
    n++;
    m=m+bitss[n];
    n++;
    tabTrans[j]=m;
  }
}
//***************************************************************************

/////////////////////////// MAIN FUNCTION//////////////////////////////////////

void main(void)
{
int x,p,q,s, addition,parity;
//*********Inicialization function*************
```

```
MCU_init(); /* call Device Initialization */
init_timer();
init_chronometer();
PTAD_PTAD1=1;
EnableInterrupts;

//*********loop forever*******************

while(1)
{
  while(flag_translate==1)
  {
  times_to_bit();

//*********calculate parity bit***************

 addition=0;
for(x=0;x<binze;x++)
{
 addition=addition + bitss[x];
}
parity=addition%2;

  //*********check the parity*****************

if (parity!=bitss[binze])
    error=3;// mismatch parity bit

//********** convertion from binary to decimal**

binary_to_decimal(); // convertion from binary to decimal

//**********calculate the password************

p=0;
```

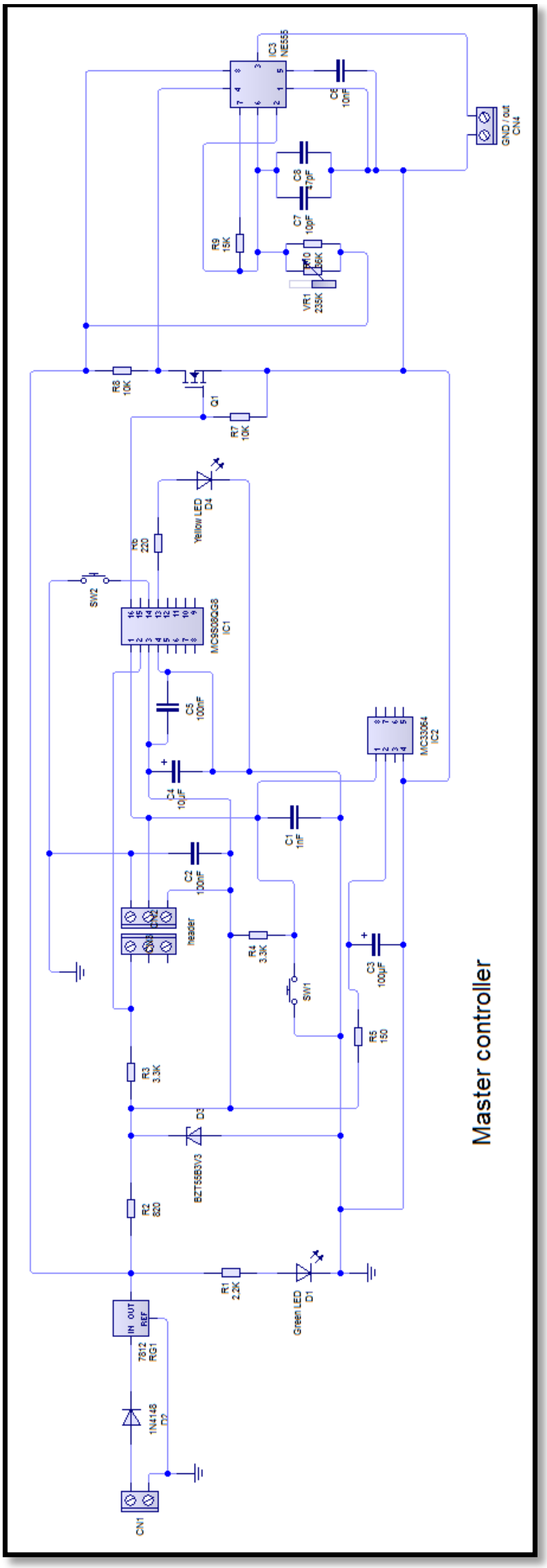
```
    for(s=0;s<pwdze;s++)
     {
      pwd[s]=(tabTrans[p]+1)%8;
      p++;
     }

 //*********check the password*************
 p=0;
 q=0;
    for(s=rdmze;s<decze;s++)
     {
      if(tabTrans[s]==pwd[p])//is the same number??
         q++;
      else
        q=0;
      p++;
     }
    if (q==pwdze) // is our password equal than the received password??
    {
      PTAD_PTAD3=1;//activate the led
      PTAD_PTAD0=0;//actívate the switch
      timerr=0;// reset timer 10 seconds
      MTIMSC_TSTP=0;     //0= start the timer 10 seconds
    }
   else
      error=4;// mismatch the password
  flag_translate=0;//deactivate the flag
  }
 }
}

      /*UNASSIGNED_ISR,          Int.no. 23 Vrti (at FFD0)              Unassigned */
      /*UNASSIGNED_ISR,          Int.no. 22 Reserved2 (at FFD2)         Unassigned */
      /*UNASSIGNED_ISR,          Int.no. 21 Reserved3 (at FFD4)         Unassigned */
      /*UNASSIGNED_ISR,          Int.no. 20 Vacmp (at FFD6)             Unassigned */
```

```
/*UNASSIGNED_ISR,           Int.no. 19 Vadc (at FFD8)              Unassigned */
/*isrVkeyboard,             Int.no. 18 Vkeyboard (at FFDA)         Used */
/*UNASSIGNED_ISR,           Int.no. 17 Viic (at FFDC)              Unassigned */
/*UNASSIGNED_ISR,           Int.no. 16 Vscitx (at FFDE)            Unassigned */
/*UNASSIGNED_ISR,           Int.no. 15 Vscirx (at FFE0)            Unassigned */
/*UNASSIGNED_ISR,           Int.no. 14 Vscierr (at FFE2)           Unassigned */
/*UNASSIGNED_ISR,           Int.no. 13 Vspi (at FFE4)              Unassigned */
/*UNASSIGNED_ISR,           Int.no. 12 Vmtim (at FFE6)             Unassigned */
/*UNASSIGNED_ISR,           Int.no. 11 Reserved13 (at FFE8)        Unassigned */
/*UNASSIGNED_ISR,           Int.no. 10 Reserved14 (at FFEA)        Unassigned */
/*UNASSIGNED_ISR,           Int.no.  9 Reserved15 (at FFEC)        Unassigned */
/*UNASSIGNED_ISR,           Int.no.  8 Reserved16 (at FFEE)        Unassigned */
/*isrVtpmovf,                       Int.no.  7 Vtpmovf (at FFF0)             Used */
/*UNASSIGNED_ISR,           Int.no.  6 Vtpmch1 (at FFF2)           Unassigned */
/*UNASSIGNED_ISR,           Int.no.  5 Vtpmch0 (at FFF4)           Unassigned */
/*UNASSIGNED_ISR,           Int.no.  4 Reserved20 (at FFF6)        Unassigned */
/*UNASSIGNED_ISR,           Int.no.  3 Vlvd (at FFF8)              Unassigned */
/*UNASSIGNED_ISR,           Int.no.  2 Virq (at FFFA)              Unassigned */
/*UNASSIGNED_ISR,           Int.no.  1 Vswi (at FFFC)              Unassigned */
```

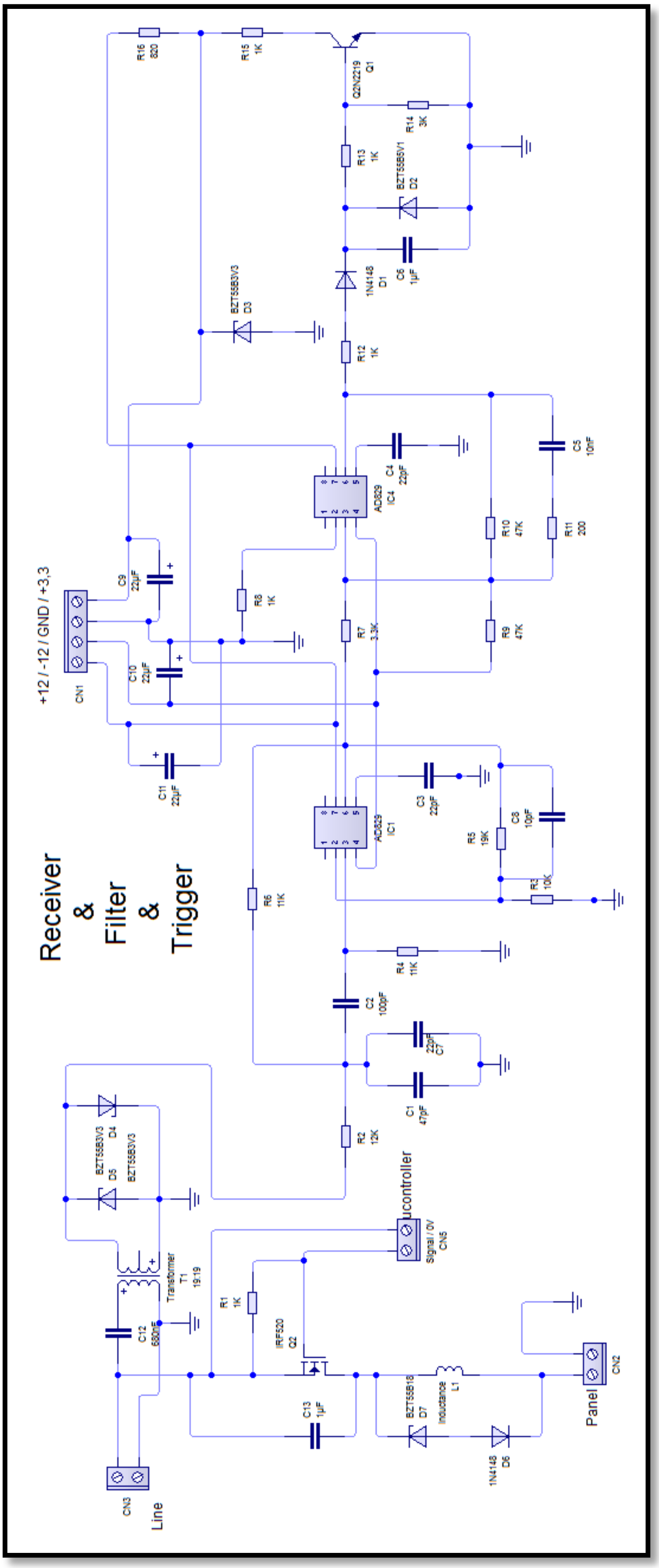
## 6.14 Appendix 14 Schematic and pictures



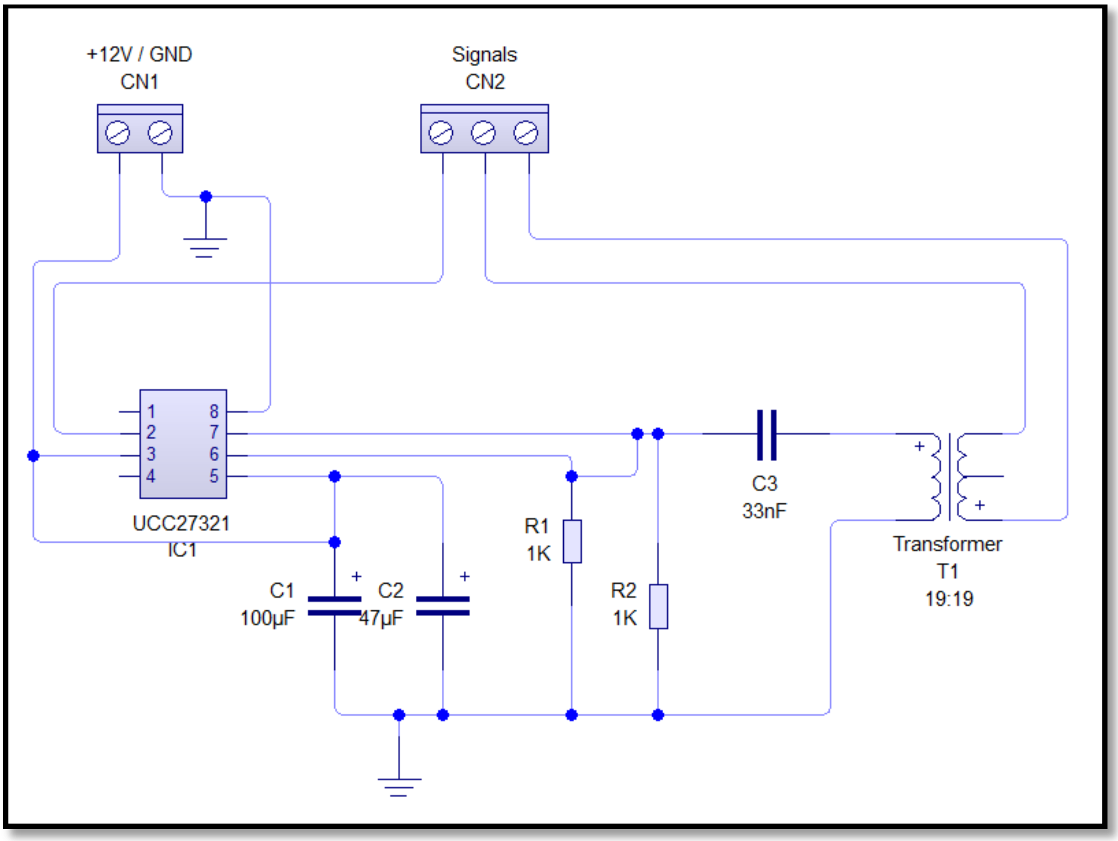

Picture 98 Slave controller schematic



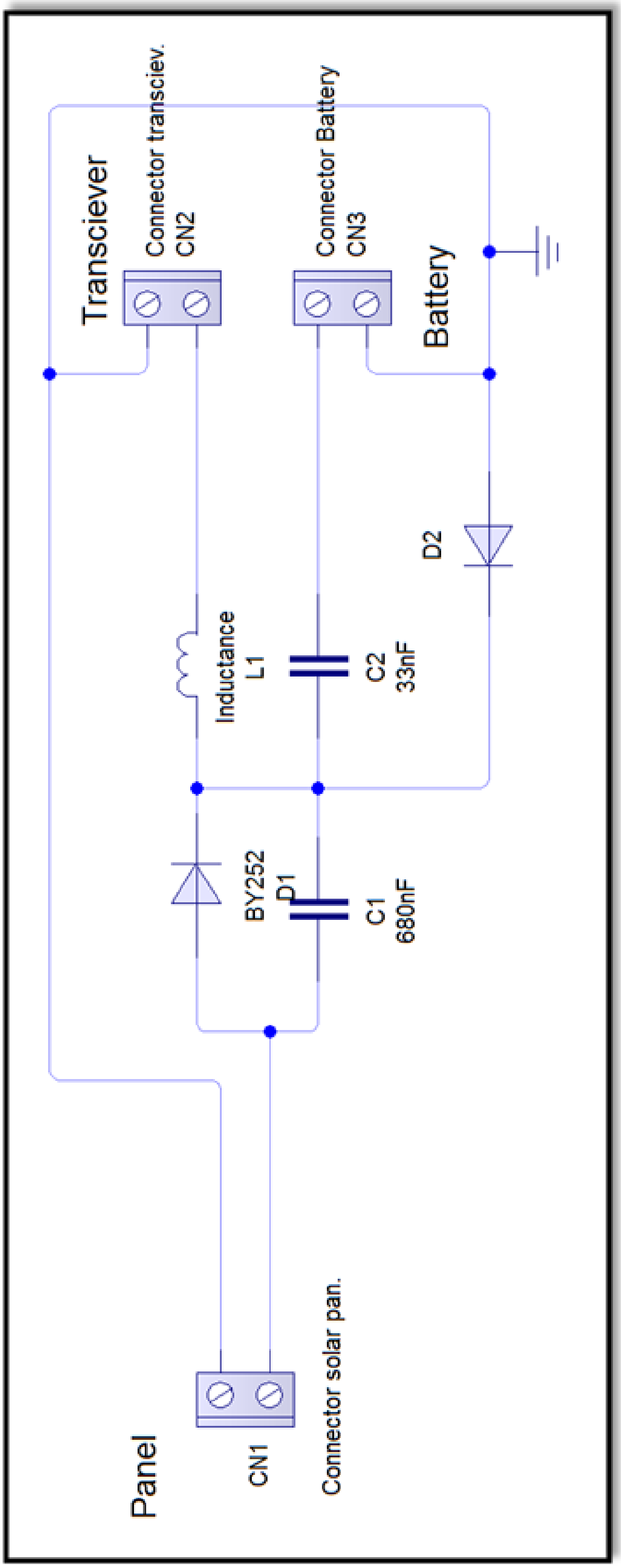

Picture 99 Master controller schematic



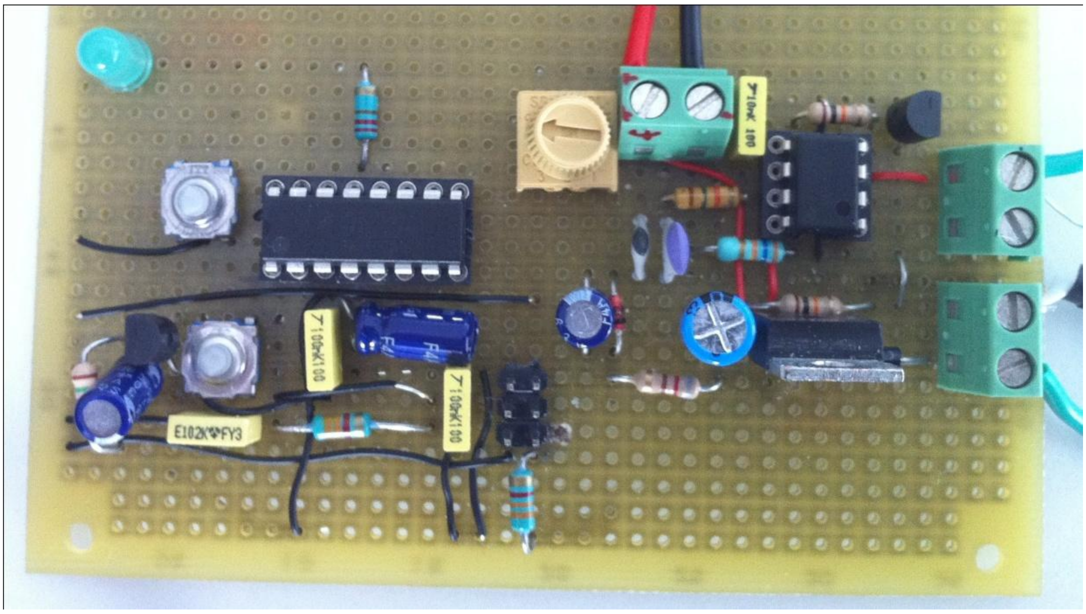
Picture 100 Receiver, filter & trigger schematic



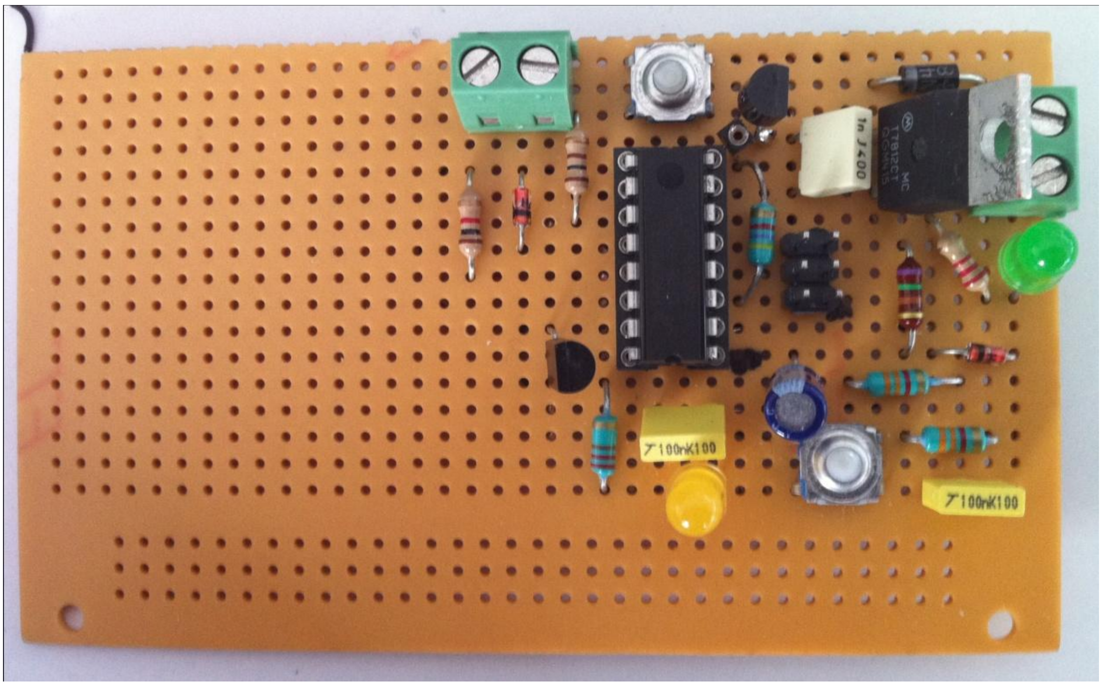

Picture 101 Transmiter


Transciever
Connector transciev.
CN2
Connector Battery
CN3
Battery
D2
Inductance
L1
C2
33nF
BY252
D1
C1
680nF
Panel
CN1
Connector solar pan.

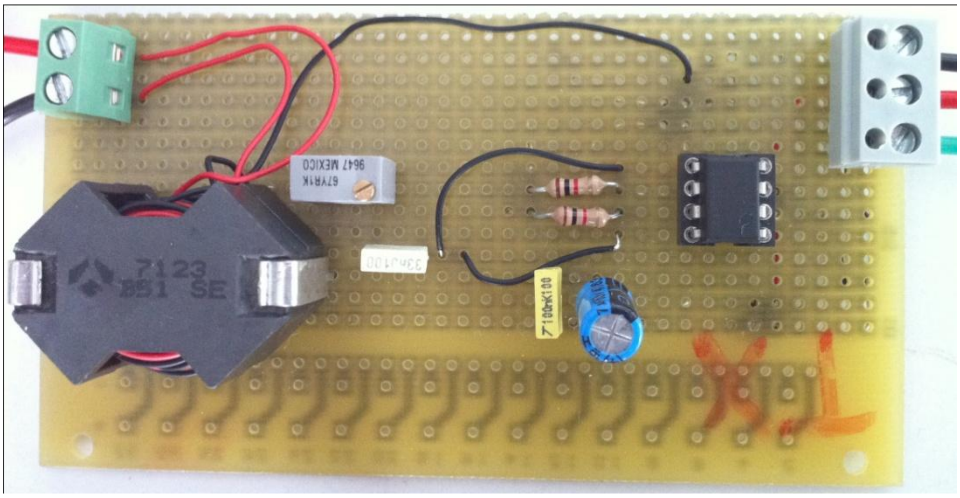
Picture 102 Connector



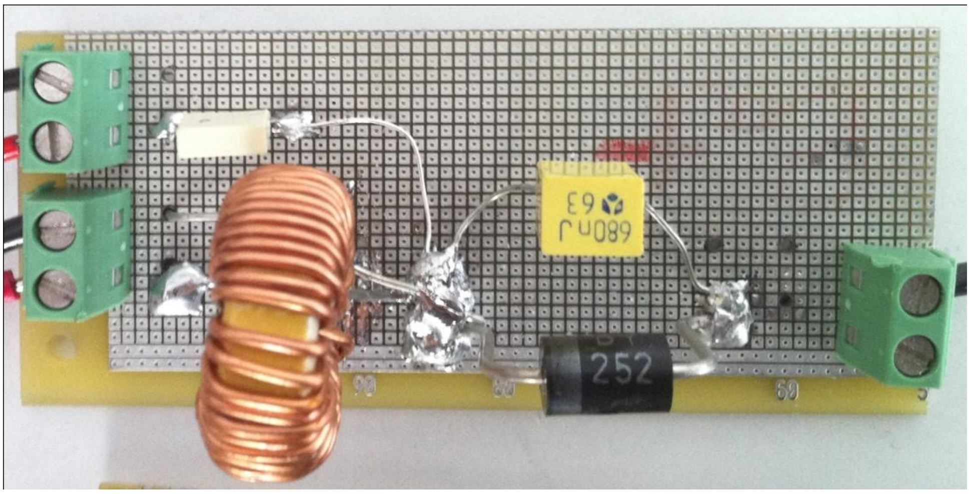
Picture 103 Master controller



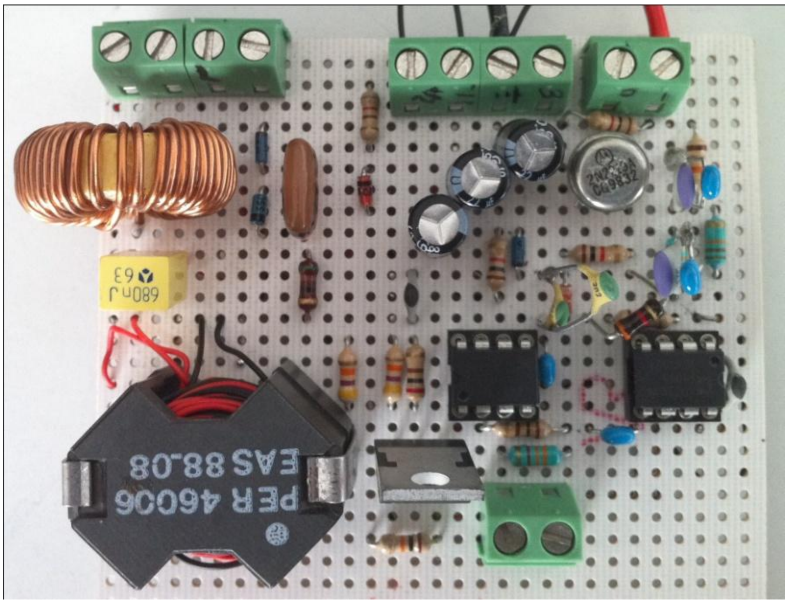
Picture 104 Slave controller



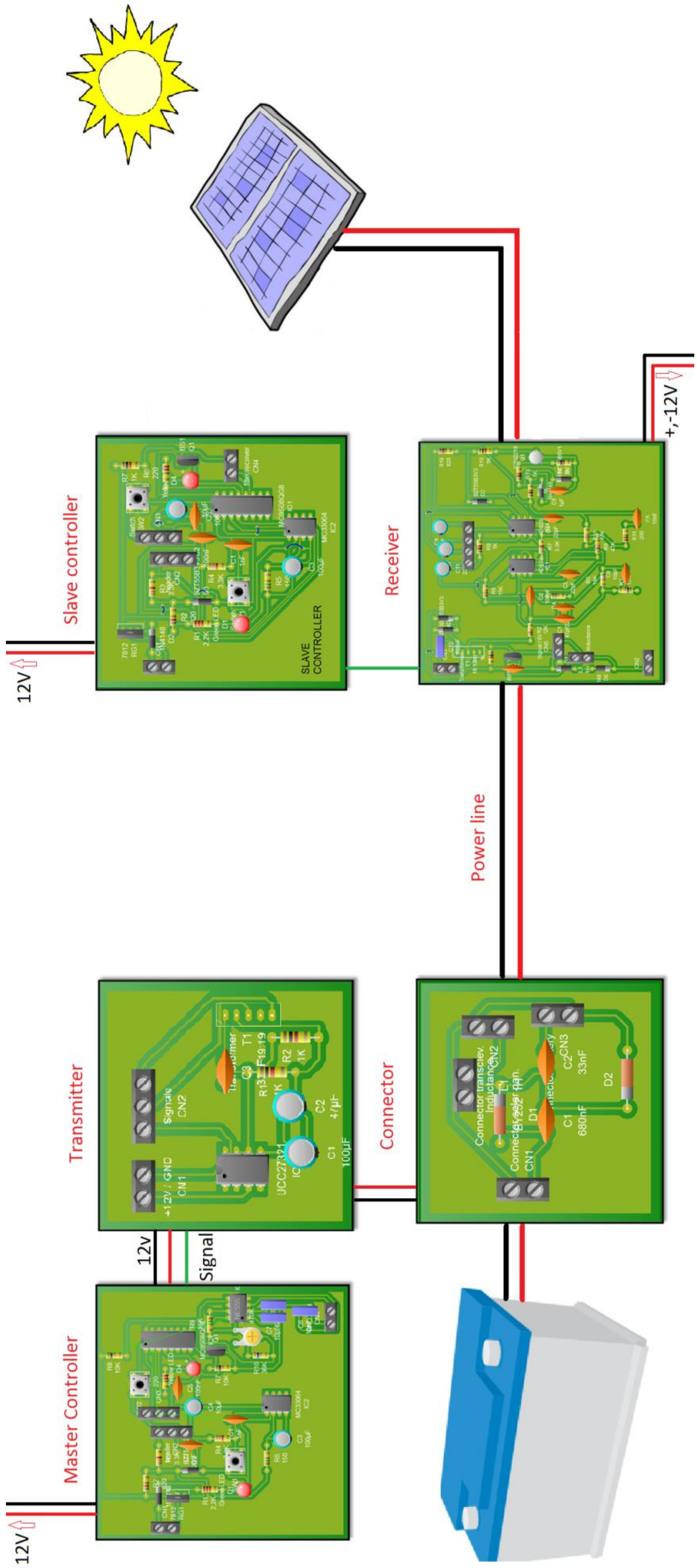
Picture 105 Transmitter



Picture 106 Connector




Picture 107 Receiver - Filter – Trigger




Picture 108 General sketch

## 6.15 Appendix 15 List of the components used

| Master controller | | | |
|---|---|---|---|
| Component | Name | Value | Number |
| Resistors | | 2,2 KΩ | 1 |
| | | 820 Ω | 1 |
| | | 3,3 KΩ | 2 |
| | | 36 kΩ | 1 |
| | | 150 Ω | 1 |
| | | 15 kΩ | 1 |
| | | 220 Ω | 1 |
| | | 10 kΩ | 2 |
| Potentiometers | | 470 kΩ | 1 |
| Capacitors | | 100 uF | 1 |
| | | 10 nF | 1 |
| | | 47 pF | 1 |
| | | 10 pF | 1 |
| | | 1 nF | 1 |
| | | 100 nF | 2 |
| | | 10 uF | 1 |
| Zeners | BZT55B3V3 | 3,3 Vz | 1 |
| Diodes | 1N4148 | | 1 |
| Transistors | BS170 | | 1 |
| LEDS | | Green | 1 |
| | | Yellow | 1 |
| Headers | | 6 pins | 1 |
| Voltage references | 7812 | 12 V | 1 |

| | | |
|---|---|---|
| Switches | Normally open | 2 |
| IC | MC9S08QG8 | 1 |
| | NE555 | 1 |
| | MC33064 | 1 |
| Connectors | 2 pin terminal block | 2 |

| Transmitter | | | |
|---|---|---|---|
| Component | Name | Value | Number |
| Capacitors | | 680 nF | 1 |
| | | 33 nF | 1 |
| Inductances | | | 1 |
| Diodes | 1N4148 | | 1 |
| | 1N4007 | | 1 |
| Connectors | 2 pin terminal block | | 3 |

| Slave controller | | | |
|---|---|---|---|
| Component | Name | Value | Number |
| Resistors | | 2,2 KΩ | 1 |
| | | 820 Ω | 1 |
| | | 3,3 KΩ | 2 |
| | | 150 Ω | 1 |
| | | 220 Ω | 1 |
| | | 1 kΩ | 1 |
| Capacitors | | 100 uF | 1 |
| | | 1 nF | 1 |
| | | 100 nF | 2 |
| | | 10 uF | 1 |
| Zeners | BZT55B3V3 | 3,3 Vz | 1 |
| Diodes | 1N4148 | | 1 |
| Transistors | BS170 | | 1 |

| | | | |
|---|---|---|---|
| LEDS | | Green | 1 |
| | | Yellow | 1 |
| Headers | | 6 pins | 1 |
| Voltage reference | 7812 | 12 V | 1 |
| Switches | Normally open | | 2 |
| IC | MC9S08QG8 | | 1 |
| | MC33064 | | 1 |
| Connectors | 2 pin terminal block | | 2 |

Receiver_Filter_Trigger

| Component | Name | Value | Number |
|---|---|---|---|
| Resistors | | 1 kΩ | 5 |
| | | 12 kΩ | 1 |
| | | 11 kΩ | 2 |
| | | 10 kΩ | 1 |
| | | 19,1 kΩ | 1 |
| | | 3,3 kΩ | 1 |
| | | 47 kΩ | 2 |
| | | 200 Ω | 1 |
| | | 3 kΩ | 1 |
| | | 820 Ω | 1 |
| Capacitors | | 680 nF | 1 |
| | | 1 uF | 2 |
| | | 47 pF | 1 |
| | | 22 pF | 4 |
| | | 100 pF | 1 |
| | | 10 pF | 1 |
| | | 22 uF | 3 |
| | | 10 nF | 1 |
| Inductances | | | 1 |
| Transformers | | 19:19 | 1 |
| Zeners | BZT55B3V3 | 3,3 Vz | 3 |
| | BZT55B5V1 | 5,1 Vz | 1 |
| | BZT55B18 | 18 Vz | 1 |
| Diodes | 1N4148 | | 2 |
| Transistors | IRF520 | | 1 |
| | Q2N2219 | | 1 |
| IC | AD829 | | 2 |
| Connectors | 2 pin terminal block | | 3 |

| | |
|---|---|
| 4 pin terminal block | 1 |

## 6.16 Bibliography

### Bibliography

Electronic Engineering and Digital Electronics ,John H. Davies, May 21, 2004, Department of Electronics and Electrical Engineering, Glasgow University

Christiansen, Donald; Alexander, Charles K. (2005); Standard Handbook of Electrical Engineering (5th ed.). McGraw-Hill, ISBN 0-07-138421-9

Electrotechnical AC-DC, version 4,Fowler R., 1999, Tziolas, Greek version, ISBN: 960-7219-78-3

Project management appendix, Ph.Clermont, Ph Fillatreau, ENIT

Project management IN A SME appendix, Ph. Fillatreau, ENIT

Data structures course, Katsavounis 2010, Democritus University of Thrace

Electronics course, Karakatsanis 2010, Democritus University of Thrace

Electronic Engineering and Digital Electronics ,John H. Davies, May 21, 2004, Department of Electronics and Electrical Engineering, Glasgow University

Juan Gámiz Caro. Redes de comunicaciones industriales. 2004. (Asignatura optativa de intensificación. Plan 2002). UPC.

Juan Gámiz Caro. Teoria bloque 1, 2 , 3 y 4. Febrero 2010. CONTROL ELECTRÓNICO DE PROCESOS INDUSTRIALES (Asignatura optativa de la especialidad de Electrónica, plan 2002).UPC.

Herminio Martinez. Técnicas de modulación. (Comunicacions industrials). 4º cuatrimestre. Pla 2002. Enero 2008. Versión 2.0. UPC.

### Articles & Patents

The Power Line Transmission Characteristics for an OFDM Signal, A. Mori, Y.Watanabe, and M. Tokuda, Musashi Institute of Technology, Japan K. Kawamoto, Sumitomo Electric Industries

Method for transmitting a signal via a powerline network, Lothar stadelmeier stuffgart, Andreeas Schwager, Duniel Schnelder, Mar. 11 2010, Pub no. US 2010/0061433 A1, United states Patent Application Publication

BI – Directional universal serial bus booster circuit,Justin P. BANDHOLZ, Moises Cases, Bradley D. Herman, Erdern Matoglu, Bhyrav M. Mutnury, Thomas D. Pahel, Pravin S. Patel, Nam H. Pham, Christopher C. West, July 1 2008, Pub no. US 7,394,281 B1, United states Patent Application Publication

Power line communication devide with DC power output and network signal transmission abilities, Shiann-Chang Yeh, Tzu-Hsiu Hsiao, Chien-Hong Lin, ACBEL POLYTECH INC, AH04L2700FI

Power line networking, apparatus and method , Frank Sacca, Diamond Bar, Alberto Mantovani, Laguna Niguel, October 04 2000, Appl. No. 09/678,983

Power line communication system, Matthew G. Sargeant, Michael A. Neal, June 28 1993, Appl. No 84,711

Signal mixers, Liam Devlin

Maxin engineering journal, volume 63

Understanding Small Microcontrollers, Motorola

Microcontrollers, Digital Electronics Tutorial Sheets

## Websites

### *Backsheet*

http://www2.dupont.com

http://solargard.com

http://solutions.3m.co.uk

http://de.promo.web.id/entertainment/how-to-send-high-frequency-signals-through-low-frequency-transformers

http://www.freecircuits.net

***Broadband***

http://neuralenergy.blogspot.com/2009/06/powerline-communications.html

http://www.infocellar.com/networks/new-tech/BPL/BPL.htm

http://www.grc.nasa.gov/WWW/RT/RT1998/5000/5650kifle.html

http://www.faqs.org/patents/app/20090154594#ixzz17tlXK1vt

http://en.kioskea.net/faq/4137-plc-power-line-carrier

http://www.echelon.com/products/transceivers/powerline/default.htm

http://machinedesign.com/article/your-next-network-connection-could-be-a-powerline-0824

http://es.kioskea.net/contents/cpl/cpl-architecture.php3

http://www.caslon.com.au/powerlinenote.htm

http://www.yamar.com/benefits.html

http://dspace.mit.edu/handle/1721.1/27144

**Switch**

http://www.electronics-tutorials.ws/transistor/tran_7.html

***Converter***

http://www.unicrom.com/Tut_reg_con_zener.asp http://www.unicrom.com/Tut_ReguladoresTransistorizados.asp

### *Mosfet & IGBT*

http://www.semikron.com/skcompub/en/application_manual-193.htm

## Reference documents

MAX2769 GPS Reference Design, Maxim

GPS USB Reference Design with the MAX2769, Maxim

IQ GPS receiver reference, Lassen

IEE 802.15.4 Technology, Freescale

IEE 802.15.4 Development, Freescale

## Datasheets

### *Converter*

Converter model MC34063A, Texas Instrument

Converter model MC34063A, Micrel

### *Thyristor*

Thyristor model BT151, NXP

### *MOSFET*

MOSFET model SUP57N20-33, Vishay

MOSFET model FDP39N20 / FDPF39N20, Fairchild

MOSFET model IRFR4620PbF, International Rectifer

MOSFET model IRFR4615PbF, International Rectifer

MOSFET IRF520

MOSFET BS170

### *IGBT*

IGBT model STGD10NC60H, ST

IGBT model STGB6NC60HD, ST

### *Solar panel*
Sharp panels, NT & NU series

Sunpower panels, SPR & E18 series

Sunmodule panels, SW series

Table of cells solarcom

### *GPS*
GPS model max2769, Maxim

GPS model MN5010HS, Micromodular technologies

GPS model Ls20030, Locosys technologies

### *WIFI*
WIFI models MC1319x, MC1320x & MC1321x, Freescale

### *Microcontroller*
Microcontroller HCS08, Freescale

Microcontroller HCS12, Freescale

### *PLC Comunication*
PLC model  ST7570, ST

PLC model  ST7540,ST

PLC model 2991, Maxim

PLC Model SIG60 & SIG61, Yamar

### *RF*
RF model MO-RX3400-A & MO-SAWR-A, Holy Stone Enterprise

### *More*
Voltage Regulators LM7812

BIPOLAR TIMERS NE 555

UCC 27321 high speed mosfet driver

Amplifier AD829