TFE4850 - EiT - Student satelite

Groundstation network

Group 2
Eirik Skjeggestad Dale
Hanne Thorshaug Andresen
Marius Ekerholt
Børge Irgens
Leif-Einar H. Pettersen
Hallstein Skjølsvik

March 20, 2013

Abstract

This project aimed to create and test a BlueBox, an USB-driven ground station, developed for AAUSAT3 by Aalborg University to send and recieve signals from satelites. The NUTS project have a problem with low transfer rate between the single ground station at the university and the satelite, and by setting up a network of BlueBoxes at different ground stations around the globe, we can increase the time which the satelite is available for data transfering.

Since setting up the network proved too time consuming, we set as our goal to make a BlueBox, and test that it can recieve signals from a satelite, and in this way make it easier for others to set up the network, in that they already have documentation on how to set up and configure each indiviual ground station.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter will contain a short introduction to our project. That means what we're doing and what people are working on the project. It will also be discussed some short background on why we're doing this project.

1.1 The Project group

The project group consisted of 6 people from 5 different directions, so we had a wide variety of competance to use in the project. The group members can be seen in **??**.

Name	**Background**
Marius Ekerholt	Computer technology
Eirik Skjeggestad Dale	Computer technology
Hanne Thorshaug Andresen	Energy and Environmental technology
Leif-Einar H. Pettersen	Electronics and Telecommunications
Børge Irgens	Theoretical Physics
Hallstein Skjølsvik	Electronics and digital design

Table 1.1: Group members

Chapter 2

Theory

Before we started our project, we did quite alot of prestudy, covering which technology was available, to determine which fit our project best. This chapter will contain a summary of the different technologies, and also some theory behind the reasoning for having multiple ground stations listen to our satelite.

2.1 Communicating with the satellite

Since we don't know when the satellite will be launched we don't know the exact orbit. The project manager, Roger Birkeland, told us that the orbit will have $h \in (450km, 650km)$ and $i = 98$. The uncertainty in h is critical, because the height above the earth dramatically changes the access time of a ground station, see Fig. **??**.

A ground station can only communicate with a satelitte when it has a certain elevation. This elevation can differ from case to case depending on atmospheric effects, frequency and more. In the following we assume that the minimum elevation is 25 degrees, maximum elevation is 90 degrees and no constraints on the azimuth angle. For a illustration see Fig. 2.2.

The result of this is that the ground station can communicate with the satelitte whenever the ground track is inside a rough circle centered on the ground station, see Fig. 2.3 for the estimated "range" of the ground station at Gløshaugen operating with these constraints.

This ground station will on average be able to communicate with the satellite t seconds per day. That means that we will be able to download

$$D = d \cdot t \tag{2.1}$$

where d is the bandwidth bandwidth available for transferring scientific data. d is assumed to be 2.4 kbps. If the satellite has a orbit with $h = 450km$ we can communicate with the satellite 540 seconds per day on average which gives 1.3 Mbit of data per day. If the satellite has a orbit with $h = 650$ we

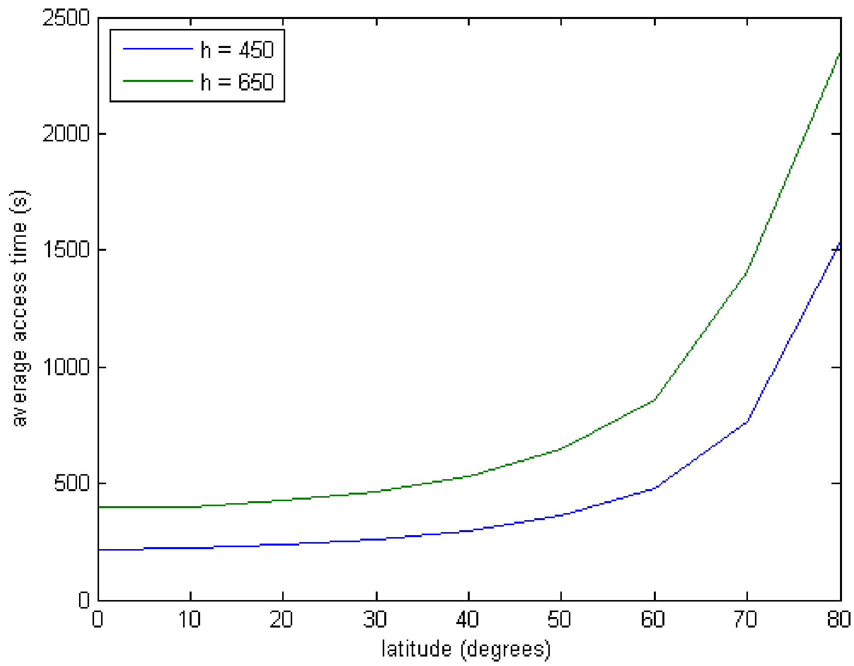

Figure 2.1: Access times for $h = 450km$ and $h = 650km$ as a function of ground station latitude

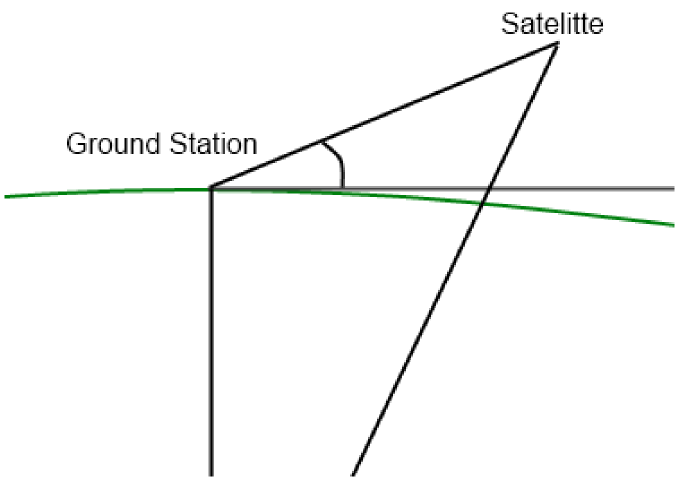

Figure 2.2: Illustration of geometry between a ground station and a satellite

Figure 2.3: NTNU ground station range

can communicate 980 seconds on average, which gives 2.4 Mbit. The amount of data we need to download per day is unknown, but the project manager told us 1 MB per day would be a good target.

2.2 Network technology for ground stations

When we decided to work on a network of ground stations, we first looked into four different ground station network technologies. We first hoped to work on a BlueBox, but this would require support from Aalborg that we couldn't get, as they were busy with a satelite launch of their own. Because of this, we decided to look into Carpcomm, that seemed more complete and doable than connecting to PYXIS with a BlueBox.

2.2.1 Pico

information about Pico

2.2.2 Genso

information about Genso

2.2.3 Carpcomm Space Network

Carpcomm is a private company that delivers a plug and play ground station [3] that costs $700. The software for the ground station is open source and is provided pre.compiled for x86 and arm debian. It is compatible with the Carpcomm Space Network [4]. The advantage of using this solution is that the network is actually functioning, though there are few other operational ground stations.

2.2.4 PYXIS

The BlueBox is part of a distributed ground station network called PYXIS, developed primarily for the AAUSAT3 by Aalborg University (2013 [2]). The PYXIS goal is to offer a robust and effective ground station network for satellite developers, and one of the key factors is that everyone is free to setup a ground station using the open source BlueBox hardware.

The PYXIS concept includes a backend server, BlueBox hardware and a Ground Station Server (GSS). The backend server runs an individual instance for each satellite utilizing the BlueBox, and is operated by the persons responsible for the ground station.

The BlueBox itself is hardware to recieve and transmit signlas from the satellites.

Control of the BlueBox and ground station mechanics is handled by the GSS, and both the BlueBox and the GSS is operated by the responsible for the Ground station.

Both the backend server and the GSS is already in place at each ground station, and to join the PYXIS network we would only have to make a BlueBox, and test that it works.

2.3 Raspberry pi

Raspberry Pi is a small computer, with everything gathered in one board. In our project we will be using the B model, revition 2, which have a 700MHz ARM CPU, 512MB of RAM and a SC-card reader, in addition to the leads to connect to different devices, for the full overview, see figure 2.4. The recommended operating system is Raspbian, a linux distribution based on Debian.

The Raspberry pi was originally intended to help teach programming, but it can also perform many of the standard computer tasks, and it can be connected to a monitor or tv using an HDMI lead. In our project we hope to be able to use a Raspberry pi to run the software required to control the ground stations. The software provided by the carpcomm project has Raspbian as one of its supported plattforms, so we hope this will work well.

To control the movement of the antennas, an serial port is needed. Raspberry Pi has an serial port included in its gpio (general purpose input/output) connector. This serial port uses ttl-standard for its voltage levels, this is 0/3.3V while RS232 which is the standard used in computers uses (3V-15V)/-(3V-15V). Because of this an converter is needed. We chose to make an custom circuit board using the MAX3232 RS232 line driver. The circuit board is designed to be mounted on the gpio connector, because of small space in the case for the Raspberry Pi, the output is connected with cable to the external connector.

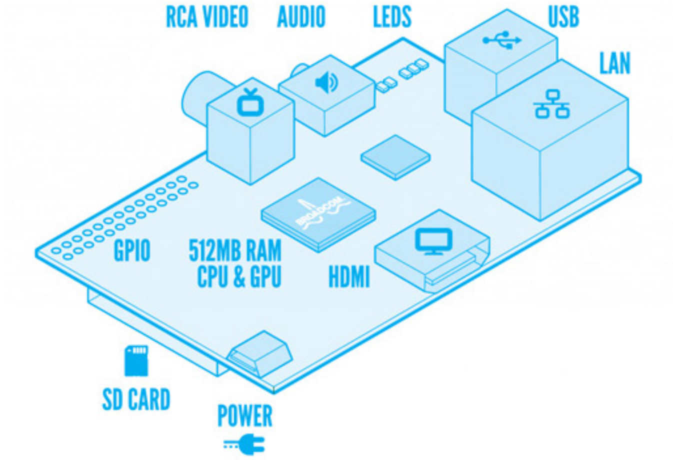

Figure 2.4: A highlevel schemantic of the Raspberry pi, mobel B rev 2

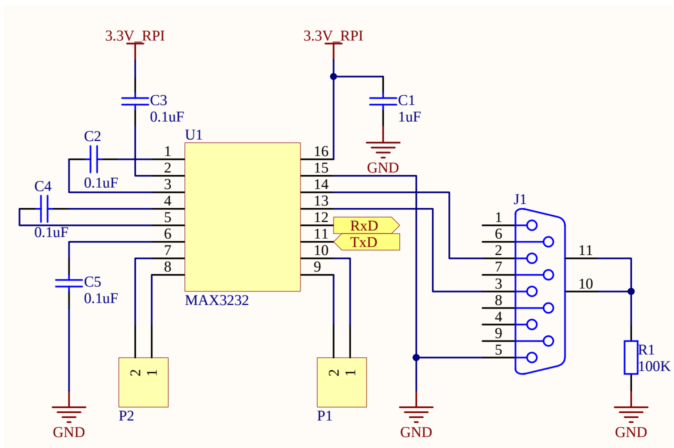

Figure 2.5: Schematics for the RS232-converter

2.4 SunPower

When a satellite communicates to its groundstation and operates its payload it is dependent on power supply. Without any electrical power a satellite will have noe function other than to drift around in orbit unable to communicate. NUTS double CubeSat uses sunlight as an energy source through solar panels.

Solar panels base their operation on the ability to convert sunlight into electricity by exploiting the photovaltaic effect by using semiconductors. The convertion process where the suns radiation is converted into an electical current is achieved by creating mobile charged particles in the semiconductor. They are in turn separated by the device structure and produce the electrical current.(kilde) The use of photovaltaic solar generators is the best choice for providing electrical power to satellites in an orbit around the Earth (kilde: Solar Electricity, page 180, kilden skal skrives ordenlit!).

To get information on how long communication time the CubeSat can achiece through a ground station network the energy supply must be calculated.

Solar power calculations:
Solar constant: $S = 1367\frac{W}{m^2}$
Area of the Earth presented to the Sun: A_S
Area of the Earth: A_S

Total energy flux on the Earth: $\boldsymbol{S_E \times A_E = S \times A_S}$

$$S_E = \frac{S \times \pi \times R^2}{4 \times \pi \times R^2}$$
$$S_E = \frac{S}{4} = 342\frac{W}{m^2}$$

Orbitale period of the satellite: $T = 5700sek$ (from Atg simultation program)
Efficiency of the solarcells: $\eta = 0.16$
Average area exposed to the sun: $A_Sat = 0.016213185m^2$
Input energy to the satellite from the sun: $E[J]$
Power from the sun to the satellite: $P[W]$

$\boldsymbol{E = A_Sat \times S \times T \times \eta}$
$E = 20213.04J$
$\boldsymbol{P = A_Sat \times S \times \eta}$
$P = 3.55W$

Chapter 3

The project

Bibliography

[1] *Navn på item*. October 07 2012. Available at: <www.wikipedia.org>

[2] *AAUSAT3*. February 13 2013. Available at: <http://www.space.aau.dk/aausat3/>

[3] *Carpcomm Ground Station 1*. October 2012. Available at: <http://carpcomm.com/gs1/>

[4] *Carpcomm Space Network*. Available at: <http://carpcomm.com/howitworks.html>

[5] *Raspberry pi*. Available at: <http://www.raspberrypi.org/>