Attitude determination of the NCUBE satellite

Kristian Svartveit

Department of Engineering Cybernetics
June, 2003

NTNU
Norges teknisk-naturvitenskapelige universitet

Fakultet for informasjonsteknologi, matematikk og elektroteknikk
Institutt for teknisk kybernetikk

HOVEDOPPGAVE

Kandidatens navn: Kristian Svartveit

Fag: Teknisk Kybernetikk

Oppgavens tittel (norsk): **Estimering av attityde for NCUBE satellitten**

Oppgavens tittel (engelsk): **Attitude determination of the NCUBE satellite**

Oppgavens tekst:

1. Do the final choice of actuators and sensors for NCUBE
2. According to the NCUBE project plan; write an implementation specification for ADCS for NCUBE
3. Study the use of Pulse Width Modulation for control of current in the coils. Find upper and lower bounds on the frequency of modulation.
4. Finalize the method of using the solar panels as sun sensor. Test this experimentally.
5. Design a Kalman filter for estimation of attitude and angular velocity. The filter should be based on measurements from magnetometer and sun sensor.
6. There is a need for an orbit estimator in order to calculate the position of the satellite. This can be done by e.g. using the SGP4 algorithm. However, this algorithm requires large computational recourses. With this in mind, design a simplified orbit estimator suited for NCUBE.

Oppgaven gitt: 6. Januar 2003

Besvarelsen leveres: 20. Juni 2003

Besvarelsen levert:

Utført ved Institutt for teknisk kybernetikk

Trondheim, den 6. Januar 2003

Jan Tommy Gravdahl
Faglærer

Summary

This master thesis describes the different sensors and actuators suitable for attitude determination and control of a picosatellite. The sensor that is to be used on the NCUBE satellite is a three axis magnetoresistive digital magnetometer from Honeywell. The International Geomagnetic Reference Field will be used as comparison to obtain attitude information. The possibilities of aiding the magnetometer with solar panel measurements is investigated, and preliminary results looks promising. A simple orbit estimator with good performance is implemented to provide position information to the magnetic reference field and the reference frame transformations. The satellite will be actuated using magnetic coil torquers. They will be controlled using pulse width modulation similar to motor control. The NCUBE satellite is modelled in Simulink together with a Kalman filter for attitude estimation based on magnetometer measurements. The theory behind this, and behind extending the Kalman filter to include solar panel measurements, is also presented.

Contents

Chapter 1

Introduction

The Cubesat concept on which the NCUBE is based, is briefly presented. The NCUBE, and some of its features, is presented to provide the setting for the work presented in this thesis. The motivation behind this master thesis is discussed, and the chapter concludes with the thesis' outline and main contributions.

1.1 Background

1.1.1 Cubesat

In order to have his students complete a satellite project during their educational time, preferably in just a year, professor Bob Twiggs at Stanford University came up with the Cubesat concept. The main idea was to make a very small standardized satellite, and this resulted in the $10 \times 10 \times 10$ cm cube weighing in at under one kilogram. The launcher was also designed to launch multiple satellites in order to reduce launch costs. The design of the launcher, see right hand side of Figure 1.1, also enables the possibility to launch a satellite twice the length, and one standard length satellite or similar combinations. See Appendix C for some more information on the Cubesat from the 5th ESA Conference on Spacecraft Guidance, Navigation and Control Systems, ESA GNC Conference.

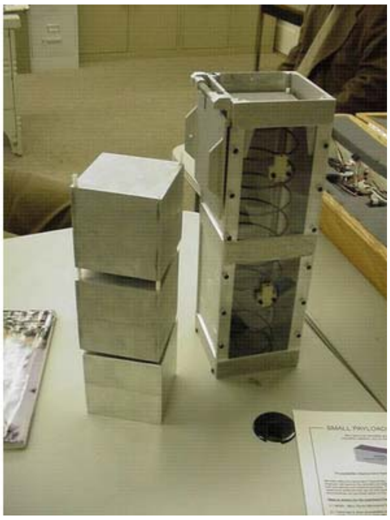


Figure 1.1: The NCUBE cubesat and the cubesat launcher

1.1.2 NCUBE

Andøya Rocketrange, ARS, and the Norwegian Space Centre, NRS, have evaluated the possibilities of starting a Norwegian student satellite programme, and taken the initiative to start such a programme. The goal is to build competence in space related fields within teaching institutions, and among students, while at the same time building competence for future Norwegian satellite activity. The goal will be obtained by designing, building, testing, and launching a small satellite based on student work at Norwegian universities and colleges. Based on the success of this project, future projects would also be feasible.

A satellite project will increase the interest in scientific fields among Norwegian students, by letting the students take part in a practical technical project, whilst at the same time studying related satellite technology and theory. Resources from Norwegian industrial and research environments are involved, and give support within their fields of expertise.

The project is expected to last until the second half of 2003, with a launch to a polar orbit as the goal. The project is now in it's final implementing phase, to be finished by 2003. Students at the Norwegian University of Science and

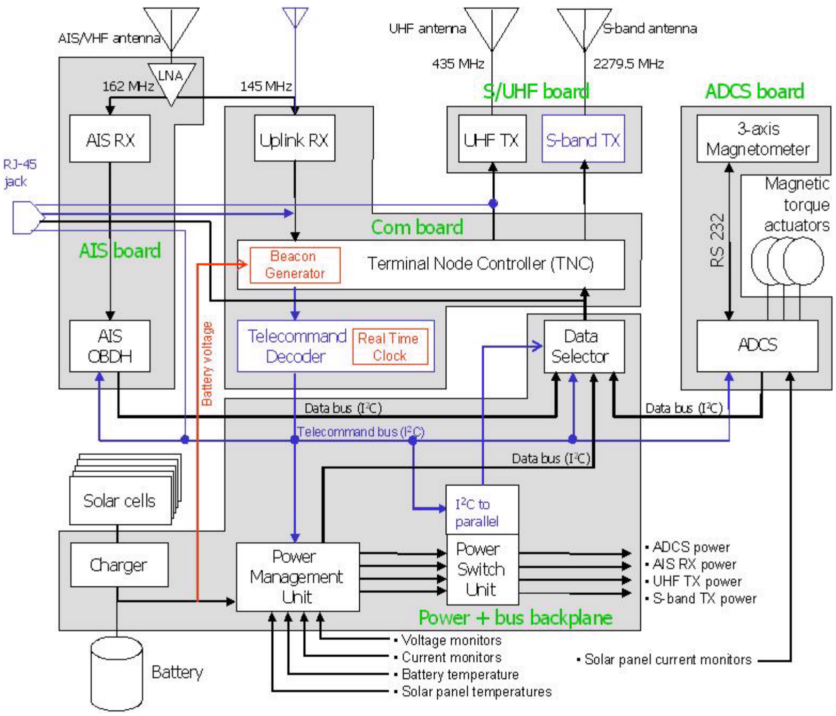


Figure 1.2: The NCUBE system architecture

Technology, NTNU, the University College of Narvik, HiN, and the Agricultural University of Norway, NLH, have already performed pre-studies, and student groups at these universities and from the University of Oslo, UiO, take part in ongoing projects. 7 students at NTNU, 8 at NLH, 1 at UiO and 6 at HiN are involved in the spring semester of 2003. The project has been split into 7 subgroups, where NTNU will study the areas of Structure, Power, Attitude Determination and Control Systems, Communications and downlink, On Board Data Handling, and technical framework for payload. HiN will study Power and Ground Segment/Uplink. UiO will perform testing and study power supply with emphasis on the solar panels, and NLH will study payload and applications. Resources from the Norwegian Defence Research Establishment, FFI, Kongsberg Defence & Aerospace, Kongsberg Satellite Services, KSAT, Telenor and Nammo Raufoss support the students, and give valuable comments and feedback, in addition to input from internal expertise from ARS, NRS, NTNU, HiN and NLH.

The satellite named NCUBE is based on the Cubesat concept, and the payload has been decided to be an automatic identification system, AIS. The AIS is a mandatory system on all larger ships that transmits identification and position data messages on 162 MHz maritime VHF band. The NCUBE will fetch such messages sent from ships, and from reindeer collars produced by NLH students. The system architecture with all the different subsystems is shown in Figure 1.2.

Launch of Cubesat satellites are organized, and relatively low-cost, through existing concepts. The Norwegian student satellite launch will be with other student satellites, and foreseen in the first half of 2004. Many universities are already performing Cubesat related projects, and the potential of international exchange of information is large, and easily available through internet.

1.1.3 This thesis

In order to utilize the broadband downlink antenna, the NCUBE satellite must have attitude control. The antenna will have a 10-20 degree beam, so attitude control with accuracies in this region is necessary. To obtain this accuracy an attitude determination and control system, ADCS, is needed. This document is based on the work done to establish which sensors and actuators to use on the NCUBE, and what hardware and software needed to utilize them. Also, the design of the Kalman filter that is to be used to combine the available measurements is described. A number of implementation issues are also discussed in order to realize the theory on the NCUBE satellite

1.2 Outline and contributions of this thesis

This document is mainly a compilation of the information needed for attitude determination of small satellites in general, Cubesats in particular. The main contribution of this document lies in the determination of the sun vector from solar panel efficiency measurements, and the extension of the Kalman filter to incorporate this. Also an orbit estimator requiring less computational force while still producing good estimates is developed, and implemented in Matlab. The satellite model with its environment and the structure of the magnetometer Kalman filter is also implemented in Matlab/simulink. The thesis is part of the background for the publications at the 17th AIAA/USU Conference on Small Satellites, and the 54th International Astronautical Congress. The abstract of the former, and the complete latter is enclosed in Appendix E.

Chapter 2 Briefly presents the different sensors and actuators used in attitude determination and control systems, ADCS, from a Cubesat implementation point of view. The sensors and actuators chosen for the NCUBE satellite is somewhat more thoroughly presented. For both the sensors and actuators, a table presenting the performances and advantages/disadvantages is introduced in summary Chapters 2.1.7 and 2.2.7.

Chapter 3 Defines the notation used in the document. It also defines the mathematical background on which the mathematical modelling in Chapter 4 is based. Rotation matrices is introduced, as is euler parameters with their advantages in attitude representation.

Chapter 4 Describes the mathematical modelling of the chosen sensors, actuators, orbit estimators, and the magnetic reference model. It also contains a derivation of all the rotation matrices needed to bring all references to the same frame; the orbit frame. The chapter also includes the modelling of the satellite's dynamics and it's environment.

Chapter 5 Outlines the theory of Kalman filters, and extended Kalman filters for nonlinear systems. The derivation of Matlab/Simulink implementations of the Kalman filter is also presented.

Chapter6 Takes into consideration some of the implementational issues of the ADCS on NCUBE. Micro controller architecture and interfacing with sensors, actuators, and other parts of the satellite is described.

Chapter 7 Sums up the results of this document, and gives recommendations for future work and development.

1.2.1 Outline of the appendices

This section gives an overview of the appendices and their content. Some of the appendices is an important part of the work in this thesis, but is nevertheless more suited to be placed in an appendix. Other parts are background material useful mainly for implementational issues.

Appendix A The definition of the Norad Two-Line-Element set, TLE, as defined on the website of Celestrak (2003). Both the structural definition, and descriptions of the parameters, are included.

Appendix B All matlab source code and simulink diagrams used to produce the results presented in this document. The initial satellite and environment model included was, as described in Chapter 4.5, done together with K.M Fauske and F.M. Indergaard.

Appendix C The report from the 5th International ESA Conference on Guidance Navigation and Control Systems, where several small satellite concepts were presented. Founder of the Cubesat concept, professor B. Twiggs, also introduced his concept and visions for the future of student built satellites. A workshop on Cubesats was held where both Italian and German universities attended.

Appendix D Documentation of the work done by the ADCS group; K.M Fauske , F.M. Indergaard and the author. This document is the base for the implementation of the ADCS on the NCUBE.

Appendix E The abstract of the accepted paper for the 54th International Astronautical Congress, Bremen, Germany, and the paper submitted to the 17th AIAA/USU Conference on Small Satellites, Utah, USA.

Appendix F The datasheet for the chosen three axis digital magnetometer from Honeywell (2003)

Chapter 2

Sensor and actuator evaluation

In this chapter some sensors and actuators and their performance are described. The infeasible sensors and actuators are only briefly mentioned, while the recommended ones are more thoroughly presented and discussed. This is meant as an assistance in deciding which devices to use, and which not to use in Cubesat ADCS.

2.1 Sensors

All the presented attitude sensors, except the gyroscope, are reference sensors. Reference sensors give a vector to some object which position is known. The rotation between the local body frame, and the frame in which the known vector is given, can then be computed. With only a single measurement, the rotation around the measured vector is unknown. It is therefore necessary either to have two different measurements, or to utilize information from the past. The most common way to incorporate measurement history, is to combine the measurements in a Kalman filter. This is further discussed in Chapter 5.

2.1.1 Magnetometer

A magnetometer can only be used in orbit close to earth where the magnetic field is strong and well modelled. As the NCUBE is to orbit at approximately 700km, a Low Earth Orbit, LEO, this is feasible. The magnetometer consists of three orthogonal sensor elements which measure the earths magnetic field in three axes in the sensor frame. If the magnetometer is aligned with the satellites axes, or the rotation between the body and sensor frame is known, the magnetic field in the body frame is obtained. This measurement is compared in the Kalman filter, as described in Chapter 5, to a model of the earths magnetic field which gives the magnetic field in orbit coordinates. The most used model is the International Geomagnetic Reference Field, IGRF, which is an empirically developed model presented in Chapter 4.3.
The accuracy of the magnetometer is, according to Bak (1999), limited mainly by three factors:

- Disturbance fields due to spacecraft electronics

The magnetometer will measure not only the earth's magnetic field, but also the satellite's. If the satellite is not magnetically clean, the performance of the attitude estimation will degrade. With this in mind, it is important to consider magnetic radiation and shielding when implementing satellite hardware, and different component's orientation in relations to each other.

- Modelling errors in the reference field model

As the model is just that, a model, it will never be totally accurate. This will induce errors in the attitude estimation. The error becomes smaller the better the model being used is, hence the IGRF model will give better results than using a simple dipole model. Errors in the estimate of the satellite position will also induce errors as the measurement is compared to the magnetic field at another position.

- External disturbances such as ionospheric currents

As the ionosphere is inherently unpredictable, and current flowing in it produces magnetic disturbances, it is not possible to predict the influence of this on the final attitude estimate.

The different types of magnetometers

Induction Coil Magnetometer The induction coil is one of the most widely used vector measurement tools, as well as one of the simplest. It is based on Faraday's law, which states that a changing magnetic flux, ϕ, through the area enclosed by a loop of wire, causes a voltage e to be induced which is proportional to the rate of change of the flux

$$e(t) = -\frac{d\phi}{dt} \tag{2.1}$$

The magnetometer thus consists of one or more coils, and voltage measurements are used to calculate the magnetic field.

Fluxgate Magnetometer A fluxgate magnetometer is popular for use in many applications. Characteristically, it is small, reliable, and does not require much power to operate. It is able to measure the vector components of a magnetic field in a 0.1 nT to 1 mT range. The fluxgate magnetometer is a transducer which converts a magnetic field into an electric voltage. Fluxgates are configured with windings through which a current is applied. If there is no component of the magnetic field along the axis of the winding, the flux change detected by the winding is zero. If there is a field component present, the flux in the core changes from a low level to a high level when the material goes from one saturation level to another. From Faraday's law, a changing flux produces a voltage at the terminals of the winding proportional to the rate of change of the flux.

SQUID Magnetometer The Superconducting Quantum Interference Device, SQUID, magnetometer works on the principle that the magnitude of a superconducting current flowing between two superconductors separated by a thin insulating layer is affected by the presence of a magnetic field. These magnetometers are the most sensitive devices available to measure the magnetic field strength. However, one disadvantage is that only the change in the magnetic field can be measured, instead of the absolute value of the field.

Magnetoresistive Gaussmeter Magnetoresistive gaussmeters utilizes the resistivity of a ferromagnetic material which changes under the influence of a magnetic field. The amount of change is based on the magnitude of the magnetization, as well as the direction of flow of the current to measure resistivity.

Magnetoresistive magnetometers from Honeywell

Although the fluxgate magnetometer gives the best performance with very low power consumption, the magnetoresistive magnetometers are smaller in size and therefore preferred. Honeywell (2003) produces an array of magnetoresistive magnetometers, three of which is presented below.

The integrated circuit, IC, magnetometer HMC 1001/1002 Single chip one-axis and two-axis magnetoresistive magnetometers, respectively. They measure both only 1.9x2.54 cm. Using these IC sensors will require some more computational force in the satellites internal computer, and to obtain 3D measurements of the magnetic field at least two of them are necessary. When mounting these ICs on circuit boards on board the spacecraft, close attention must be paid to get them perfectly perpendicularly aligned. If they are misaligned, the measurement will be of very little value, at least when the misalignment is not known. The price on orders directly from the Honeywell website (2003) is 20$ for the one-axis and 22$ for the two axis IC.

The analog three-axis magnetometer HMC 2003 Also a one-chip sensor with a 2.54x1.9cm footprint weighing only a few grams, and using approximately 0.2 watts maximum. This device has three analog outputs giving full three-dimensional vector measurements of the magnetic field. The device is also equipped with the possibility of setting an offset on each of the three axes. This enables the possibility of measuring the magnetic field and applying a control torque simultaneously, as the magnetometer can be offset to compensate for the applied magnetic field. If other on board equipment produces well known magnetic field this could also be compensated for. However, it is not likely that such fields are well known or modeled. The magnetometer requires an external degaussing circuit providing a high voltage peak to reset the magnetometer. The price is 199$.

The digital smart magnetometer HMR 2300 The same magnetometer as the HMC 2003, but mounted on a 7.49x3.05 cm circuit board weighing 28g. This board gives the magnetometer a digital interface with a 9600 baud serial RS232 communication through a nine pin connector, and it eliminates the need for an external degaussing circuit. There are no offset possibilities, but as long as the total magnetic field is within the magnetometers range, the offset could be done in the satellites on board microcontroller. The price, including test and simulation software, is 750$, and without it's 675$. The data sheet is enclosed in Appendix F

The choice There are several factors influencing the final choice. For simplicity, and for avoiding the chance of misaligning the IC sensors, it would probably be best to use one of the three axes magnetometers, even though the prices are higher. The advantage in size and offset possibilities favors the analog magnetometer, as long as one has enough analog inputs and outputs on the microcontroller. But as testing done by Busterud (2003) shows that the ADCS performance is not degraded by alternating between measuring and actuating, the benefit from the offset possibilities is not a deciding factor. The analog magnetometer requires a degaussing circuit, and if additional D/A and A/D converters is needed, the digital magnetometer might be a better solution regarding both space and cost. If the volume/mass budget allows the use of the digital magnetometer one can also benefit from the test and simulation software available from Honeywell (2003). The implementation of the entire ADCS will also be easier with the digital magnetometer, as less software for determination of the magnetic vector is needed.

2.1.2 Sun sensor

If a vector pointing towards the sun could be determined, this would aid the computing of the satellite attitude. A sun sensor in it's simplest form is a photodiode on each side of the satellite which will tell which side is most likely to be towards the sun. More advanced and accurate sun sensors are larger and heavier, and will most likely not fit into the mass budget. Another possibility for determining a sun vector is utilizing measurements of the currents from the solar panels. As there are only solar panels on five of the six sides of the satellite, a light dependent resistor, LDR, could be placed on the last side for determining whether it is pointing towards the sun or not.

2.1.3 Star tracker

The star tracker is by far the most accurate attitude determination system available, with accuracies down to a few thousands of a degree. The Charged Coupled Device, CCD, or the Active Pixel Sensors, APS, produces an image of the stars. This image is compared with an on board catalogue of the starry sky to determine the attitude. The star tracker is, however, so heavy and big, especially the baffle needed to shield the sensor from sun, earth, and moon shine, that it is infeasible for the NCUBE.

2.1.4 Horizon scanner

The horizon scanner determines where the earth is relative to the spacecraft. This is usually done by measuring the IR radiation from the earth. Usually only two axes is determined, hence this sensor is best suited in combination with other sensors. Accurate horizon scanners are relatively expensive and big sensors, and hence not suited for the NCUBE.

2.1.5 Gyroscope

The gyroscope is not a reference sensor, but an inertial sensor. It measures angular acceleration, and these measurements must be integrated twice to obtain the attitude estimate. The integration leads to a drift which leads to the need for calibration. The traditional gimballed gyroscope is too big and heavy, but a modern ring laser or piezo electric gyroscope is small enough. The problems are the prize and the drift. The gyroscope is more suited in conjunction with other sensors, or for measuring rapid changes in attitude.

2.1.6 GPS

A GPS receiver was wanted on board the NCUBE for two reasons. Primarily for the magnetometer measurement to be compared with the IGRF model which needs satellite position to determine the magnetic field. The second function to fulfill was as a payload for measuring occultations in the lower atmosphere. This second task, however, requires a dual frequency receiver if the results are to be really useful. Two frequencies receivers are too large, too heavy, too power consuming, and too expensive to fit in the NCUBE budgets.

The greatest challenge to overcome is to get hold of a receiver that violates the restrictions set by the COCOM trade agreement. They state, that a GPS receiver should not function both at altitudes over 18km, and speeds over 1000knots = 515,4 m/s. The NCUBE will at 700km orbit travel at approximate 7500 m/s, and thus exceed both limitations. The alternative to using a GPS for determining the satellite position, is to use an orbit estimator, and to update this with a satellite position measurement taken from the ground at bypass. This approach, with several orbit estimators, is discussed in Chapter 4.2.1.

A GPS receiver can also be used to determine the satellite attitude. By placing two antennas a distance apart from each other, and measuring the difference in carrier wave phase between the two antennas, the attitude, except for the rotation around the axis on which the two antennas is placed, can be determined with an integer ambiguity. This requires an additional antenna and continuous measuring which obviously requires more power.

On the 5th ESA International Conference on Guidance, Navigation and Control Systems, see Appendix C, Oliver Montenbruck, (2002), presented LEO flight experience with a GPS receiver. The GPS receiver used was the smallest available without space limitations, the ORION GPS receiver. This receiver with casing measures $5 \times 7.5 \times 12.5$cm , which is to large for the NCUBE. It could be stripped down to fit, but will still consume 2.5 W, which is too much for anything but the main payload on a Cubesat. With GPS receivers sized down to 2.5×2.5 cm, and consuming less than 100mA, Fastrax (2002), the technology regarding size allows for GPS receiver to be used in Cubesats. But as the manufacturers refuses to make GPS receivers without COCOM limitations, this remains as the greatest challenge.

2.1.7 Sensor summary

Attitude determination summary with accuracy values as suggested by Wertz and Larson (1999).

Sensor	Acc [deg]	**Pros**	**Cons**
Sun Sensor	0.1	Cheap, simple, reliable	No measurement in eclipse
Horizon Scanner	0.03	Expensive.	Orbit dependant, poor in yaw
Magnetometer	1	Cheap, continuous coverage	Low altitude only
Star tracker	0.001	Very accurate.	Expensive, heavy complex
Gyroscope	0.01/hour	High bandwidth	Expensive, drifts with time

The choice already made in the NCUBE project, which is well argued for in the above text, is to use the digital three axis magnetometer. The digital interface, which simplifies the implementation, proved to be the deciding factor, as both mass and volume budget allowed for the extra circuit board. Because of the problems concerning small GPS receivers without space limitations, a GPS is not included in the NCUBE. The possibility of using solar cells as an aid to the magnetometer is also investigated further.

2.2 Actuators

All the actuators but the magnetic torquers are only briefly described as the likelihood of them being used is small. There are experiments with pico sized Reaction Control Systems in Italy, Santoni (2002), but so far they are so big that they require to be the main payload of a Cubesat in order to fit in.

2.2.1 Magnetic torquers

Magnetic torquers enforces a torque on the satellite by creating a magnetic field which interacts with the earths magnetic field.

Torque Coils The torque coil is simply a long copper wire, winded up into a coil. The coils will produce a momentum given as

$$\mathbf{T} = \mathbf{B} \times \mathbf{M} = \mathbf{B} \times iNA, \tag{2.2}$$

where $\mathbf{B}$ is the earths magnetic field, i is the current in the coil, N is the number of windings in the coil, and A is the area spanned by the coil. The implementation and control of the magnetic coils is further discussed in Chapter 4.4

Torque Rods Alternatively, torque rods can be used. Torque rods operate on the same principle as torque coils, but instead of a big area coil the windings is spun around a piece of metal with very high permeability. Such materials are called ferromagnetic materials, and can have a relative permeability, μ, of up to 10^6. The relative permeability enters equation (2.2) so that, according to Egeland (2001),

$$\mathbf{T} = \mathbf{B} \times iN\mu A. \tag{2.3}$$

Hence, the current needed to produce the same torque is then much lower, however the weight increases drastically because of the metal core in the rods. Another inconvenience of the torque rods, is that the ferromagnetic core have memory, and hysteresis is thus introduced in the control loop, Mansfield & O'Sullivan (1998). Different ferromagnetic alloys have different hysteresis characteristics, and this, together with the permeability, must be taken into consideration in designing a magnetic torque rod. Several companies manufacture torque rods, but usually bigger than feasible on the NCUBE satellite. ZARM Technik (2003) has produced very small torque rods with near zero hysteresis and is willing to look into production of torquers with minimized weight for Cubesat fitting. They have made torque rods with a linear momentum of 0.5 Am^2, only 90mm long, and a diameter of 8.5mm. The weight is only 30g, and the current at maximum momentum is 54 mA. The good linearity of the torque rods suggests that the current for a sufficient 0.02Am^2 momentum is as low as

$$i = \frac{0.02}{0.5} \cdot 54 = 2.16mA \tag{2.4}$$

One of the other advantages with the torquers from ZARM is that it is equipped with digital nine pin connection, which eliminates the need for the power supply circuit, and current control hardware needed for the magnetic coils.

2.2.2 Permanent magnet

If a large permanent magnet is put in the spacecraft, this magnet will interact with the earths magnetic field in much the same way as a compass. The south pole of the magnet will be drawn towards the magnetic north pole of the earth, and vice versa. This will lead to a slight tumbling mode with two revolutions per orbit and no possibilities of controlling spin around the magnets axis. Without any other means of detumbling, this will also be very slow.

2.2.3 Spin stabilization

If the satellite rotates around one axis, the gyroscopic effect of this will reduce the fluctuations on the other axes. The spin can be obtained in various ways. If the satellite is colored differently on each other side, the solar pressure will be greater on the lighter surfaces than on the darker ones. This however is a very slow method. Spinning could also be obtained by a thruster and maintained by magnetic torquers. Instead of spinning the entire satellite, a momentum wheel inside the satellite can do the same job.

2.2.4 Gravity gradient

If a boom with a tip mass is deployed from the satellite, the innermost of the two masses will be in a lower orbit and pull on the outermost, preferably the satellite. The pull and the change in the satellites moment of inertia will stabilize the two axes perpendicular to the boom. The challenge, especially on a Cubesat, is the deployment and construction of the boom. The boom should be stiff to avoid oscillation, but must be stored in a small compartment of the satellite during launch.

2.2.5 Reaction Control System, RCS

A RCS utilizes Newtons third law which states that every action has an equal and contrary reaction. Some gas is propulsed out of a nozzle and the satellite moves in opposite direction. If the nozzles are not pointed directly away from the center of inertia this will lead to rotational torques as well. The gas is stored in a tank on board the satellite. There are mounted six thrusters in pairs to generate the momentum needed for control. The RCS is highly accurate, but is large and heavy, and will eventually have spent all the available gas, and will thus not perform for long without a very large tank, hence the weight problem.

2.2.6 Reaction wheels

The raction wheels uses the rotational variant of Newtons third law. If the action is accelerating a wheel inside the spacecraft, the spacecraft will accelerate just as much in the opposite direction. Three, or four in a tetrahedron configuration for redundancy, reaction wheels in a satellite makes up the most accurate attitude control actuator for satellites. The size of today's systems is however so large that this solution is not suited for Cubesats.

2.2.7 Actuator summary

This tables shows the obtainable accuracy, as suggested by Wertz and Larson (1999), together with some advantages/disadvantages of the different actuators.

Method	**Acc. [deg]**	**Pros**	**Cons**
Spin Stabilization	0.1-1.0	Passive, simple Cheap	Inertially oriented
Gravity gradient	1-5	Passive, simple Cheap	Central body oriented
RCS	0.01-1	quick response	Consumables
Magnetic torquers	1-2	Cheap	Slow, lightweight, LEO only
Reaction Wheels	0.001-1	Expensive, precise, faster slew	Weight

The conclusion that has already been drawn in the NCUBE project is to use magnetic torquers in conjunction with a gravity gradient. A choice well argued for in the previous text. The choice between magnetic coils and magnetic rods, however, is not that obvious, and depends on whether it is the mass or power budget that is tightest. For simplicity, the magnetic rods with their digital interface would have been the best choice, but as the price for a single rod could be no lower than 3000€, it was not feasible economically for the NCUBE student satellite. Instead, locally produced coils will be used. The implementational issues concerning the magnetic torque coils are discussed in Chapter 4.4

Chapter 3

Definitions and notation

In order to describe the orientation of the satellite, the mathematics behind sensor modelling, and the Kalman filtering, some notational defining is required. Both positions and orientations are expressed through vectors and matrices. A vector or matrix needs a reference to be unambiguous. The superscript on a vector or a matrix indicates in which of the reference frames described in Chapter 3.1 the vector or matrix is represented, hence $\boldsymbol{\omega}_{IB}^{I}$ is the angular velocity of the body frame relative the ECI frame given in the ECI frame, $\boldsymbol{\omega}_{IB}^{B}$ is the same vector in body frame.

3.1 Reference frames

This section describes the definitions of the different reference frames used throughout this document. It is necessary to have these different reference frames as different measurements and modelling are done in different frames and the rotation from one to another must be unique and well defined.

Earth-Centered Inertial (ECI) Reference Frame For Newtons laws to be valid, one must have a non accelerating frame. The ECI is such an non-accelerating inertial frame. The frame is located in the center of the earth and fixed towards the stars. This reference frame will be denoted I, and the earth rotates around its z-axis. The x-axis points towards the vernal equinox, and the y-axis completes a right hand Cartesian coordinate system.

Earth-Centered Earth Fixed (ECEF) Reference Frame The frame shares it's origin and z-axis with the ECI frame and is denoted E. The x-axis intersects the earths surface at latitude $0°$ and longitude $0°$. The y-axis completes the right hand system. The ECEF rotates with the earth with a constant angular velocity ω_E, and is therefore not an inertial reference frame, and hence the laws of Newton is not valid.

North East Down (NED) Reference Frame The NED frame has its origin at the surface of the earth and is denoted N. The x-axis points northwards in the tangent plane of the earth, and the y-axis points eastwards. The z-axis points downwards perpendicular to the tangent plane to complete the right hand system.

Orbit Reference Frame The orbit frame has its origin at the point which the spacecraft has its center. The origin rotate at an angular velocity ω_O relative to the ECI frame and has its z-axis pointed towards the center of the earth. The x-axis points in the spacecraft's direction of motion tangentially to the orbit. It is important to note that the tangent is only perpendicular to the radius vector in the case of a circular orbit. In elliptic orbits, the x-axis does not align with the satellite's velocity vector. This is illustrated in Figure 3.1 where the orbit frame and it's relation to the earth centered orbit frame is shown. The y-axis completes the right hand system as usual. The satellite attitude is described by roll, pitch and yaw which is the rotation around the x-, y-, and z-axis respectively. The orbit reference frame is denoted O.

Earth centered orbit reference frame This is the frame which the keplerian elements of Chapter 4.2.1 defines. The frame is centered at the earth's center, with x-axis towards perigee, y-axis along the semiminor-axis, and z-axis

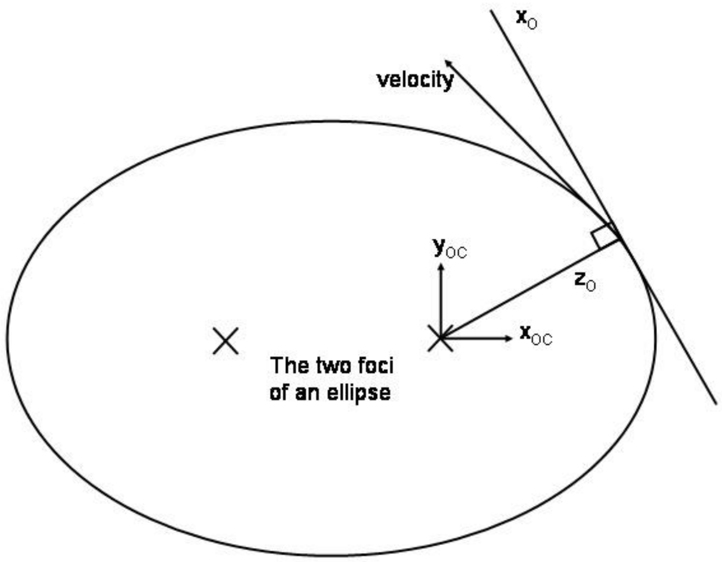


Figure 3.1: The difference between the velocity and a normal to a vector from a focus

perpendicular to the orbital plane to complete the right hand system. The earth centered orbit frame is denoted OC.

Body Reference Frame The body frame shares it's origin with the orbit frame and is denoted B. The rotation between the orbit frame and the body frame is used to represent the spacecraft's attitude. It's axes are locally defined in the spacecraft, with the origin in the center of gravity or the center of the volume. The nadir side of the spacecraft, intended to point towards the earth, is in the z-axis direction and similar with the other two sides coinciding with the orbit frame.

3.2 Rotation matrices

The rotation matrix is a description of the rotational relationship between two reference frames. The rotation matrix can therefore be used to transform a vector from one reference frame to another. The rotation matrix $\mathbf{R}_B^I$ rotates the vector $\boldsymbol{\omega}_{IB}^B$ from the body frame to the ECI frame such that $\boldsymbol{\omega}_{IB}^I = \mathbf{R}_B^I \boldsymbol{\omega}_{IB}^B$. For the rotation matrices to be unambiguous $\boldsymbol{\omega}_{ab}^a = \mathbf{R}_b^a \mathbf{R}_a^b \boldsymbol{\omega}_{ab}^a$ must be true, hence

$$\mathbf{R}_b^a \mathbf{R}_a^b = \mathbf{I} \tag{3.1}$$

where $\mathbf{I}$ is the identity matrix. This leads to $\mathbf{R}_b^a = \left(\mathbf{R}_a^b\right)^{-1}$, and it can also be shown that $\mathbf{R}_b^a = \left(\mathbf{R}_a^b\right)^T$ and that the determinant $det\mathbf{R}_b^a = 1$,Egeland (2001). Rotation matrices belongs to the set of matrices denoted $SO(3)$, defined by

$$SO\left(3\right) = \left\{\mathbf{R}|\mathbf{R} \in R^{3\times3}, \quad \mathbf{R}^T\mathbf{R} = \mathbf{I} \quad \text{and} \quad \det \mathbf{R} = \mathbf{1}\right\} \tag{3.2}$$

3.2.1 Angular velocity

The rate at which a rotation matrix changes is called angular velocity, $\boldsymbol{\omega}_{ab}^a$. To establish the angular velocity, it's relationship with the rotation matrix, and the time derivative of the rotation matrix consider the following. First equation (3.1) is derivated yielding

$$\frac{\delta}{\delta t}\left(\mathbf{R}_b^a \mathbf{R}_a^b\right) = \dot{\mathbf{R}}_b^a \mathbf{R}_a^b + \mathbf{R}_b^a \dot{\mathbf{R}}_a^b = \mathbf{0} \tag{3.3}$$

The matrix $\mathbf{S}$ is then defined by

$$\mathbf{S} = \dot{\mathbf{R}}_b^a \mathbf{R}_a^b \tag{3.4}$$

$\mathbf{S}$ is a skew symmetric matrix as $\mathbf{S}^T = -\mathbf{S}$, an all skew symmetric matrices can be written as

$$\mathbf{S} = \begin{bmatrix} 0 & -\boldsymbol{\omega}_3 & \boldsymbol{\omega}_2 \\ \boldsymbol{\omega}_3 & 0 & -\boldsymbol{\omega}_1 \\ -\boldsymbol{\omega}_2 & \boldsymbol{\omega}_1 & 0 \end{bmatrix} \tag{3.5}$$

which is the same as the skew symmetric form of the vector $\boldsymbol{\omega}_{ab}^a$

$$\mathbf{S}\left(\boldsymbol{\omega}_{ab}^a\right) = \begin{bmatrix} 0 & -\boldsymbol{\omega}_3 & \boldsymbol{\omega}_2 \\ \boldsymbol{\omega}_3 & 0 & -\boldsymbol{\omega}_1 \\ -\boldsymbol{\omega}_2 & \boldsymbol{\omega}_1 & 0 \end{bmatrix}, \qquad \boldsymbol{\omega}_{ab}^a = \begin{bmatrix} \boldsymbol{\omega}_1 \\ \boldsymbol{\omega}_2 \\ \boldsymbol{\omega}_3 \end{bmatrix} \tag{3.6}$$

Thus writing

$$\mathbf{S}\left(\boldsymbol{\omega}_{ab}^a\right) = \dot{\mathbf{R}}_b^a \left(\mathbf{R}_b^a\right)^T \tag{3.7}$$

which with post-multiplication with $\mathbf{R}_b^a$ gives

$$\dot{\mathbf{R}}_b^a = \mathbf{S}\left(\boldsymbol{\omega}_{ab}^a\right)\mathbf{R}_b^a = \mathbf{R}_b^a \mathbf{S}\left(\boldsymbol{\omega}_{ab}^b\right) \tag{3.8}$$

3.3 Attitude representations

Representation of attitude is not as straight forward as position representation. There are three rotational degrees of freedom, hence the attitude can be represented by three parameters. However, a three parameter representation is singular for some attitude. To avoid singularity more parameters is needed, but then there is redundancy in the representation, and it must be subjected to constraints.

3.3.1 Euler angles

The euler angles $\begin{bmatrix} \psi & \theta & \phi \end{bmatrix}$ are the rotation around the x-, y-, and z-axis called roll, pitch and yaw or azimuth respectively. The euler angles are the base for the rotation matrix. The rotation matrix can be decomposed into three rotations about three orthogonal axes.

$$\mathbf{R}_{x,y,z}\left(\psi,\theta,\phi\right) = \mathbf{R}_x\left(\psi\right)\mathbf{R}_y\left(\theta\right)\mathbf{R}_z\left(\phi\right) \tag{3.9}$$

where

$$\mathbf{R}_x\left(\psi\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} \tag{3.10}$$

$$\mathbf{R}_y\left(\theta\right) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{3.11}$$

$$\mathbf{R}_z\left(\phi\right) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.12}$$

which yields

$$\mathbf{R}_{x,y,z}\left(\psi,\theta,\phi\right) = \begin{bmatrix} c\theta c\psi & s\theta s\phi c\psi - c\phi s\psi & s\theta c\phi s\psi + s\phi s\psi \\ c\theta s\psi & s\theta s\phi s\psi + c\phi c\psi & s\theta c\phi s\psi - s\phi c\psi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{3.13}$$

where $\cos = c$ and $\sin = s$ is used for notational shortening. As an attitude representation the rotation matrix has nine parameters and hence six redundant. It is also singular for $\theta = \pm 90°$. When used in numerical analysis it is important to maintain orthogonality, which can be quite difficult.

3.3.2 Euler parameters

The euler parameters are an attitude parameterization, also referred to quaternions as they are essentially the same. The quaternion is the attitude parameterization preferred in most computational aspects. It has four parameters, no singularities, and the constraint is easy to uphold. A rotation of an angle θ around an axis $\boldsymbol{\lambda}$ gives the rotation vector

$$a_\theta = \theta\boldsymbol{\lambda} \tag{3.14}$$

and the quaternion

$$\mathbf{q} = \begin{bmatrix} \eta \\ \boldsymbol{\epsilon} \end{bmatrix} = \begin{bmatrix} \cos\frac{\theta}{2} \\ \boldsymbol{\lambda}\sin\frac{\theta}{2} \end{bmatrix} \tag{3.15}$$

where the constraint required is described by one of the following:

$$\begin{aligned} \|\mathbf{q}\| &= 1 && (3.16) \\ \boldsymbol{\epsilon}^T\boldsymbol{\epsilon} + \eta^2 &= 1 && (3.17) \\ \epsilon_1^2 + \epsilon_2^2 + \epsilon_3^2 + \eta^2 &= 1 && (3.18) \end{aligned}$$

The main advantage of the quaternion is that rotations are expressed by the quaternion product. If the rotation from $\mathbf{u}$ to $\mathbf{v}$ is described by $\mathbf{q}_1 = \begin{bmatrix} \eta_1 \\ \boldsymbol{\epsilon}_1 \end{bmatrix}$

$$\begin{bmatrix} 0 \\ \mathbf{u} \end{bmatrix} = \mathbf{q}_1 \otimes \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix} \otimes \overline{\mathbf{q}_1} \tag{3.19}$$

where $\overline{\mathbf{q}_1} = \begin{bmatrix} \eta_1 \\ -\boldsymbol{\epsilon}_1 \end{bmatrix}$ and the quaternion product is defined as

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = \begin{bmatrix} \eta_1 \\ \boldsymbol{\epsilon}_1 \end{bmatrix} \otimes \begin{bmatrix} \eta_2 \\ \boldsymbol{\epsilon}_2 \end{bmatrix} = \begin{bmatrix} \eta_1\boldsymbol{\epsilon}_2 + \eta_2\boldsymbol{\epsilon}_1 + \mathbf{S}(\boldsymbol{\epsilon}_1)\,\boldsymbol{\epsilon}_2 \\ \eta_1\eta_2 - \boldsymbol{\epsilon}_1^T\boldsymbol{\epsilon}_2 \end{bmatrix} \tag{3.20}$$

Also when $\mathbf{q}_1$ represents an attitude, and $\mathbf{q}_2$ a rotation, the new attitude is given by the quaternion product.

$$\mathbf{q} = \mathbf{q}_2 \otimes \mathbf{q}_1 \tag{3.21}$$

Differentiating the quaternions yields according to Egeland (2001)

$$\begin{aligned} \dot{\eta} &= -\frac{1}{2}\boldsymbol{\varepsilon}^T\boldsymbol{\omega} && (3.22) \\ \dot{\boldsymbol{\varepsilon}} &= \frac{1}{2}\left[\eta\mathbf{I} + \mathbf{S}(\boldsymbol{\varepsilon})\right]\boldsymbol{\omega} && (3.23) \end{aligned}$$

where $\boldsymbol{\omega}$ is the angular velocity of the body frame in relation to the frame as attitude is given, given in body frame.

Chapter 4

Mathematical modeling

The chapter presents the mathematical modelling of the coarse sun sensor, the satellite orbit, the earts magnetic field, the torque coils, and the satellite with it's environment.

4.1 Coarse sun sensor

The coarse sun sensor and the mathematical modelling needed to produce both the measured sun vector, and a vector for comparison is presented.

4.1.1 Sun vector in body frame

The University of Oslo together with the Institute for Energy Technology has offered to make all the solar cells needed. These cells are single junction silicone cells with an efficiency of 18%. The big advantage with custom made cells is that the surface of the satellite can be completely covered. Commercial cells are made in standard sizes and would have left more unused area. The current from a solar cell is dependent the area of the solar cell exposed to sunlight. This area is dependent on the angle between the solar panel and the sun through a cosine law. This leads to the assumption that the currents I are dependent on the suns angle of attack α_s on the solar panel as

$$I = I_{\max} \sin \alpha_s \tag{4.1}$$

where $I_{\max}$ is the current induced in the solar panels when the sun shines directly at it, $\alpha_s = \frac{\pi}{2}$. To verify this, a solar panel was set up in a dark room, and the current flow for different incoming light angles was measured. As seen in Figure

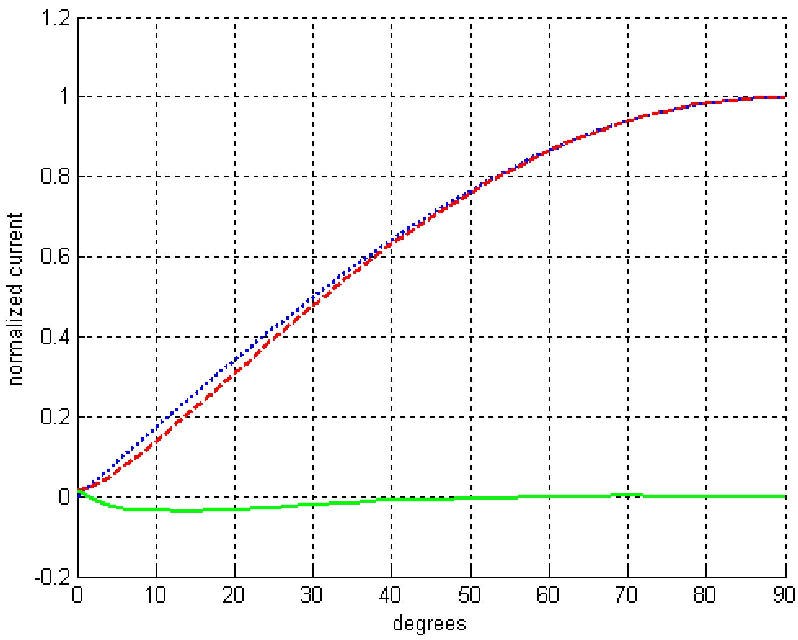


Figure 4.1: Normalized current measurement and a sinus for comparison.

4.1, where the dashed red line is the normalized current measurement, the dotted blue is the sinus and the green line is the difference between the two others, the current is very near the assumption of equation (4.1). The non zero current for a zero degree angle of attack is due to the light not coming as parallel light waves or from a single point, but a light bulb half visible even for the zero degree angle of attack. The test setup with exaggerated dimensions to illustrate this issue is

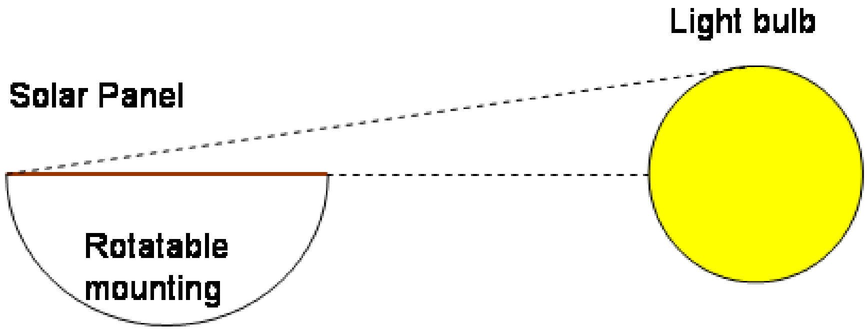


Figure 4.2: Solar panel test setup

shown in Figure 4.2. In the Master Thesis of Appel (2003) very similar results are presented for angular dependencies of solar cells. Combining these results from all five solar panels makes it possible to determine the sun vector in the body frame. For the planar case, the attitude is computed from the two highest

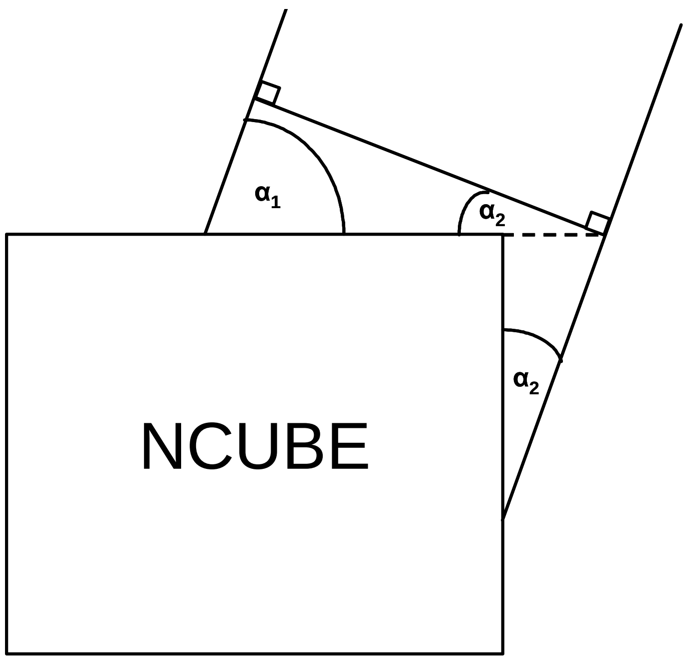


Figure 4.3: The NCUBE and incoming sunbeam

current measurements as shown in Figure 4.3. If the two sides of an angle is in pair perpendicular to the two sides of an other angle, the two angles are equal, hence α_2 is found as shown in Figure 4.3. Since the sum of all angles in a triangle equals π,

$$\alpha_1 + \alpha_2 = \frac{\pi}{2} \tag{4.2}$$

which through trigonometric relationships yields

$$\sin \alpha_2 = \cos \alpha_1 \tag{4.3}$$

Equation (4.1) applied to two sides of the satellite results in

$$I_1 = I_{\max} \sin \alpha_1 \tag{4.4}$$
$$I_2 = I_{\max} \sin \alpha_2 \tag{4.5}$$

Dividing equation (4.4) with equation (4.5) yields

$$\frac{I_1}{I_2} = \frac{\sin \alpha_1}{\sin \alpha_2} \tag{4.6}$$
$$\frac{I_1}{I_2} = \frac{\sin \alpha_1}{\cos \alpha_1} = \tan \alpha_1 \tag{4.7}$$

hence

$$\alpha_1 = \arctan \frac{I_1}{I_2} \tag{4.8}$$

In the three dimensional case it is not possible to find the local sun vector if the sun is shining on the bottom of the satellite where there is no solar panel. But when the sun is shining on the top of the satellite, this is recognized when the solar panel delivers power, the same computation as for the planar case can be used again. When the sunbeam is not in the plane of Figure 4.3, but at an angle of attack on the top panel, α_3, as indicated in Figure 4.4, equations (4.4) and (4.5) becomes

$$I_1 = I_{\max} \cos \alpha_3 \sin \alpha_1 \tag{4.9}$$
$$I_2 = I_{\max} \cos \alpha_3 \sin \alpha_2 \tag{4.10}$$

with $I_{\max} \cos \alpha_3$ substituted for $I_{\max}$. The current induced in the top panel becomes

$$I_3 = I_{\max} \sin \alpha_3 \tag{4.11}$$

Dividing equation (4.11) with equation (4.9) yields

$$\frac{I_3}{I_1} = \frac{I_{\max} \sin \alpha_3}{I_{\max} \sin \alpha_1 \cos \alpha_3} \tag{4.12}$$
$$\frac{I_3}{I_1} = \frac{\tan \alpha_3}{\sin \alpha_1} \tag{4.13}$$
$$\tan \alpha_3 = \frac{I_3}{I_1} \sin(\alpha_1) \tag{4.14}$$

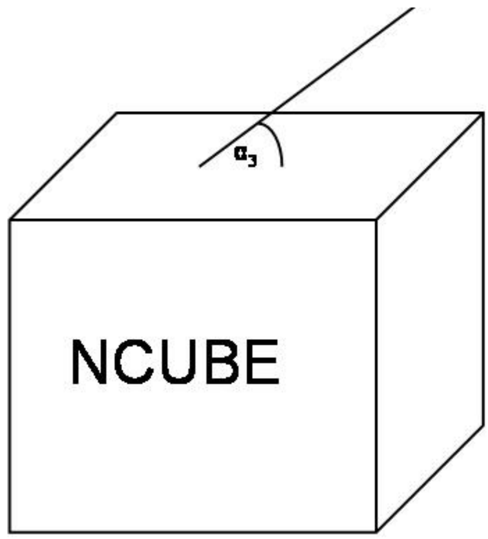


Figure 4.4: Sunbeam 3D angle of attack

hence

$$\alpha_3 = \arctan\left(\frac{I_3}{I_1}\sin(\alpha_1)\right) \tag{4.15}$$

Combining these results under the assumption that the sun shines on positive x and y-axis, and negative z-axis, gives the sun vector in the body frame as

$$\mathbf{v}_S^B = \begin{bmatrix} \sin\alpha_1\cos\alpha_3 \\ \sin\alpha_2\cos\alpha_3 \\ -\sin\alpha_3 \end{bmatrix} \tag{4.16}$$

which when compared to equations (4.9), (4.10) and (4.11) is seen to be

$$\mathbf{v}_S^B = \begin{bmatrix} I_1 \\ I_2 \\ -I_3 \end{bmatrix}\frac{1}{I_{\text{max}}} \tag{4.17}$$

As I_{max} is a scalar, and it is only the direction of the sun vector that gives attitude information, I_{max} can be removed from the equation, and the measured sun vector in body frame is reduced to

$$\mathbf{v}_S^B = \begin{bmatrix} I_1 \\ I_2 \\ -I_3 \end{bmatrix} \tag{4.18}$$

or

$$\mathbf{v}_S^B = \begin{bmatrix} X & 0 & 0 \\ 0 & Y & 0 \\ 0 & 0 & Z \end{bmatrix}\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} \tag{4.19}$$

where X, Y and Z are ± 1 depending on whether the solar cells on the positive or negative side of the satellite is delivering current, and hence points towards the sun. In the computations $I_{\max}$ is eliminated, and this is crucial as $I_{\max}$ depends directly on the load resistance, which is highly variable, and depends on which subsystems are being used, and on whether the batteries are being recharged or not.

4.1.2 Sun vector in orbit frame

To utilize the measured body frame sun vector, the sun vector in orbit frame must be known in such a way that the rotation between the two could be estimated. For the computation of the sun vector, the earths orbit is assumed to be circular with an orbit time of 365 days, and the satellite is regarded as being positioned in the center of the earth. This leads to a maximum error of $e = \arctan \frac{R_o}{R_e} \approx 4.65 \cdot 10^{-5}$ $[rad]$, where R_o is the radius of the satellite orbit and R_e is the earth orbit radius. This error is negligible compared to the accuracies achieved with this crude sun sensor. The elevation, ϵ_s, as seen on Figure 4.5 will, due to the earth's rotation

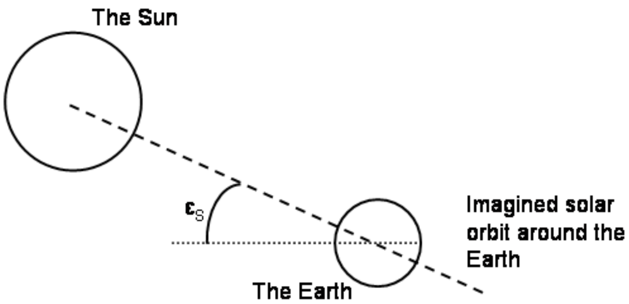


Figure 4.5: The suns imagined orbit around the earth.

axis not being perpendicular to it's orbital plane, be between $\pm 23°$ with a period of 365 days approximately, and is zero at the first day of spring and fall. This leads to the following simplified equations as suggested by Kristiansen (2000)

$$\epsilon_s = \frac{23\pi}{180} sin \frac{T_s \cdot 2\pi}{365} \tag{4.20}$$

$$\lambda_s = \frac{T_s \cdot 2\pi}{365} \tag{4.21}$$

where λ_s is the azimuth angle towards the sun with zero towards the vernal equinox, and T_s the time in days since the earth passed the vernal equinox.The sun vector on the day the earth passes the vernal equinox expressed in the ECI frame will according to the definition of the ECI be

$$\mathbf{v}_{S0}^{I} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{4.22}$$

Rotation of λ_s about the z_I axis, ε_s about the y_I axis gives the sun vector in ECI frame as

$$\mathbf{v}_{S}^{I} = \mathbf{R}_y(\varepsilon_s)\,\mathbf{R}_z(\lambda_s)\,\mathbf{v}_{S0}^{I} \tag{4.23}$$

$$\mathbf{v}_{S}^{I} = \begin{bmatrix} \cos\lambda_s \cos\epsilon_s \\ \sin\lambda_s \\ \cos\lambda_s \sin\epsilon_s \end{bmatrix} \tag{4.24}$$

And again as it is only the direction of the vector giving attitude information, $\cos\lambda_s$ is set outside the vector and removed, giving

$$\mathbf{v}_{S}^{I} = \begin{bmatrix} \cos\epsilon_s \\ \tan\lambda_s \\ \sin\epsilon_s \end{bmatrix} \tag{4.25}$$

For comparison with the measured body frame sun vector from the solar panels, the sun vector is transformed to orbit frame by

$$\mathbf{v}_{S}^{O} = \mathbf{R}_{I}^{O}\mathbf{v}_{S}^{I} \tag{4.26}$$

Where $\mathbf{R}_{I}^{O}$ is discussed in Chapter 4.3. When the sun vector is determined, it is combined with the magnetometer measurement in the Kalman filter, as described in Chapter 5, to calculate the optimal estimate based on available measurements.

4.1.3 The earth albedo error

The accuracy of the coarse sun sensor is deteriorated heavily by the reflection of the suns energy from the earth. Only half the earth is illuminated by the sun, and only parts, if any, of this half is visible from the satellite. The albedo is also very different from place to place on the earth. For instance will the polar areas and sandy areas such as deserts reflect much more light than ocean and forest areas. A model for predicting the influence of the energy reflected from earth on a coarse sun sensor is currently being made by Appel (2003). The early results suggests that the magnitude of the deterioration will be in the order of 16°. For a more exact utilizing of the coarse sun sensor the finished results of Appel (2003) should be investigated, and the implementation of the albedo model should be considered as part of future work.

4.2 The satellite orbit

The satellite's orbit and it's position in orbit is needed to determine the rotational relationship between the ECEF frame, in which the magnetic vector is given, the ECI frame, in which the sun vector is given, and the orbit frame in which measurements are taken. The position in ECEF coordinates is also needed in the computation of the magnetic reference vector. To obtain this, an orbit estimator will be implemented. The NCUBE satellite will travel approximately 700 km above the earths surface in a near circular orbit. The orbit period will then be approximately 98.8 minutes. The inclination, as described in Chapter 4.2.1, will be 98 degrees, which gives a near polar orbit. Figure 4.6 shows the earth and a

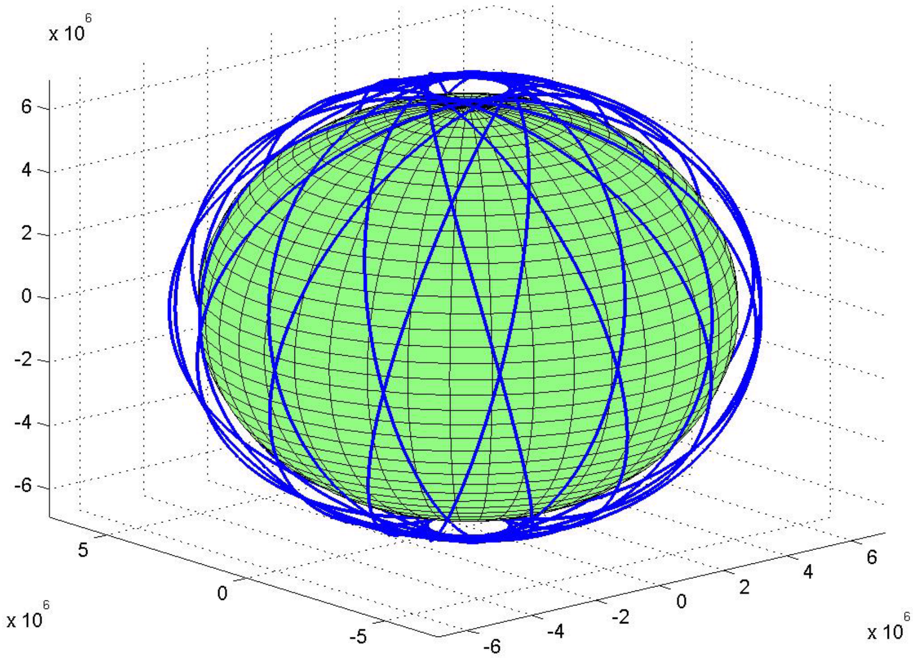


Figure 4.6: The earth with satellite track

satellite track for 15 orbits, which is what the NCUBE satellite manages in just over 24 hours.

4.2.1 Satellite orbits and keplerian elements

The physical laws describing the motion of planets and satellites was first described by Johann Kepler [1571-1630] and is mathematically derived from Newton's equations of motion for instance in Forssell (1991). Kepler's three laws state that:

1. The orbit of each planet is an ellipse, with the Sun at one of the foci.

2. The line joining the planet to the Sun sweeps out equal areas in equal times.

3. The square of the period of a planet is proportional to the cube of its mean distance from the Sun.

These laws are also applicable to satellites orbiting the earth, with one modification. The orbit is not limited to an ellipse, but can be any conical section. The laws of Kepler are valid under the assumption that the satellite and the earth are point masses, and that gravitational forces are the only forces acting on the two bodies. It is also assumed that the two masses are not under influence of gravitational forces from other celestial bodies than each other. The satellite's orbit will, as it is not leaving the earth, be an ellipse with the center of the earth at one of the foci.

The mathematical background for Kepler's laws comes from Newton's laws. To establish the dynamics of a satellite's orbit position, it is useful to look into Newtons gravitational studies. The law of universal gravitation states that the force, $\mathbf{F}$, acting on the satellite with mass, m, due to the earth's mass, M_e, is

$$\mathbf{F} = -\frac{\mathbf{G}M_e m}{r^2}\frac{\mathbf{r}}{r} \tag{4.27}$$

where $\mathbf{r}$ is the vector from the center of the earth to the satellite, and $\mathbf{G}$ is the universal gravitational constant. With no other forces than gravity, Newton's second law

$$\sum \mathbf{F} = m\ddot{\mathbf{r}} \tag{4.28}$$

gives

$$m\ddot{\mathbf{r}} = -\frac{\mathbf{G}M_e m}{r^2}\frac{\mathbf{r}}{r} \tag{4.29}$$

hence

$$\ddot{\mathbf{r}} = -\frac{\mathbf{G}M_e}{r^2}\frac{\mathbf{r}}{r} \tag{4.30}$$

This is not the dynamics of the distance between the two, but the dynamics of the satellite in a reference frame not moving with the earth. The earth's movement due to the satellite's mass described in the same reference frame is

$$\ddot{\mathbf{r}} = -\frac{\mathbf{G}m}{r^2}\frac{\mathbf{r}}{r} \tag{4.31}$$

hence the dynamics of the distance between the two is

$$\ddot{\mathbf{r}} = -\frac{\mathbf{G}\left(M_e + m\right)}{r^2}\frac{\mathbf{r}}{r} \tag{4.32}$$

but as m compared to M_e is totally negligible, the approximation in equation (4.30) is used instead. This means that satellite orbit dynamics is independent of the satellite's mass, which is convenient as the estimate is valid for any satellite. Building an estimator on the basis of equation (4.30) is possible, but impractical as it doesn't incorporate known regularities of an orbit. It is also hard to update with accurate information from ground observations.

Keplerian elements

Kepler's laws are the foundation for the Keplerian elements which are much better suited for use in estimating a satellite's orbit and position. The comprehension of these elements is crucial to the understanding of the orbit estimator and is therefore presented here. The Keplerian elements, sometimes called orbital elements, define an ellipse, orient it about the earth, and place the satellite on the ellipse at a particular time. In the Keplerian model, satellites orbit in an ellipse of constant shape and orientation. The Earth is at one focus of the ellipse, not the center, unless the orbit ellipse is a perfect circle and the two foci coincide with the center. There are six Keplerian elements, and they are:

1. Orbital Inclination
2. Right Ascension of Ascending Node (R.A.A.N.)
3. Argument of Perigee
4. Eccentricity
5. Mean Motion
6. Mean Anomaly

As these elements describe a satellites position at a specific time, this time must also be given. The most widely used format is called epoch, and gives the year and day of the year as a decimal number. Based on this time, the ascension of the zero meridian, θ, can also be calculated. The ascension of the zero meridian is needed in the rotation between ECI and ECEF reference frames given by

$$\mathbf{R}_E^I = \mathbf{R}_z(\theta) \tag{4.33}$$

Orbital Inclination Denoted i. The orbit ellipse lies in a plane known as the orbital plane. The orbital plane always goes through the center of the earth, but may be tilted any angle relative to the equator. Inclination is the angle between the orbital plane and the equatorial plane. By convention, inclination is a number between 0 and 180 degrees. Orbits with inclination near 0 degrees are called equatorial orbits, and orbits with inclination near 90 degrees are called polar. The intersection of the equatorial plane and the orbital plane is a line which is called the line of nodes. The line of nodes is more thoroughly described below. See Figure 4.7 and 4.8 for visual description of all the keplerian elements.

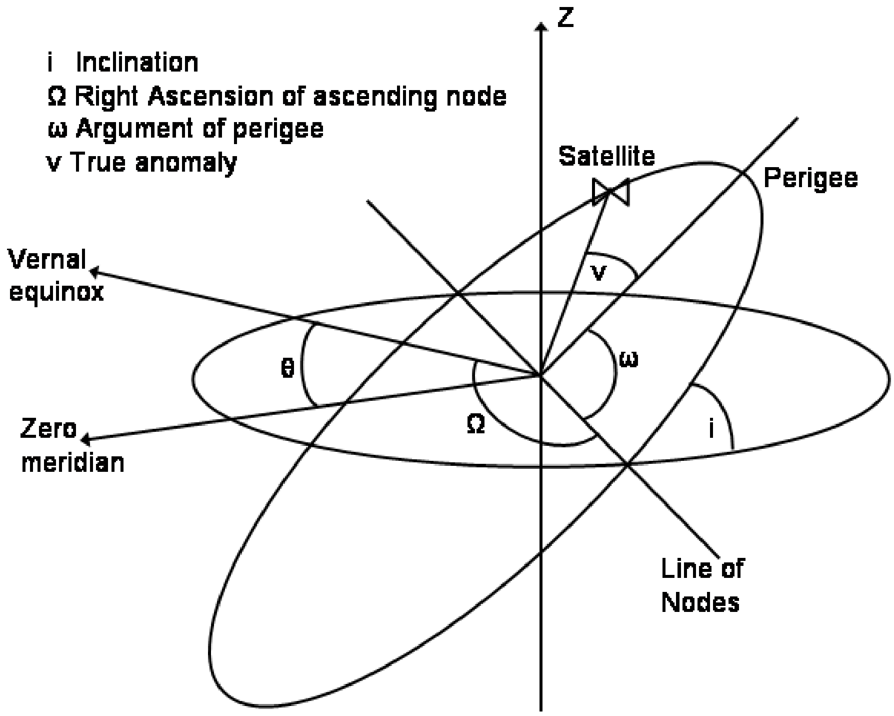


Figure 4.7: The keplerian elements

Right Ascension of Ascending Node Denoted Ω. The line of nodes intersect the equatorial plane two places; in one of them the satellite passes from south to north, this is called the ascending node. The other node where the satellite passes from north to south is called the descending node. The right ascension of ascending node is an angle, measured at the center of the earth, from the vernal equinox to the ascending node. By convention, the right ascension of ascending node is a number between 0 and 360 degrees. Together with the inclination, the right ascension of ascending node defines the orbital plane in which the satellites elliptic orbit lies.

Argument of Perigee Denoted ω. The perigee is the point in the ellipse closest to the focus point in which the earth lies. The point in the ellipse farthest from the earth is called the apogee. The angle between the line from perigee through the center of the earth to the apogee, and the line of nodes is the argument of perigee. To clarify the ambiguity caused by the multiple angles where to lines intersect, the argument of the perigee is defined as the angle from the ascending node to the perigee. By convention, the argument of perigee is a number between 0 and 360 degrees.

Eccentricity Denoted e. Given the semimajor-axis, a, as half the distance between the apogee and the perigee, and the semiminor-axis, b, as half the length between the edges perpendicular to a, the eccentricity is given as

$$e = \sqrt{1 - \frac{b^2}{a^2}} \tag{4.34}$$

For an ellipse e lies between 0 and 1. For a perfect circle $a = b$ and thus $e = 0$.

Mean Motion Denoted n. The first four Keplerian elements specifies the orientation of the orbital plane, the orientation of the orbit ellipse in the orbital plane, and the shape of the orbit ellipse. The mean motion is the average angular velocity, or number of revolutions per days, and describes the size of the ellipse. The mean motion in rad/sec, and the semimajor-axis is related through Kepler's third law by

$$n^2 a^3 = \mu_e \tag{4.35}$$

where $\mu_e = GM_e$ is the earth's gravitational constant. Because of this relationship, the keplerian element mean motion is sometimes replaced by the semimajor-axis.

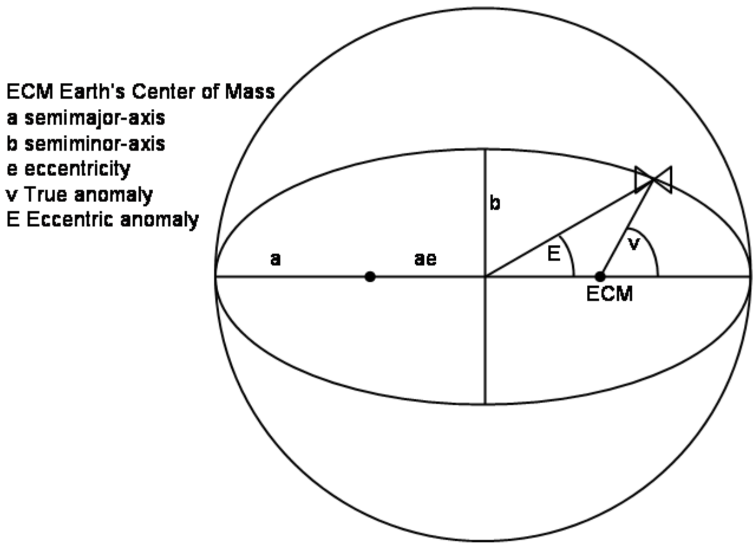


Figure 4.8: The keplerian elements in plane

Mean Anomaly Denoted M. The last keplerian element defines where in the ellipse the satellite is positioned. Mean anomaly is an angle that marches uniformly in time from 0 to 360 degrees during one revolution. It is defined to be 0 degrees at perigee, and therefore is 180 degrees at apogee. It is important to note that in a non-circular ellipse, this angle does not give the direction towards the satellite except at perigee and apogee. This is because satellite doesn't have a constant angular velocity. The different anomalies used is shown in Figure 4.8, where true anomaly, ν, is the direction from the earth center towards the satellite, and eccentric anomaly is the direction from the center of the ellipse towards the point on a circle, centered at the same place as the orbital ellipse with a radius equal the semimajor axis, where a line, perpendicular to the semimajor axis through the satellites position, crosses the circle. The relationship between true anomaly and eccentric anomaly is

$$\cos\nu = \frac{\cos E - e}{1 - e\cos E} \tag{4.36}$$

$$\sin\nu = \frac{\sqrt{1-e^2}\sin E}{1 - e\cos E} \tag{4.37}$$

And the relationship between mean anomaly and eccentric anomaly is

$$M = E - e\sin E\,(t) \tag{4.38}$$

The difference between eccentric and mean anomaly for eccentricities between 0 and 0.1 is shown on the left hand side of Figure 4.9, while the right hand side

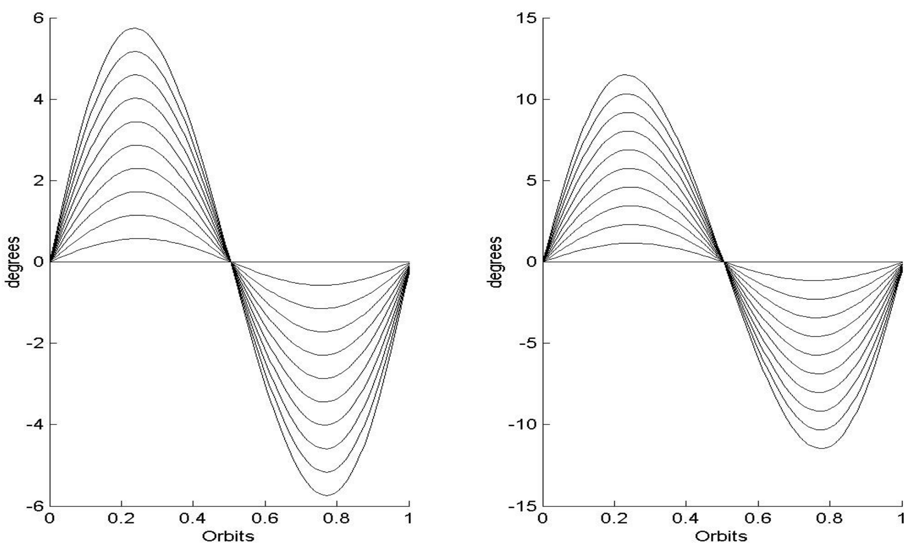


Figure 4.9: The difference between eccentric and mean anomaly (left), and true and mean anomaly (right).

shows the difference between true and mean anomaly. For a circular orbit, $e = 0$, it is seen that mean, eccentric, and true anomalies coincide.

4.2.2 Simple orbit estimator

Given the keplerian elements for a single point in time, the estimation of the future position becomes relatively straight forward. The mean anomaly marches uniformly in time, and the future prediction is therefore

$$M(t_0 + t) = M(t_0) + n \cdot t \tag{4.39}$$

To transform the estimate into ECEF coordinates, the format needed for the IGRF model, one needs to solve Kepler's equation

$$E(t) = M(t) + e \sin E(t) \tag{4.40}$$

which relates the eccentric anomaly to the mean anomaly. Kepler's equation has a unique solution, but is a transcendental equation, and so cannot be inverted and solved directly for E given an arbitrary M. Simple iterative methods such as

$$E_{i+1} = M + e \sin E_i \tag{4.41}$$

with $E_0 = 0$ gives a good estimate, as does Newtons method

$$E_{i+1} = E_i + \frac{M + e \sin E_i - E_i}{1 - e \cos E_i} \tag{4.42}$$

Given the eccentric anomaly, the vector from the center of the earth to the satellite in earth centered orbit frame is

$$\mathbf{r}_{OC} = a \begin{bmatrix} \cos E - e \\ \sqrt{1-e^2} \sin E \\ 0 \end{bmatrix} \tag{4.43}$$

Transforming $\mathbf{r}$ into ECI and ECEF frame yields

$$\mathbf{r}_I = \mathbf{R}_z(-\Omega)\,\mathbf{R}_x(-i)\,\mathbf{R}_z(-\omega)\,\mathbf{r}_{OC} \tag{4.44}$$

$$\mathbf{r}_E = \mathbf{R}_z(-\Omega + \theta)\,\mathbf{R}_x(-i)\,\mathbf{R}_z(-\omega)\,\mathbf{r}_{OC} \tag{4.45}$$

where θ is the ascension of the zero meridian, or the angle from the vernal equinox to the zero meridian, and R_z and R_x are rotation matrixes as described in Chapter 3.3.1. Figure 4.10 shows a simple propagation of keplerian elements plotted with latitude against longitude. The maximum latitude is equal to the inclination, or as in the NCUBE case with inclination equal 98°

$$lat_{\max} = 180° - i = 82° \tag{4.46}$$

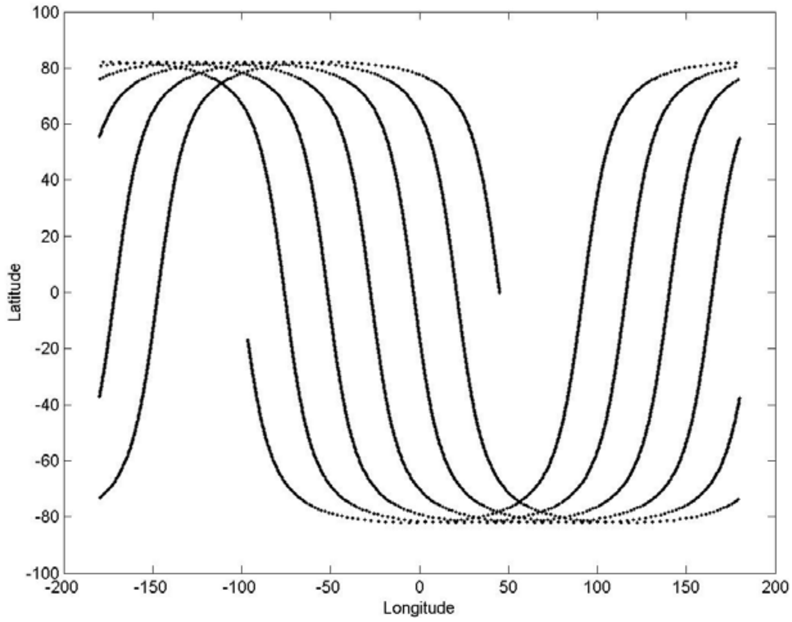


Figure 4.10: Simple orbit propagation

4.2.3 Enhanced simple orbit estimator

As the assumptions on which Kepler's Laws are based is not accurate, certain improvements utilizing known irregularities can be made. The biggest source of error is due to the earth not being perfectly circular. The deformation is often parameterized by the geopotentional function as described in Wertz and Larson (1999), which uses the deformation coefficients J_i for i'th order deformations. Gravitational forces from the sun and the moon, tidal earth and ocean, and different electromagnetic radiations, also have more or less influence on the perturbations of a satellite orbit. The perturbations are usually divided into secular, short period, and long period perturbations. The short period perturbations are periodic with a period shorter than the satellite's orbital period, while the long period perturbations have a longer period than the satellite. Figure 4.11 shows a keplerian element with the different perturbations. Only the secular perturbations are included in this enhanced simple orbit estimator, as the required position accuracy does not suggest the need for more.

Perturbations due to the nonspherical earth

The earth is not spherical, in fact it has a bulge at the equator, is flattened at the poles and is slightly pear-shaped. This leads to perturbations in all keplerian elements. The second order deformation of the earth considers the fact that it is slightly flattened, and leads to the largest perturbations in the keplerian elements.

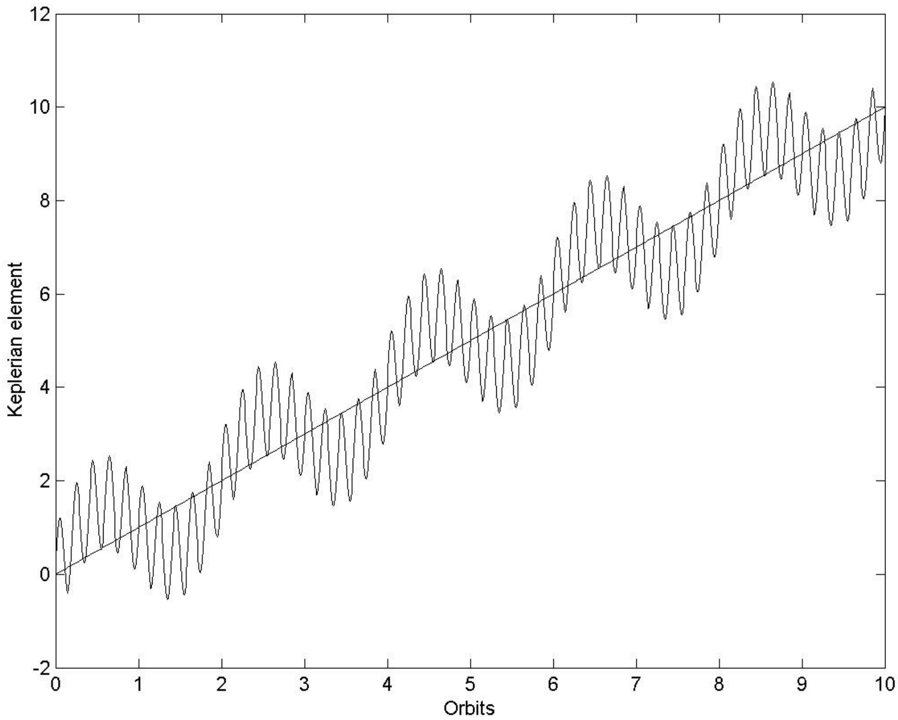


Figure 4.11: The perturbations of a keplerian element

According to the Lagrange planetary equations, the flattening factor, J_2, results in the following time derivatives of the right ascension of ascending node, and the argument of perigee

$$\dot{\Omega}_{J2} = -\frac{3}{2} n a_e^2 \frac{\cos i}{a^2 \left(1 - e^2\right)^2} J_2 \tag{4.47}$$

$$\dot{\omega}_{J2} = \frac{3}{4} n a_e^2 \frac{5 \cos^2 i - 1}{a^2 \left(1 - e^2\right)^2} J_2 \tag{4.48}$$

where a_e is the earth radius, and the numerical value of J_2 for the earth is $1.08284 \cdot 10^{-3}$. Both of these first order time derivatives are smallest for circular orbits, $e = 0$, but has their minimum for different inclinations. The inclination giving perturbations equal zero are

$$i_{\min \dot{\Omega}} = \cos^{-1}(0) = 90^\circ \tag{4.49}$$

$$i_{\min \dot{\omega}} = \cos^{-1} \sqrt{\frac{1}{5}} \approx 63.43^\circ \quad \text{or} \quad 116.57^\circ \tag{4.50}$$

This indicates that the NCUBE's high inclination at 98° is good to minimize these perturbations.

Perturbations due to the sun and the moon

The sun and the moon causes periodic variations in all keplerian elements, but secular perturbations only to the right ascension of ascending node and argument of perigee. For nearly circular orbits, an approximation suggested by Wertz and Larson (1999) yields

$$\dot{\Omega}_{moon} = -0.00338\frac{\cos i}{n} \tag{4.51}$$

$$\dot{\Omega}_{sun} = -0.00154\frac{\cos i}{n} \tag{4.52}$$

and

$$\dot{\omega}_{moon} = 0.00169\frac{5\cos^2 i - 1}{n} \tag{4.53}$$

$$\dot{\omega}_{sun} = 0.00077\frac{5\cos^2 i - 1}{n} \tag{4.54}$$

where n is the number of revolutions pr day, and $\dot{\Omega}$ and $\dot{\omega}$ are given in degrees/day. These perturbations have their minima for the same inclinations, i, as the nonspherical earth perturbations. As one could have assumed they also become larger for higher altitude orbits/ lower number of revolutions per day. This means that for the low orbit, high inclination NCUBE satellite the sun and the moon have relatively small effect .

Perturbation due to atmospheric drag

The atmospheric drag is a force creating an acceleration, a_D, in the opposite direction of the satellite's velocity. The magnitude of this acceleration depends on the density of the atmosphere, ρ, the cross section area, A, and mass, m, of the satellite and of course the magnitude of the velocity, V, and is given by

$$a_D = -\frac{1}{2}\rho\frac{C_D A}{m}V^2 \tag{4.55}$$

where C_D is the drag coefficient. The drag coefficient is further discussed in Wertz and Larson (1999), and is suggested approximated as $C_D \approx 2.2$. The atmospheric drag is a breaking force and hence removes energy from the satellite in orbit. This leads to a decrease in orbital height, but at very low rates, thus it is not included in the enhanced simple orbit estimator.

Perturbations due to solar radiation

The acceleration caused by solar radiation has a magnitude of

$$a_R = -4.5 \cdot 10^{-6}\,(1+r)\,\frac{A}{m} \tag{4.56}$$

where r is a reflection factor between 1 and 0. The perturbations due to solar radiation is in the same magnitude as atmospheric drag perturbations for altitudes at 800 km, and less for lower orbits, Wertz and Larson (1999). The solar radiation perturbations is therefor not implemented in the enhanced simple orbit estimator.

Implementation

As all included perturabations are linear first time derivatives of Keplerian elements, the position at any given future, t, time is easily predicted from initial Keplerian elements at t_0. The ECEF position needed for the magnetic reference model is given by

$$\begin{aligned}\mathbf{r}_E &= \mathbf{R}_z\left(-\left(\dot{\Omega}_0+\left(\dot{\Omega}_{J2}+\dot{\Omega}_{moon}+\dot{\Omega}_{sun}\right)t\right)+\theta_0+\omega_e\right)\cdot\mathbf{R}_x\left(-i\right)\cdot \\ &\quad \mathbf{R}_z\left(-\left(\omega_0+\left(\dot{\omega}_{J2}+\dot{\omega}_{moon}+\dot{\omega}_{sun}\right)t\right)\right)\cdot a\begin{bmatrix}\cos E-e\\ \sqrt{1-e^2}\sin E\\ 0\end{bmatrix}\end{aligned} \tag{4.57}$$

where E is from the solving of Kepler's equation (4.40). The Matlab function is shown in Appendix B.1.

4.2.4 NORAD two-line element sets

The North American Aerospace Defence Command, NORAD, keeps track of all satellites and all larger space debris. To describe these objects they use what is called the NORAD two-line element set, TLE. The set format is described in Appendix A. The two-line elements contains the keplerian elements and is used in several orbit estimators. The TLE also contains a variable called BSTAR, or B*, which is a drag coefficient. In aerodynamic theory, every object has a ballistic coefficient, B, that is the product of its coefficient of drag, and its cross-sectional area divided by its mass.

$$B=\frac{C_D\cdot A}{m} \tag{4.58}$$

The ballistic coefficient represents how susceptible an object is to drag, the higher the number, the more susceptible. The coefficient is found in equation (4.55). The first and second derivative of the mean motion, n, is also given by the TLE. The derivatives of the mean motion indicates the change in semimajor axis, and is due to energy dissipation in the satellite orbit.

4.2.5 The Simplified General Perturbations version 4

The SGP4, or Simplified General Perturbations version 4, model is based on the SGP model. They are both described in Spacetrack Report No.3 by Hoots and

Roehrich (1980). The purpose of this model is to provide means of propagating TLE sets in time to obtain a position, and velocity of a space object. The SGP4 utilizes the way in which the TLE was constructed. This means that the periodic variation, and the way that they were removed, is taken into consideration when the orbit is reconstructed and propagated. The SGP4 model reconstructs all periodic perturbations, both the short period ones and the long period ones. This might not be necessary as the required accuracy of the ADCS is not that high.

4.2.6 The Choice

Which estimator to use depends on the accuracy needed, and the frequency of Keplerian element update. A given accuracy can be met by the simple estimator

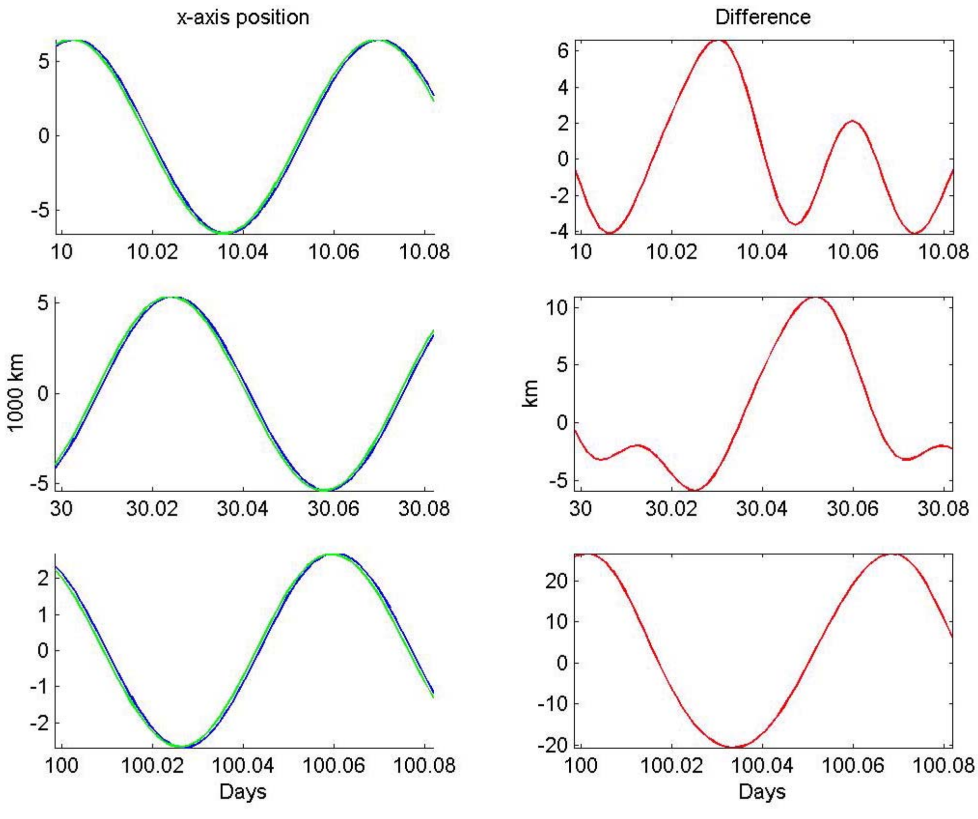


Figure 4.12: Estimates made by the SGP4 and the enhanced simple orbit estimator.

for a certain amount of time, then the estimate becomes poorer and poorer until useless. The SGP4 will be able to retain accuracy for a much longer period. The enhanced simple estimator will of course give better results than the simple estimator, but not as good as the SGP4. To use SGP4 however, the TLE is needed. One can not update the model before NORAD has released the TLE. Both the two simple estimators can be updated with Keplerian elements obtained from any source, including the TLE, and is therefore more flexible. The left hand side of Figure 4.12 shows the x-components of the satellite's ECI position in thousands of kilometers. The green line is data created with the SGP4, and the blue line is data from the enhanced simple orbit estimator. The right hand side of the plots shows the difference between the estimates in kilometers. It is seen that although the difference between the two estimates increase over time, the enhanced simple estimator still gives very good results even after a month or more. Figure 4.13 shows the total pointing error in degrees from zero to sixteen

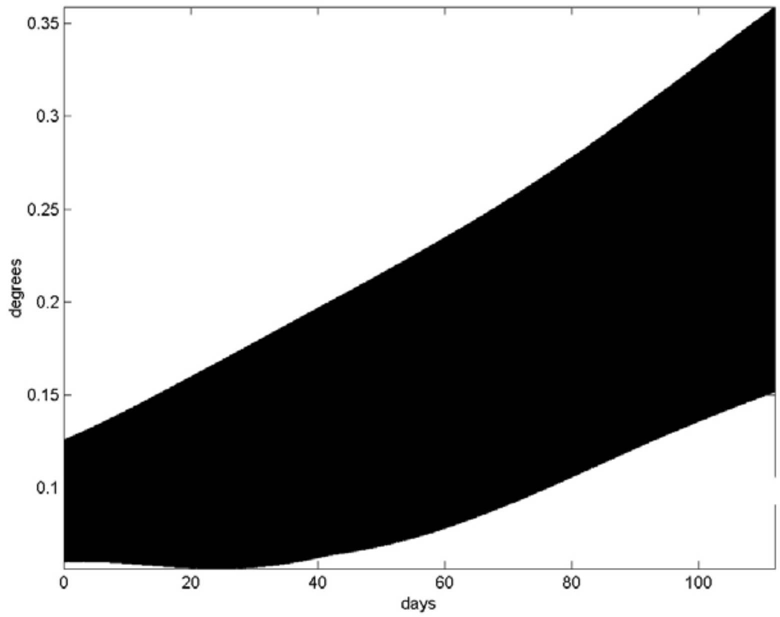


Figure 4.13: Difference in angle from the earth's center.

weeks or 112 days. The thickness of the line is due to the periodic nature of the pointing difference. It would seem that updates in the keplerian elements every one or two months would suffice. The data presented in the two figures above is created with a constant mean motion, both in the enhanced simple orbit estimator, and in the SGP4. First and second time derivatives of the mean motion retrieved from TLE sets are possible to include in the enhanced simple orbit estimator, and simply setting them equal zero if they are not available. Implementing the orbit estimator on the NCUBE can also bring constraints to the choice. The ADCS micro controller must be able to handle the orbit estimator together with its other tasks, thus the complexity of the orbit estimator might have to be minimized. On the implementational level there is also a benefit in the SGP4 model as it has been implemented in several languages. Most of these implementations is based on the FORTRAN implementation found in Hoots and Roehrich (1980). The SGP4 implementation used to calculate the data displayed in Figure 4.12 is a pascal program written by Dr. TS Kelso at Celestrak (2003), the source code of both the enhanced simple orbit estimator, and the pascal code used to interface the SGP4 program of Dr. Kelso is enclosed in Appendix B

4.3 IGRF

For determination of a magnetic vector for comparison with the magnetic vector from the magnetometer, the earth's magnetic field must be known or estimated. As shown in Figure 4.14, the magnetic field is varying strongly over the earth's

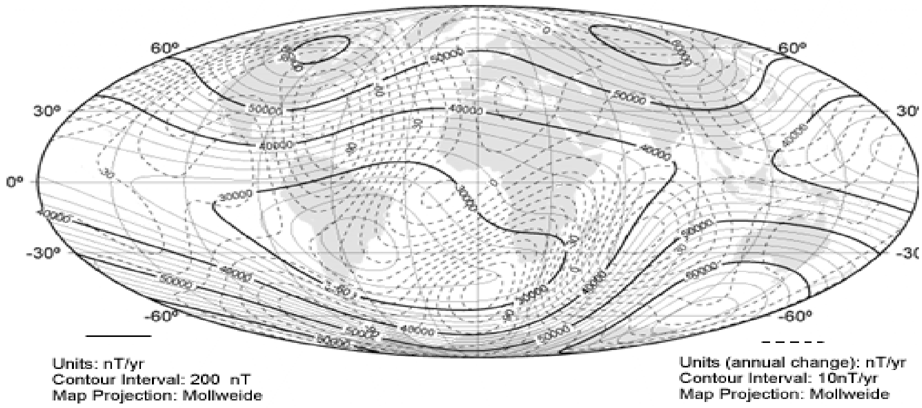


Figure 4.14: Magnitude of the earths magnetic field.

surface, hence a complete table with high resolution is too large to bring on board a satellite's microcontroller. The Earth's magnetic field crudely resembles that of a dipole. On the surface of the earth, the field varies from being horizontal and of magnitude about 30000 nT near the equator, to vertical and about 60000 nT near the poles. The internal geomagnetic field also varies in time, on a time scale of months and longer, in an unpredictable manner.

The International Geomagnetic Reference Field, IGRF, is an attempt by the International Association of Geomagnetism and Aeronomy (2003), IAGA, to provide a model acceptable to a variety of users. It is meant to give a reasonable approximation, near and above the Earth's surface, to that part of the Earth's magnetic field which has its origin in the earths core. At any one time, the IGRF specifies the numerical coefficients of a truncated spherical harmonic series. At present the truncation is at n=10, so there are 120 coefficients. The IGRF model is specified every 5 years, for epochs 1900.0, 1905.0 etc. The latest IGRF model specified is thus the IGRF 2000, which is used in the NCUBE ADCS.

The IGRF model comprises a set of spherical harmonic coefficients called Gauss coefficients, g_n^m, h_n^m, in a truncated series expansion of a geomagnetic potential function of internal origin :

$$V = a \sum_{n=1}^{N} \sum_{m=0}^{n} \left(\frac{a}{r}\right)^{n+1} \left(g_n^m \cos m\phi + h_n^m \cos m\phi\right) P_n^m (\cos\theta) \tag{4.59}$$

where a is the mean radius of the Earth (6371.2 km), and r, ϕ, θ are the geocentric spherical coordinates. r is the distance from the centre of the Earth, ϕ is the longitude eastward from Greenwich, and θ is the colatitude equal 90° minus the latitude. The $P_n^m(\cos\theta)$ are Schmidt quasi-normalized associated Legendre functions of degree n and order m, where $n \geq 1$ and $m \leq n$. The maximum spherical harmonic degree of the expansion is N. Together with the orbit estimator, an estimate of the magnetic field can be made. As the magnetic field revolves with the earth the magnetic field, $\mathbf{B}$, from an IGRF model is in ECEF frame. Using the inverted rotation from equation (4.45) the magnetic field in earth centered orbit frame is

$$\mathbf{B}_{OC} = \left(\mathbf{R}_z(-\Omega+\theta)\,\mathbf{R}_x(-i)\,\mathbf{R}_z(-\omega)\right)^{-1}\mathbf{B}_{ECEF} \tag{4.60}$$

$$\mathbf{B}_{OC} = \mathbf{R}_z(\omega)\,\mathbf{R}_x(i)\,\mathbf{R}_z(\Omega-\theta)\,\mathbf{B}_{ECEF} \tag{4.61}$$

as

$$\mathbf{R}(-\alpha) = \mathbf{R}(\alpha) \tag{4.62}$$

holds for simple rotations. Rotating from earth centered orbit frame to orbit frame is done by

$$\mathbf{B}_O = \mathbf{R}_x\left(\frac{\pi}{2}\right)\mathbf{R}_z\left(\nu+\frac{\pi}{2}\right)\mathbf{B}_{OC} \tag{4.63}$$

$$\mathbf{B}_O = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}\begin{bmatrix} -\sin\nu & \cos\nu & 0 \\ -\cos\nu & -\sin\nu & 0 \\ 0 & 0 & 1 \end{bmatrix}\mathbf{B}_{OC} \tag{4.64}$$

where ν is the true anomaly from Figure 4.8, $\cos\nu$, and $\sin\nu$ is calculated as in equations (4.36) and (4.37). Figure 4.15 shows the three components of the

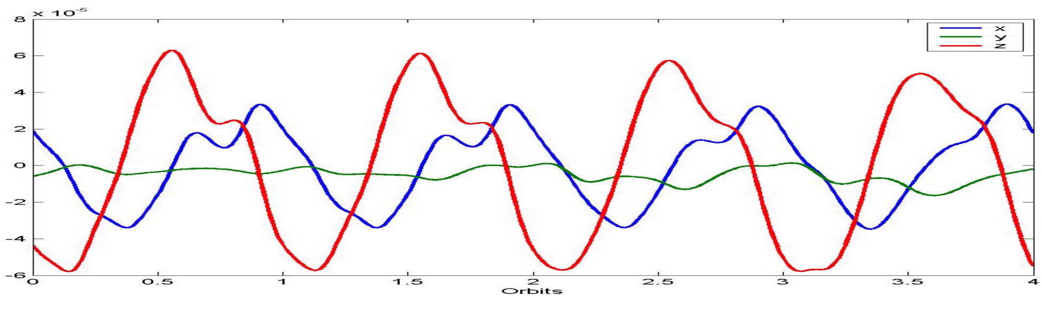


Figure 4.15: The earth's magnetic field from IGRF model

magnetic field in orbit frame for 4 orbits, based on orbit data from the enhanced orbit estimator. The small variations in the y-component of the magnetic field is because the y-axis of the orbit frame point in more or less the same direction all the time.

4.4 Torque Coils

For actuation of the satellite, magnetic torque coils will be used. The momentum

Figure 4.16: The magnetic coil produced at NTNU

produced will, as presented in Chapter 2.2.1, be

$$\mathbf{T} = \mathbf{B} \times \mathbf{M} = \mathbf{B} \times iNA, \tag{4.65}$$

where $\mathbf{B}$ is the earths magnetic field, i is the current in the coil, N is the number of windings in the coil, and A is the area spanned by the coil. To maximize the available control torque, the spanned area and the number of windings should be maximized. This of course leads to more copper wire, and thus more resistance, R, in the coil. As power consumption is given by

$$P = iR^2, \tag{4.66}$$

the wire used in the coils should have as low resistance as possible Given a copper wire with resistance per meter r, and assuming square coils the resistance in the coil becomes

$$R = 4N\sqrt{A} \tag{4.67}$$

and power consumption is then given by

$$P = 16iN^2A \tag{4.68}$$

The fraction

$$\frac{P}{M} = \frac{16iN^2A}{iNA} = 16N \tag{4.69}$$

tells us that increasing the number of windings to create a larger magnetic field, increases the power consumption by a factor of $16N$ more than the magnetic field, while increasing the area or the current to create a larger magnetic field only proportionately increases the power consumption. The coils are produced at the motor winding lab at NTNU which has experience in making low weight coils with hundreds of windings. The coils outer dimensions are 66×66 mm, the inner dimensions are 58×58 mm, and the thickness is 3 mm, as shown in figure 4.16. The number of windings is $N = 140$. Assuming the micro processor controlling the coils drives at $V_{IN} = 3.3$ [V] and can deliver a maximum of $i_{\max} = 25$ [mA], the maximum magnetic field each coil can produce is given by equation (2.2) as

$$\mathbf{M} = iAN = 25 \cdot 10^{-3} A \cdot (0.06m)^2 \cdot 140 = 0.013 Am^2 \tag{4.70}$$

According to a conservative estimate made by Fauske (2002), a magnetic field of $0.02Am^2$ is more than enough to control the satellite. It should therefore be possible to control the satellite with the available micro controller current. Discussions on implementations of a current amplifying and alternative control circuits are given in Chapter 6.

4.4.1 PWM

To control the torque provided by the coils one must control the current flowing through the coil. Pulse Width Modulation, PWM, is a well tested method of controlling the current in an inductive circuit. The voltage V_{IN} on Figure 4.17 is

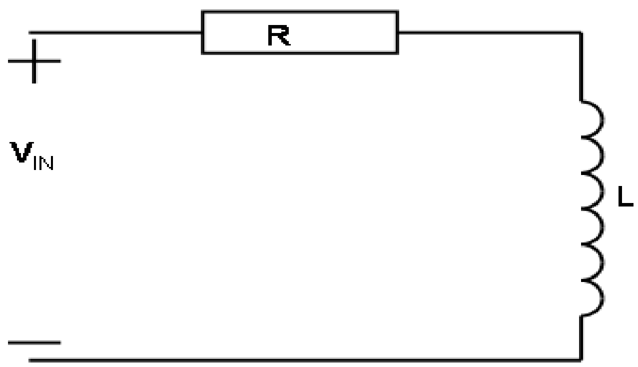


Figure 4.17: The RL circuit representing the magnetic coil

a rectangular waveform where the duty cycle, or the fraction of a period where the signal is kept high, is the controlled parameter. Using high frequencies, the steady state current becomes

$$i = \frac{V_{IN}}{R} \cdot duty \tag{4.71}$$

where R is the resistance in the coil, and $duty$ is the fraction between the pulse width and the period of the signal. Hence, when both V_{IN} and R is known, the equation that calculates the control parameter, $duty$, from the requested current i_{ref} is

$$duty = \frac{R}{V_{IN}} i_{ref} \tag{4.72}$$

To determine how large PWM frequency required for acceptable low ripples on the steady state current, the time constant, $\tau = \frac{L}{R}$, of the RL circuit of Figure 4.17 is needed. The measured values for the coils are $R = 16.5\ \Omega$ and $L = 3.45\ mH$, hence the time constant τ becomes

$$\tau = \frac{L}{R} = \frac{0.00345}{16.5} \approx 0.209 \cdot 10^{-3}[\text{sec}] \tag{4.73}$$

that is approximately 0.2 msec. Solving the current discharge in the RL circuit in the time domain yields

$$i(t) = duty \cdot \frac{V_{IN}}{R}\left(1 - e^{-\frac{t}{\tau}}\right) \tag{4.74}$$

With PWM period T, the time for the discharge is given by

$$t = T\,(1 - duty) \tag{4.75}$$

and the ripple becomes

$$i_{ripple} = duty \cdot \frac{V_{IN}}{R}\left(1 - e^{-\frac{T(1-duty)}{\tau}}\right) \tag{4.76}$$

As seen from Figure 4.18, where ripple is plotted as a function of duty cycle with

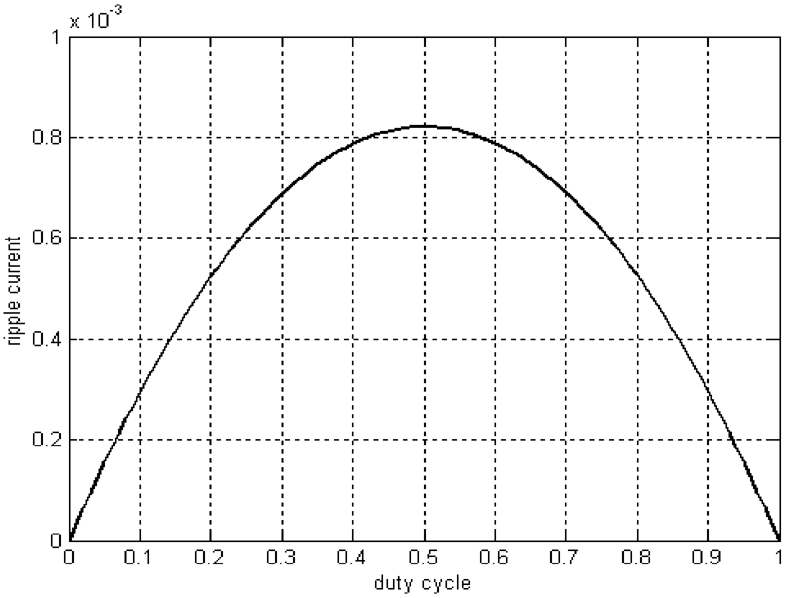


Figure 4.18: Ripple current as a function of duty cycle.

$\frac{\tau}{T} = 50$, the largest ripple occurs for a duty cycle of 0.5. However the largest duty cycle possible is given by the largest current available and becomes:

$$duty_{\max} = \frac{R}{V_{IN}} i_{\max} = \frac{16.5}{3.3} 25 \cdot 10^{-3} \approx 0.125 \tag{4.77}$$

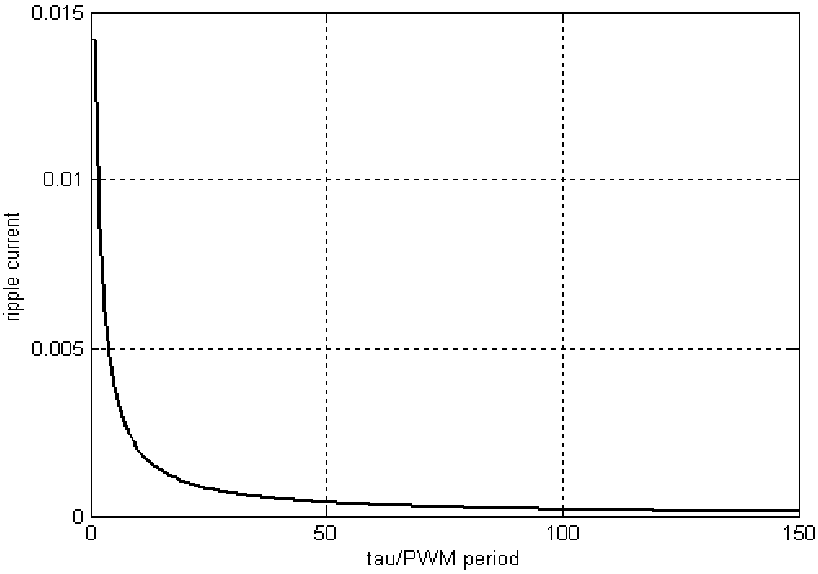


Figure 4.19: Ripple as a function of PWM frequency

In Figure 4.19 the ripple is plotted against the fraction $\frac{\tau}{T}$, which indicates how much shorter the PWM period must be compared with the time constant of the coil. From the figure, it is seen that a period of between 1/50 and 1/100 of the time constant gives very small ripples, and it also shows that there is little to gain from increasing the frequency more. With a time constant $\tau = 0.2 \cdot 10^{-3}$ sec this proposes a PWM frequency between 250 kHz and 500 kHz approximately. To verify these results, a matlab simulink model of the coil is obtained. Using Laplace transformation on the inductor in Figure 4.17 and utilizing Kirchof's first law, the transfer function from voltage to current becomes

$$\frac{i}{V_{IN}} = \frac{1}{R + sL} = \frac{\frac{1}{R}}{\frac{L}{R}s + 1} \tag{4.78}$$

The simulink diagram is shown in Appendix B.4 and the response of a PWM with period $\tau/75$ and duty cycle 0.15 is shown in Figure 4.20. The zoomed part shows the ripples to be approximately $i_{ripple} \approx 0.28mA$,.which is the same result as equation (4.76) gives with $\frac{\tau}{T} = 75$.

To accurately control the current, it is obvious from equation (4.72) that both the voltage over, and the resistance in the coil must be known. One can assume that the voltage is known, but the resistance is temperature dependant and will thus vary in space. A current measurement should be made to have an inner control loop on the current. A measured current indicates that the resistance in the coil is given as

$$R_{new} = \frac{duty \cdot V_{IN}}{i_{measure}} \tag{4.79}$$

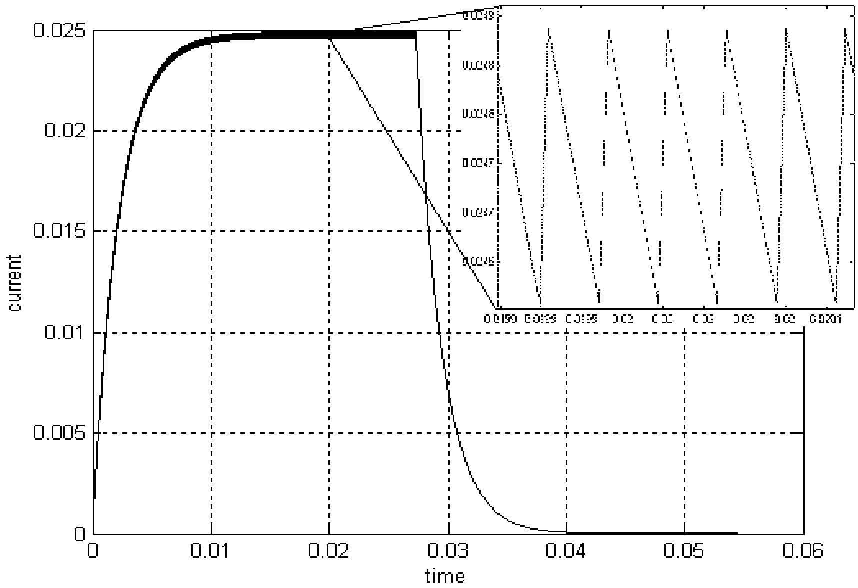


Figure 4.20: PWM controlled current in RL circuit

which indicates that the duty cycle for the given current should instead be

$$duty_{new} = \frac{R_{new}}{V_{IN}} i \tag{4.80}$$

$$duty_{new} = \frac{\frac{duty \cdot V_{IN}}{i_{measure}}}{V_{IN}} i \tag{4.81}$$

$$duty_{new} = duty \frac{i}{i_{measure}} \tag{4.82}$$

As seen in Figure 4.20, there is a transient period before the current reaches its steady state. It is important to wait for the steady state before the current measurement is made, and the duty cycle is updated.

4.5 Satellite model

This modelling was done together with K.M Fauske and F.M. Indergaard. The work was based on a model developed in matlab by Breivik et al (2002) in the prestudy face of the NCUBE project during the spring semester of 2002. The model was transformed from matlab m-files to simulink, and redesigned into a more modular structure.

The satellite attitude is modelled by the satellite's angular momentum, by it's orbit characteristics, and the earth's rotation. Given this information, the attitude can be presented in the body-, orbit-, and ECEF frames. The dynamic equation of motion is derived from the definition of angular momentum. Given the momentum $\mathbf{p}$, and the position vector $\mathbf{r}$, the inertial momentum $\mathbf{h}$ is

$$\mathbf{h} = \mathbf{r} \times \mathbf{p} \tag{4.83}$$

Differentiation and utilization of Newtons second law together with $\mathbf{v} \times \mathbf{v} = 0$ and $\mathbf{p} = m\mathbf{v}$ leads to

$$\frac{\delta}{\delta t}\mathbf{h} = \frac{\delta}{\delta t}\mathbf{r} \times \mathbf{p} + \mathbf{r} \times \frac{\delta}{\delta t}\mathbf{p} = m\mathbf{v} \times \mathbf{v} + \mathbf{r} \times m\mathbf{a} = \mathbf{r} \times \mathbf{F} = \boldsymbol{\tau} \tag{4.84}$$

Where $\boldsymbol{\tau}$ is the vector of all torques applied on the satellite. The angular momentum can alternatively be expressed by the moment of inertia $\mathbf{I}$, and the angular velocity, $\boldsymbol{\omega}$, of the satellite as

$$\mathbf{h} = \mathbf{I}\boldsymbol{\omega} \tag{4.85}$$

This is defined in an inertial frame. To express this in applicable frames the following is needed

$$\begin{aligned}
\mathbf{h}^B &= \mathbf{I}^B \boldsymbol{\omega}_{IB}^B && (4.86)\\
\mathbf{R}_B^I \mathbf{h}^B &= \mathbf{R}_B^I \mathbf{I}^B \boldsymbol{\omega}_{IB}^B && (4.87)\\
\mathbf{h}^I &= \mathbf{R}_B^I \mathbf{I}^B \boldsymbol{\omega}_{IB}^B && (4.88)\\
\mathbf{h}^I &= \mathbf{R}_B^I \mathbf{I}^B \mathbf{R}_I^B \boldsymbol{\omega}_{IB}^I && (4.89)
\end{aligned}$$

Hence the moment of inertia in the inertial frame is

$$\mathbf{I}^I = \mathbf{R}_B^I \mathbf{I}^B \mathbf{R}_I^B \tag{4.90}$$

And thus dependent on the attitude. Differentiating equation (4.88), and applying the time derivatives of the rotation matrix presented in Chapter 3.2 yields

$$\dot{\mathbf{h}}^I = \dot{\mathbf{R}}_B^I \mathbf{I}^B \boldsymbol{\omega}_{IB}^B + \mathbf{R}_B^I \mathbf{I}^B \dot{\boldsymbol{\omega}}_{IB}^B = \mathbf{S}(\boldsymbol{\omega}_{IB}^B)\mathbf{R}_B^I \mathbf{I}^B \boldsymbol{\omega}_{IB}^B + \mathbf{R}_B^I \mathbf{I}^B \dot{\boldsymbol{\omega}}_{IB}^B \tag{4.91}$$

Which expressed in the body frame is

$$\mathbf{R}_I^B\dot{\mathbf{h}}^I = \mathbf{R}_I^B\mathbf{S}(\boldsymbol{\omega}_{IB}^B)\mathbf{R}_B^I\mathbf{I}^B\boldsymbol{\omega}_{IB}^B + \mathbf{R}_I^B\mathbf{R}_B^I\mathbf{I}^B\dot{\boldsymbol{\omega}}_{IB}^B = \mathbf{R}_I^B\mathbf{S}(\boldsymbol{\omega}_{IB}^B)\mathbf{R}_B^I\mathbf{I}^B\boldsymbol{\omega}_{IB}^B + \mathbf{I}^B\dot{\boldsymbol{\omega}}_{IB}^B \tag{4.92}$$

As shown in equation (4.84) $\dot{\mathbf{h}} = \boldsymbol{\tau}$ which combined with equation (4.92) gives

$$\dot{\mathbf{h}}^B = \mathbf{S}(\mathbf{R}_I^B\boldsymbol{\omega}_{IB}^I)\mathbf{I}^B\boldsymbol{\omega}_{IB}^B + \mathbf{I}^B\dot{\boldsymbol{\omega}}_{IB}^B = \boldsymbol{\tau}^B \tag{4.93}$$

This leads to the dynamics implemented in the simulink model

$$\mathbf{I}^B\dot{\boldsymbol{\omega}}_{IB}^B + \boldsymbol{\omega}_{IB}^B \times (\mathbf{I}^B\boldsymbol{\omega}_{IB}^B) = \boldsymbol{\tau}^B \tag{4.94}$$

4.6 Environment model

The total torque $\boldsymbol{\tau}$ acting on the satellite body is made up from several sources. The gravity field from the earth makes the most important contribution.

4.6.1 Gravity torque

The gravity torque can be modelled, according to Kyrkjebø (2000), as

$$\mathbf{g}^B = 3\boldsymbol{\omega}_0^2\mathbf{c}_3 \times \mathbf{I}^B\mathbf{c}_3 \tag{4.95}$$

where $\mathbf{c}_3$ is the direction cosine from the rotation matrix $\mathbf{R}_O^B$, and ω_0 is the angular velocity of the satellite in orbit given by $\omega_0 \approx \sqrt{\frac{GM}{R^3}}$ where G again is Newton's specific gravity constant, M the mass off the earth, and R the radius of the orbit. For better accuracies the R computed in the orbit estimator should be used when calculating ω_0

4.6.2 Magnetic torque

The magnetic torque acting on the satellite is the cross product of the satellites magnetic field and the earths magnetic field. The satellites magnetic field consists of the field produced by the control torquers and magnetic disturbances in the satellite. The Magnetic disturbances is ignored as they are assumed much smaller than the controlled magnetic field.

4.6.3 Ignored sources

The following sources of torque are all ignored due to the fact that they are all very small compared to the main gravity and magnetic torque.

- The gravity torque model does not take into account the tidal forces created by the earth -moon system.
- The sun radiates a vast amount of particles known as solar wind and electromagnetic particles known as solar pressure. Both solar wind and solar pressure will establish a torque on the satellite
- As the satellite orbits in LEO, the atmosphere is still present and the atmospheric drag will be none-zero
- The satellite itself will generate different torques. The deployment of both antennas and boom will produce great torques, but they will be short lived. All electric components on board might produce electromagnetic fields interacting with the earths magnetic field in the same way as the control torque.

Chapter 5

Kalman filter

The Kalman filter is the most widely used method to incorporate multiple sensors and their measurement history in attitude determination for satellites. Both the continous, and discrete, Kalman filter is presented together with some implementational issues regarding the magnetometer and the coarse sun sensor.

5.1 Theory and modeling

The Kalman filter was introduced by R. E. Kalman in 1960, and was rapidly identified as being very useful, especially by engineers in the field of navigation. The Kalman filter is an alternative way of formulating the minimum mean square error filtering problem using state space methods (Brown & Hwang, 1997). Under the assumption of stochastic covariance modelling, the Kalman filter produces the optimal state estimate in the sense of minimizing the covariance of the state variable. With modelling of both the measurement noise and the process noise, the Kalman filter weighs the measurement and the modelled measurement to produce the estimate. As it is a state space approach, it is limited to linear systems, but the Extended Kalman Filter, EKF, incorporates nonlinearities by linearizing the process and observation models around the best state estimate. All variables in all equations below can be time dependent, but this is left out to shorten the notation. With the state variable $\mathbf{x}$, a process is defined as

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{w}, \tag{5.1}$$

where $\mathbf{w}$ is process noise. The measurement, $\mathbf{z}$, of the state is defined as

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{5.2}$$

where $\mathbf{v}$ is measurement noise. The Kalman filter for such a process is shown in Figure 5.1. The Kalman filter gain $\mathbf{K}$ is defined by

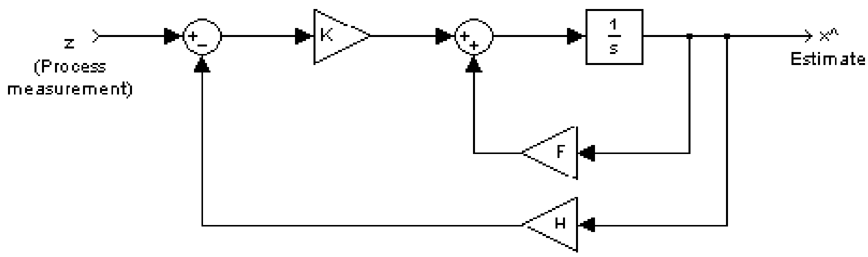


Figure 5.1: Kalman filter

$$\mathbf{K} = \mathbf{P}\mathbf{H}^T\mathbf{R}^{-1} \tag{5.3}$$

where $\mathbf{R}$ is the covariance matrix for the measurement noise and $\mathbf{P}$ is the covariance of the state variable $\mathbf{x}$ given by the solution of

$$\dot{\mathbf{P}} = \mathbf{F}\mathbf{P} + \mathbf{P}\mathbf{F}^T - \mathbf{P}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P} + \mathbf{G}\mathbf{Q}\mathbf{G}^T \tag{5.4}$$

where $\mathbf{Q}$ is the covariance matrix for the process noise. In the case of the EKF $\mathbf{F}$ and $\mathbf{H}$ are substituted with linearizations and the model of Figure 5.1 could be replaced by a nonlinear one. For a complete presentation of the Kalman filter see Brown & Hwang (1997).

5.2 Discrete Kalman filter

The Kalman filter is due to the discrete nature of many forms of measurements often described with discrete equations. The digital magnetometer that is to be used on the NCUBE satellite delivers discrete measurements, and the Kalman filter will be implemented on a micro controller, hence the discrete Kalman filter form will be used. Time is given in discrete intervals and denoted with subscript k. The process is given by

$$\mathbf{x}_{k+1} = \boldsymbol{\phi}_k \mathbf{x}_k + \mathbf{w}_k \tag{5.5}$$

where $\boldsymbol{\phi}_k$ is the linearization of $\mathbf{F}$ from equation (5.1) and $\mathbf{w}_k$ is again the process noise. The measurement is unchanged but discrete

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{5.6}$$

At a given point in time, t_k, before the measurement is made, the estimate of the process state is $\hat{\mathbf{x}}_k^-$ where the superscript $(\hat{\ })$ denotes that it is an estimate, and the superscript $(^-)$ indicates that it is before the measurement or a priori. Based on the a priori estimate and it's error covariance $\mathbf{P}^-$, the Kalman filter gain is computed as

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1} \tag{5.7}$$

Notice that there is a difference between the way the discrete and the continuous Kalman filter gain in equation (5.3) is computed. Incorporating this, the a posteriori state estimate, or the measurement updated state estimate, is calculated as

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \left(\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- \right) \tag{5.8}$$

and the a posteriori state error covariance is updated by one of many possible schemes as described in Brown & Hwang (1997). For instance

$$\mathbf{P}_k = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right) \mathbf{P}_k^- \tag{5.9}$$

Next the a priori estimates of the process state and it's error covariance for time t_{k+1} is

$$\hat{\mathbf{x}}_{k+1} = \boldsymbol{\phi}_k \hat{\mathbf{x}}_k \tag{5.10}$$

$$\mathbf{P}_{k+1}^- = \boldsymbol{\phi}_k \mathbf{P}_k \boldsymbol{\phi}_k^T + \mathbf{Q}_k \tag{5.11}$$

The Kalman filter loop is shown in Figure 5.2. To start it $\mathbf{P}_0^-$ and $\hat{\mathbf{x}}_0^-$, an initial estimate of the state variable and it's error covariance is needed. The output of the filter is the a posteriori state variable estimate $\hat{\mathbf{x}}_k$.

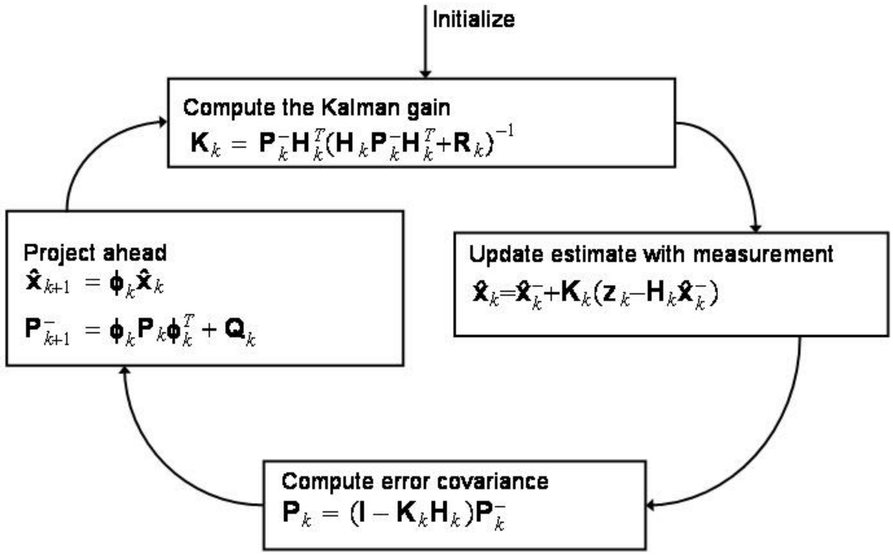


Figure 5.2: The Kalman filter loop

5.3 Implementation

The unit quaternion is used to represent the attitude, and the state variable including the angular velocities becomes

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} & \boldsymbol{\omega}_{IB}^{B^T} \end{bmatrix}^T = \begin{bmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 & \eta & \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T. \tag{5.12}$$

Based on equation (4.94) the dynamics of the angular velocity part of the state propagation is

$$\dot{\boldsymbol{\omega}}_{IB}^B = \mathbf{I}^{-1} \left(\boldsymbol{\tau}^B - + \boldsymbol{\omega}_{IB}^B \times (\mathbf{I} \boldsymbol{\omega}_{IB}^B) \right) \tag{5.13}$$

and the quaternion dynamics in ECEF frame can be expressed from equations (3.22) and (3.23) as

$$\dot{\mathbf{q}}_{EB}^B = \frac{1}{2} \begin{bmatrix} 0 & \boldsymbol{\omega}_{EB_3}^B & -\boldsymbol{\omega}_{EB_2}^B & \boldsymbol{\omega}_{EB_1}^B \\ \boldsymbol{\omega}_{EB_3}^B & 0 & \boldsymbol{\omega}_{EB_1}^B & \boldsymbol{\omega}_{EB_2}^B \\ \boldsymbol{\omega}_{EB_2}^B & -\boldsymbol{\omega}_{EB_1}^B & 0 & \boldsymbol{\omega}_{EB_3}^B \\ -\boldsymbol{\omega}_{EB_1}^B & -\boldsymbol{\omega}_{EB_2}^B & -\boldsymbol{\omega}_{EB_3}^B & 0 \end{bmatrix} \mathbf{q}_{EB}^B \tag{5.14}$$

where $\boldsymbol{\omega}_{EB}^B$ is the angular velocity of the satellite in ECEF frame given by

$$\boldsymbol{\omega}_{EB}^B = \boldsymbol{\omega}_{IB}^B - \mathbf{R}_E^B \boldsymbol{\omega}_{EI}^E \tag{5.15}$$

where $\boldsymbol{\omega}_{EI}^E$ is the earth's angular velocity and $\mathbf{R}_E^B$ in therms of the quaternion is

$$\mathbf{R}_E^B = \begin{bmatrix} \eta^2 - \varepsilon_1^2 - \varepsilon_2^2 + \varepsilon_3^2 & 2(\eta\varepsilon_1 + \varepsilon_2\varepsilon_3) & 2(\eta\varepsilon_2 - \varepsilon_1\varepsilon_3) \\ 2(\eta\varepsilon_1 - \varepsilon_2\varepsilon_3) & -\eta^2 + \varepsilon_1^2 - \varepsilon_2^2 + \varepsilon_3^2 & 2(\varepsilon_1\varepsilon_2 + \eta\varepsilon_3) \\ 2(\eta\varepsilon_2 + \varepsilon_1\varepsilon_3) & 2(\varepsilon_1\varepsilon_2 - \eta\varepsilon_3) & -\eta^2 - \varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 \end{bmatrix} \tag{5.16}$$

The magnetometer measurement $\mathbf{B}^B_{meas}$ is normalized as it is only the direction, and not the length of the vector that gives attitude information. In the remainder of this chapter, all magnetic field vectors, both from the magnetometer, and from the IGRF model, are assumed normalized. The measurement is related to the earth's magnetic field $\mathbf{B}^E$ through

$$\mathbf{B}^B_{meas} = \mathbf{R}^B_E \mathbf{B}^E \tag{5.17}$$

This equation can also give the estimated measurement when $\mathbf{R}^B_E$ is based on the estimate of $\mathbf{q}$, and $\mathbf{B}^E$ is the IGRF model of the earth's magnetic field.

$$\hat{\mathbf{B}}^B = \hat{\mathbf{R}}^B_E \mathbf{B}^E_{IGRF} \tag{5.18}$$

$$\hat{\mathbf{B}}^B = \mathbf{R}^B_E\left(\hat{\mathbf{q}}\right)\mathbf{B}^E_{IGRF} \tag{5.19}$$

Where the $(\hat{\ })$ overstrike again indicates an estimation as defined in Chapter 5.2. Instead of using the standard estimate update with

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}^- + \mathbf{K}\boldsymbol{\nu} \tag{5.20}$$

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}^- + \mathbf{K}\left(\mathbf{B}^B_{meas} - \mathbf{R}^B_E\left(\hat{\mathbf{q}}^-\right)\mathbf{B}^E_{IGRF}\right) \tag{5.21}$$

the updating of the state vector estimate is divided into updating the quaternion part and updating the angular velocity part. $\mathbf{K}_\mathbf{q}$ and $\mathbf{K}_\omega$ is used to denote the two halves of the Kalman filter gain corresponding to the quaternion and angular velocity part respectively. Another scheme for the innovation, $\boldsymbol{\nu}$, is suggested by psiaki (1990)

$$\boldsymbol{\nu} = \mathbf{B}^B_{meas} \times \mathbf{R}^B_E\left(\hat{\mathbf{q}}^-\right)\mathbf{B}^E_{IGRF} \tag{5.22}$$

which is motivated by the fact that the for the cross product between two normalized vectors b and c the following relationship holds

$$a = b \times c \tag{5.23}$$

$$|a| = \sin\alpha \tag{5.24}$$

where α is the angle between the two vectors b and c, and a is the vector of rotation. This means that $\boldsymbol{\nu}$ is proportional with sinus of the error. The corresponding update of the quaternion is suggested as

$$\hat{\mathbf{q}} = \hat{\mathbf{q}}^- \otimes \left[\begin{array}{c} \Delta\mathbf{q}_{ud} \\ \sqrt{1 - |\Delta\mathbf{q}_{ud}|^2} \end{array}\right] \tag{5.25}$$

where

$$\Delta\mathbf{q}_{ud} = \mathbf{K}_\mathbf{q}\boldsymbol{\nu} \tag{5.26}$$

The advantage of this update is twofold. Firstly, it recognizes the properties of the quaternion product as a rotation. Secondly, there is only three free variables in the quaternion update. The unity of the quaternion is thus ensured with this implicit normalization. The angular velocity is updated with the same innovation, but with normal Kalman filter update

$$\hat{\boldsymbol{\omega}}_{IB}^{B} = \hat{\boldsymbol{\omega}}_{IB}^{B-} + \mathbf{K}_{\omega}\boldsymbol{\nu} \tag{5.27}$$

The nonlinear measurement matrix $\mathbf{H}$ must be linearized around the estimate to calculate the Kalman filter gain in equation (5.3). The linearization yields, according to Bak (2001):

$$\mathbf{H} = \begin{bmatrix} 2\mathbf{S}\left(\mathbf{B}^{B}\right) & \mathbf{0} \end{bmatrix} \tag{5.28}$$

This matrix has only rank two which implies that no information about rotation around the magnetic field vector is available. The system is not observable with only the magnetometer measurement, but as the Kalman filter utilizes historical information an estimate can still be computed.

5.3.1 Introducing the crude sun sensor

In principle the sun sensor measurement, and the magnetometer measurement, can be treated similar as they are both reference sensors providing a vector to be compared with a known vector. The Kalman filter is ideal to fuse different measurements as they are modelled with different covariances, and thus will be weighted different in the estimate update through the Kalman filter gain. The measurement matrix including the sun sensor measurement will be

$$\mathbf{H} = \begin{bmatrix} 2\mathbf{S}\left(\mathbf{B}^{B}\right) & \mathbf{0} \\ 2\mathbf{S}\left(\mathbf{v}_{S}^{B}\right) & \mathbf{0} \end{bmatrix} \tag{5.29}$$

where $\mathbf{v}_{S}^{B}$ is the sun vector in body coordinates from equation (4.18).

Chapter 6

NCUBE ADCS implementation

The implementation issues discussed in this chapter are the results of work done in cooperation with Eystein Sæther and associate professor Amund Skavhaug. Sæther has the responsibility of implementing the AIS payload on board data handler, AIS-OBDH, and telecommand decoder in the NCUBE satellite, see Figure 1.2.

The block diagram in Figure 6.1 shows the ADSC micro controller and it's interfaces towards the sensors, actuators, and the rest of the satellite. The micro controller, ATmega32L, the same controller chosen for the AIS-OBDH, is responsible for the communication with the rest of the satellite, the sensors, and actuators. The Kalman filtering, orbit estimation, IGRF modelling and con-

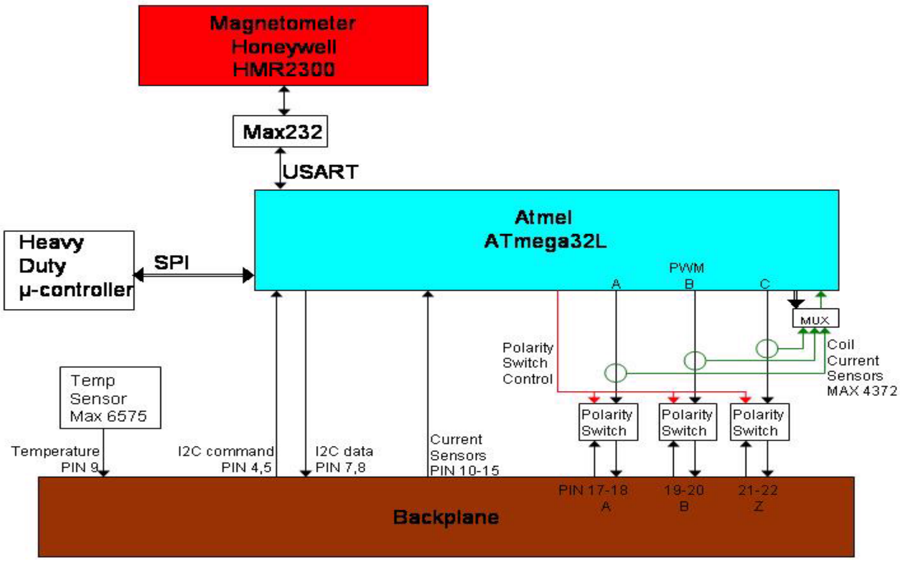


Figure 6.1: NCUBE ADCS block diagram

trol laws, and current allocation will be implemented in another micro controller with floating-point number capability. The simple detubling controller which is proposed used by Fauske (2002), is given by

$$\mathbf{m} = -k\dot{\mathbf{B}} - \mathbf{m}_c \tag{6.1}$$

where $\mathbf{m}$ is the magnetic torque produced by the coils, and k and $\mathbf{m}_c$ are constants. With simple numerical derivation of the magnetometer data for determination of $\dot{\mathbf{B}}$, the calculation of the control torque requires little processing, and can be done with the power economical ATmega32L, hence the power consuming heavy duty controller can be shut off during detumbling. The polarity switch circuits in figure 6.1 can be a simple constellation of four transistors, or if more current than obtainable from the micro controller is needed, a PWM motor controller chip can be used. The often built-in current control loop in such chips also eliminates the need for the current sensors, and the current update scheme described in Chapter 4.4.1.

Further details of ADCS implementation is also discussed in the ADCS design review enclosed in Appendix D.

Chapter 7

Conclusion and recommendations for future work

The aim of the work resulting in this thesis was to provide all necessary information for an Attitude Determination and Control System for the NCUBE satellite. Sensors and actuators should be cosen, and methods for their realization should be investigated regarding both software and hardware. The Kalman filter for optimal utilization of the measuremnts should be established.

7.1 Conclusion

7.1.1 Sensor

To obtain the accuracies needed to utilize a broad band down link antenna, approximately 20°, active control and thus sensors is necessary. The light weigh magnetometer meets the required accuracies and consumes very little power. The choice between the analog and the digital depends on whether the microcontroller has enough analog outputs and whether there is space enough for the digital magnetometer. As there is sufficient space, the digital magnetometer is preferred for three main reasons:

- The degaussing circuit is included, hence the problem with power supply for this is eliminated.
- The digital interface simplifies the implementation.
- Honeywell supplies software for testing and simulation.

As the NCUBE satellite is to be equipped with solar panels, and adding sensors to measure the currents is very cheap, the possibility of using these measurements to aid the magnetic measurements would be a near free extra sensor. The computation of a sun vector is not a difficult calculation, hence the processor is not overstrained by adding this task.

7.1.2 Actuator

As long as attitude control is not the main task to be achieved on the NCUBE satellite, the choice of actuators must be based primarily on space and weight minimization. Viewed in this light, the magnetic torquer will be the most obvious choice. If torque rods small enough for Cubesat implementation can be produced, the advantages due to digital interface and less power consumption favors them. They can only be fitted if the weight budget allows for it, as they are a lot heavier then the coils. If torque rods are chosen they could most likely not be manufactured by NTNU, or other NCUBE participants, as special treatment of cores to minimize hysteresis and achieve good linearity is necessary. This leads to large costs in providing torque rods, and thus eliminates this option in the NCUBE student satellite project.

The current in the magnetic coils will be controlled by Pulse Width Modulation, either directly from a microcontroller, or through som power amplifying or motor control circuit.

7.1.3 Kalman filter

The structure of the magnetometer Kalman filter has been established and implemented in simulink, but for the micro controller implementation, a discrete time Kalman filter will be used. Special quaternion innovation schemes are used to improve the filter. The theory for including the crude sun sensor is also presented.

7.2 Future Work

Modelling of the amount of energy reflected from the earth is necessary to obtain really useful information from the crude solar cell sun sensor. This could prove to be a tough task, and the work of Appel (2003) should be utilized. As the two sensors are to be combined in a Kalman filter, values for their covariances must be determined to achieve optimality.

The Kalman filter with crude sun sensor extension should be fully implemented in simulink with an earth albedo model, and tested for convergence and performance in the case of measurement disappearance, or bad initial values, for both covariance and state variable. It should be looked into how large inaccuracies in system modelling, and measurement covariances that are tolerated, and what influence they have on the attitude estimate.

Testing of the physical system on ground is unfortunately not possible as the gravity gradient is too large for the magnetic torque to overcome it. But all measurements should be tested together with magnetic measurement of the finished satellite.

Bibliography

[1] Appel, P. Attitude Estimation from Magnetometer and Earth-Albedo-Corrected Coarse Sun Sensor Measurements, 2003

[2] Bak, T. Spacecraft Attitude Determination - a Magnetometer Approach. Ph.D. thesis. Aalborg University, Denmark, 2002.

[3] Breivik, Morten, gruppe 3, Landsby 13 Eksperter i Team 2002. Orienteringsregulering av mikrosatellitter. 2002

[4] Brown, R. G & Hwang, P. Y. C. Introduction to Random Signals and applied Kalman Filtering, John Wiley and sons, 1997.

[5] Busterud, B. E., Orienteringsregulering av mikrosatellitter, Department of Engineering Cybernetics, NTNU, 2003

[6] Celestrak at http://www.celestrak.com/, 20.05.2003

[7] Egeland, Olav og Gravdahl, Jan Tommy. Modeling and Simulation for Control. Lecture Notes, Department of Engineering Cybernetics, NTNU, 2001.

[8] Fastrax Ltd, www. fastrax.fi, 20.05.2003.

[9] Fauske, Kjell Magne, NCUBE attitude control, Department of Engineering Cybernetics, NTNU, 2002.

[10] Forsell, Börje. Radionavigation Systems, Prentice Hall, 1991.

[11] Honeywell, www.honeywell.com, 20.05.2003. See Appendix F for some datasheets

[12] Felix R. Hoots and Ronald L. Roehrich, Spacetrack report No. 3:Models for Propagation of NORAD Element Sets, 1980

[13] Kristiansen, Raymond. Attitude Control of a Microsatellite Department of Engineering Cybernetics, NTNU.

[14] Kyrkjebø, Erik. Satellite Attitude Determination - Three-axis attitude determination using magnetometers and a star tracker. Siv.ing. Thesis. Department of Engineering Cybernetics, NTNU, 2000.

[15] Mansfield, M. & O'Sullivan, C. Understanding Physics, Wiley and Praxis, 1998.

[16] Montenbruck, Oliver. GPS Tracking of Microsatellites - PCSat flight Experience, 5th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems 2002.

[17] Psiaki, M.L. Thre-Axis Attitude determination via Kalman Filtering of Magnetometer Data, Journal of Guidance, Control and Dynamics, Vol13, May-June 1990

[18] Santoni, F.et al. The Nanosatellite Ursa Maior Micropropilsion System, 5th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems 2002.

[19] Wertz, J. R. and Larson, W. J. (editors). Space mission analysis and design. Kluwer academic publishers, 1999

[20] ZARM technik, Center of Applied Space Technology and Microgravity, www.zarm-technik.de, 10.02.2003.

Appendix A

NORAD Two-Line Element Set Format

The Norad two-Line Element Set is the format used by NORAD and NASA to describe satellites and their position in orbit. Data for each satellite consists of three lines in the following format:

```
aaaaaaaaaaaaaaaaaaaaaaaa
1 nnnnnu nnnnnaaa nnnnn.nnnnnnnn +.nnnnnnnn +nnnnn-n +n-n n n
2 nnnnn nnn.nnnn nnn.nnnn nnnnnnn nnn.nnnn nnn.nnnn nn.nnnnnnnnnnnn
```

Line 0 is a twenty-four character name (to be consistent with the name length in the NORAD SATCAT). Lines 1 and 2 are the standard Two-Line Orbital Element Set Format identical to that used by NORAD and NASA. The format description is:

Line 1

Column	Description
01	Line Number of Element Data
03-07	Satellite Number
08	Classification (U=Unclassified)
10-11	International Designator (Last two digits of launch year)
12-14	International Designator (Launch number of the year)
15-17	International Designator (Piece of the launch)
19-20	Epoch Year (Last two digits of year)
21-32	Epoch (Day of the year and fractional portion of the day)
34-43	First Time Derivative of the Mean Motion
45-52	Second Time Derivative of Mean Motion (decimal point assumed)
54-61	BSTAR drag term (decimal point assumed)
63	Ephemeris type
65-68	Element number
69	Checksum (Modulo 10) (Letters, blanks, periods, plus signs = 0; minus signs = 1)

Line 2

Column	Description
01	Line Number of Element Data
03-07	Satellite Number
09-16	Inclination [Degrees]
18-25	Right Ascension of Ascending Node [Degrees]
27-33	Eccentricity (decimal point assumed)
35-42	Argument of Perigee [Degrees]
44-51	Mean Anomaly [Degrees]
53-63	Mean Motion [Revs per day]
64-68	Revolution number at epoch [Revs]
69	Checksum (Modulo 10)

All other columns are blank or fixed. Example:

```
NOAA 14
1 23455U 94089A 97320.90946019 .00000140 00000-0 10191-3 0 2621
2 23455 99.0090 272.6745 0008546 223.1686 136.8816 14.11711747148495
```

Appendix B

Source code and Simulink diagrams

B.1 Enhanced Simple Orbit Estimator

The function orbit.m is used to produce a time series of position vectors for comparison with the SGP4 model, while the keplerian2ECEF.m gives ECEF position a given time after the keplerian elements and is intended for NCUBE implementation.

B.1.1 orbit.m

```
function [r_eci]=orbit(keplerian, time, delta)
% Estimates the vector r_eci from the epoch the keplerian elements
% given were valid until 'time' for every 'delta' seconds.
%
% Inputs
% keplerian: Holds the six keplerian elements.
% The 5th keplerian element is the semimajor axis,
% not the mean motion
% time: Scalar value for how long ahead the estimator goes
% delta: Scalar value for the time step
% theta0: The right ascencion of greenwich at the epoch
%
% Output
% r_eci: The vector to the satellite in ECI frame
% A matrix with size 3:1+ceil(time/delta)
%
% Author: Kristian Svartveit
%
inc=keplerian(1);%inclination
```

```
raan=keplerian(2);%Rigth ascension of ascending node
omega=keplerian(3);%argument of perigee
e=keplerian(4);%eccentricity
a=keplerian(5);%semimajor axis
m=keplerian(6);%mean anomaly
t=0:delta:time;
a_earth = 6.378137e6;%earth radius
T_earth_exact = 8.6164130e4; % Length of Sideral Day
w_E = 2*pi/T_earth_exact; % Earth Angular Velocity
G_const = 6.6720e-11; % Gravitational Constant
M_earth = 5.9742e24; % Mass of the Earth
my_g = G_const*M_earth; %Earth Gravitational Constant
n=sqrt(my_g/a^3); %mean motion
J2=1.08284e-3;
%the second gravitational zonal harmonic of the Earth
deg_d2rad_sec=pi/180/(60*60*24);
% Conversion from degree/day to rad/sec
raan_dot_J2=-3/2*n*a_earth^2*cos(inc)./(a.^2*(1-e^2)^2)*J2;
% Perturbation due to nonspherical earth
raan_dot_moon=-.00338*cos(inc)/15*deg_d2rad_sec;
% Perturbation due to the moon
raan_dot_sun=-.00154*cos(inc)/15*deg_d2rad_sec;
% Perturbation due to the sun
raan_dot=raan_dot_J2+raan_dot_moon+raan_dot_sun;
omega_dot_J2=3/4*n*a_earth^2*(5*cos(inc)^2-1)./(a.^2*(1-e^2)^2)*J2;
omega_dot_moon=0.00169*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot_sun=.00077*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot=omega_dot_J2+omega_dot_moon+omega_dot_sun;
raan=raan(1)+raan_dot.*t;
% Propagation in time using simple euler integration
omega=omega(1)+omega_dot.*t;
m=m(1)+n.*t;
m=mod(m,2*pi);
% Solving Kepler's equation E=M+e*sin(E) with Newton's method
E=m;
eps=1;
tol=1e-9;
while max(eps.^2)>tol
   eps=(m+e*sin(E)-E)./(1-e*cos(E));
   E=E+eps;
end
cE=cos(E);
sE=sin(E);
```

```
%argument of perigee sine and cosine
comega=cos(-omega);
somega=sin(-omega);
%raan sine and cosine
craan=cos(-raan);
sraan=sin(-raan);
%Vector from earth's center to satellite in earh centered orbit
rx=a.*(cE-e);
ry=a.*(sqrt(1-e^2)*sE);
rz=zeros(1,length(E));
r=[rx;ry;rz];
%Rotation matrix for inclination about x-axis
rx_i=[1 0 0;0 cos(-inc) sin(-inc);0 -sin(-inc) cos(-inc)];
r_eci=zeros(size(r));
tic
for i=1:length(r)
   %Rotation matrix for argument of perigee about z-axis
   rz_omega=[comega(i) somega(i) 0;-somega(i) comega(i) 0;0 0 1];
   %Rotation matrix for raan about z-axis
   rz_raan=[craan(i) sraan(i) 0;-sraan(i) craan(i) 0;0 0 1];
   r_eci(:,i)=rz_raan*rx_i*rz_omega*r(:,i);
end
```

B.1.2 keplerian2ECEF.m

```
function r_ecef=keplerian2ECEF(keplerian, time, theta0)
% Estimates the vector r_ecef at 'time' seconds
% after the epoch of the keplerian.
%
% Inputs
% keplerian: Holds the six keplerian elements.
% The 5th keplerian element is the semimajor axis,
% not the mean motion
% time: Scalar value in seconds
% theta0: The right ascencion of greenwich at the epoch
%
% Output
% r_ecef: The vector to the satellite in ECEF frame
%
%
% Author: Kristian Svartveit

inc=keplerian(1);%inclination
```

```
raan=keplerian(2);%Rigth ascension of ascending node
omega=keplerian(3);%argument of perigee
e=keplerian(4);%eccentricity
a=keplerian(5);%semimajor axis
m=keplerian(6);%mean anomaly

t=time;

a_earth = 6.378137e6;%earth radius
T_earth_exact = 8.6164130e4; % Length of Sideral Day
w_E = 2*pi/T_earth_exact; % Earth Angular Velocity
G_const = 6.6720e-11; % Gravitational Constant
M_earth = 5.9742e24; % Mass of the Earth
my_g = G_const*M_earth; % Earth Gravitational Constant
n=sqrt(my_g/a^3);%mean motion

% Time derivatives of raan, omega and M
deg_d2rad_sec=pi/180/(60*60*24);

J2=1.08284e-3;%the second gravitational zonal harmonic of the Earth
raan_dot_J2=-3/2*n*a_earth^2*cos(inc)./(a.^2*(1-e^2)^2)*J2;
raan_dot_moon=-.00338*cos(inc)/15*deg_d2rad_sec;
raan_dot_sun=-.00154*cos(inc)/15*deg_d2rad_sec;
raan_dot=raan_dot_J2+raan_dot_moon+raan_dot_sun;
omega_dot_J2=3/4*n*a_earth^2*(5*cos(inc)^2-1)./(a.^2*(1-e^2)^2)*J2;
omega_dot_moon=0.00169*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot_sun=.00077*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot=omega_dot_J2+omega_dot_moon+omega_dot_sun;
theta=mod(theta0+w_E*t,2*pi);%rotation between eci and ecef
raan=raan+raan_dot*t;
omega=omega+omega_dot*t;
m=m(1)+n*t;
% Solving Kepler's equation E=M+e*sin(E) with Newton's method
E=m;
eps=1;
tol=1e-9;
while eps^2>tol
    eps=(m+e*sin(E)-E)/(1-e*cos(E));
    E=E+eps;
end
r=a*[cos(E)-e;sqrt(1-e^2)*sin(E);0];
rx_i=[1 0 0;0 cos(-inc) sin(-inc);0 -sin(-inc) cos(-inc)];
rz_omega=[cos(-omega) sin(-omega) 0;
```

```
         -sin(-omega) cos(-omega) 0;
         0 0 1];
rz_raan_tetha=[cos(-raan+theta) sin(-raan+theta) 0;
              -sin(-raan+theta) cos(-raan+theta) 0;
              0 0 1];
r_ecef=rz_raan_tetha*rx_i*rz_omega*r;
```

B.2 Pascal SGP4 interface

```
Program SGP4_Test;
{$N+}
 Uses CRT,
 SGP_Init,SGP_Conv,
 SGP_Math,SGP_Time,
 SGP4SDP4;
var
 i : integer;
 interval : longint;
 delta,time,tsince,k1,k2 : double;
 pos,vel : vector;
 Fx,Fy,Fz : Text;
 posStr1,posStr2,posStr3 : String;
BEGIN
 Assign(Fx,'fx.txt');
 Rewrite(Fx);
 Assign(Fy,'fy.txt');
 Rewrite(Fy);
 Assign(Fz,'fz.txt');
 Rewrite(Fz);
 sat_data[1,1] := '1 88888U 99275.98708465 .00000000
                  00000-0 00000-0 0 8 ';
 sat_data[1,2] := '2 88888 98.0000 010.0000 0100000
                  45.0000 000.0000 14.89335475 105 ';
 delta := 1;
 ClrScr;
 Convert_Satellite_Data(1);
 time := Julian_Date_of_Epoch(epoch);
 tsince := 0;
 for interval := 0 to 4*28*1440 do
 begin
 SGP(time,pos,vel);
 Convert_Sat_State(pos,vel);
```

```
 tsince := tsince + delta;
 time := time + delta/xmnpda;
 Str(pos[1], posStr1);
 Str(pos[2], posStr2);
 Str(pos[3], posStr3);
 WriteLn(Fx,posStr1);
 WriteLn(Fy,posStr2);
 WriteLn(Fz,posStr3);
 end; {for int}
 Close(Fx);
 Close(Fy);
 Close(Fz);
 repeat until keypressed;
END.
```

B.3 Orbit Estimator with IGRF model

B.3.1 Orbit_with_IGRF.m

```
function [B_orb, r_eci, r_ecef]=
                 orbit_with_IGRF(keplerian, time, delta, theta0)
% Estimates the vector B_orb, r_eci and r_ecef
% from the epoch the keplerian elements
% given were valid until 'time' for every 'delta' seconds.
%
% Inputs
% keplerian: Holds the six keplerian elements.
% The 5th keplerian element is the semimajor axis,
% not the mean motion
% time: Scalar value for how long ahead the estimator goes
% delta: Scalar value for the time step
% theta0: The right ascencion of greenwich at the epoch
%
% Output
% All vectors are matrices with size 3:1+ceil(time/delta)
% They hold x-,y- and z-axis values for each time step
% B_orb: The earths magnetic field in orbit coordinates
% r_eci: The vector to the satellite in ECI frame
% r_ecef: The vector to the satellite in ECEF frame
%
%
% Author: Kristian Svartveit
```

```
%
inc=keplerian(1);%inclination
raan=keplerian(2);%Rigth ascension of ascending node
omega=keplerian(3);%argument of perigee
e=keplerian(4);%eccentricity
a=keplerian(5);%semimajor axis
m=keplerian(6);%mean anomaly
t=0:delta:time;
a_earth = 6.378137e6;%earth radius
T_earth_exact = 8.6164130e4; % Length of Sideral Day
w_E = 2*pi/T_earth_exact; % Earth Angular Velocity
G_const = 6.6720e-11; % Gravitational Constant
M_earth = 5.9742e24; % Mass of the Earth
my_g = G_const*M_earth; % Earth Gravitational Constant
n=sqrt(my_g/a^3);%mean motion
% Time derivatives of raan, omega and M
deg_d2rad_sec=pi/180/(60*60*24);
J2=1.08284e-3;%the second gravitational zonal harmonic of the Earth
raan_dot_J2=-3/2*n*a_earth^2*cos(inc)./(a.^2*(1-e^2)^2)*J2;
raan_dot_moon=-.00338*cos(inc)/15*deg_d2rad_sec;
raan_dot_sun=-.00154*cos(inc)/15*deg_d2rad_sec;
raan_dot=raan_dot_J2+raan_dot_moon+raan_dot_sun;
omega_dot_J2=3/4*n*a_earth^2*(5*cos(inc)^2-1)./(a.^2*(1-e^2)^2)*J2;
omega_dot_moon=0.00169*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot_sun=.00077*(5*cos(inc)^2-1)/15*deg_d2rad_sec;
omega_dot=omega_dot_J2+omega_dot_moon+omega_dot_sun;
theta=mod(theta0+w_E*t,2*pi);%rotation between eci and ecef
raan=raan(1)+raan_dot.*t;
omega=omega(1)+omega_dot.*t;
m=m(1)+n.*t;
% Solving Kepler's equation E=M+e*sin(E) with Newton's method
E=m;
eps=1;
tol=1e-9;
while max(eps.^2)>tol
   eps=(m+e*sin(E)-E)./(1-e*cos(E));
   E=E+eps;
end
cE=cos(E);
sE=sin(E);
%true anomaly sine and cosine
cnu=(cE-e)./(1-e*cE);
snu=(sqrt(1-e^2)*sE)./(1-e*cE);
```

```
%argument of perigee sine and cosine
comega=cos(-omega);
somega=sin(-omega);
%raan sine and cosine
craan=cos(-raan);
sraan=sin(-raan);
%raan+theta sine and cosine
craant=cos(-raan+theta);
sraant=sin(-raan+theta);
%theta sine and cosine
ctheta=cos(theta);
stheta=sin(theta);
%Vector from earth's center to satellite in earh centered orbit
rx=a.*(cE-e);
ry=a.*(sqrt(1-e^2)*sE);
rz=zeros(1,length(E));
r=[rx;ry;rz];
%Rotation matrix for inclination about x-axis
rx_i=[1 0 0;...
 0 cos(-inc) sin(-inc);...
 0 -sin(-inc) cos(-inc)];
[G,H] = IGRF2000;
nmax = 10;
mmax = 10;
Kschmidt = schmidt(nmax,mmax);
r_eci=zeros(size(r));
for i=1:length(r)
   rz_omega=[comega(i) somega(i) 0;-somega(i) comega(i) 0;0 0 1];
   rz_raant=[craant(i) sraant(i) 0;-sraant(i) craant(i) 0;0 0 1];
   rz_raan=[craan(i) sraan(i) 0;-sraan(i) craan(i) 0;0 0 1];
   r_eci(:,i)=rz_raan*rx_i*rz_omega*r(:,i);
   r_ecef(:,i)=rz_raant*rx_i*rz_omega*r(:,i);
   R_i_e = [ctheta(i) -stheta(i) 0;stheta(i) ctheta(i) 0;0 0 1];
   repe=r_ecef(:,i)';
   [A,ctilde,stilde] = recursion(repe,nmax,mmax);
   r_total = sqrt(sum(repe.^2));
   B_ecef(:,i) =...
       bfield(repe,nmax,mmax,Kschmidt,A,ctilde,stilde,G,H, r_total)';
   B_eci(:,i)=R_i_e*B_ecef(:,i);
   B_orb_c(:,i)=rz_omega'*rx_i'*rz_raan'*B_eci(:,i);
   B_orb(:,i)=[1 0 0;0 0 1;0 -1 0]*...
       [-snu(i) cnu(i) 0;-cnu(i) -snu(i) 0;0 0 1]*B_orb_c(:,i);
end
```

B.3.2 schmidt.m

```
%+==================================================================+
%
% Programmers: Carlos Roithmayr Feb 1997
%
%NASA Langley Research Center
%Spacecraft and Sensors Branch (CBC)
%757 864 6778
%c.m.roithmayr@larc.nasa.gov
%
%+------------------------------------------------------------------+
%
% Purpose:
% Compute coefficients that relate Schmidt functions to associated
% Legendre functions.
%
%+------------------------------------------------------------------+
%
% Argument definitions:
% nmax Maximum degree of contributing spherical harmonics
% mmax Maximum order of contributing spherical harmonics
% Kcoefficients that relate Schmidt functions to
%associated Legendre functions (Ref. [1]).
%
%+------------------------------------------------------------------+
%
% References:
%
% 1. Haymes, R. C., Introduction to Space Science, Wiley, New
% York, 1971.
% 2. Roithmayr, C., "Contributions of Spherical Harmonics to
% Magnetic and Gravitational Fields", EG2-96-02, NASA Johnson
% Space Center, Jan. 23, 1996.
%
%+==================================================================+
function K = schmidt(nmax,mmax)
% Seed for recursion formulae
K(2,2) = 1;
% Recursion formulae
for n = 1:nmax
 i=n+1;
 for m = 0:n
```

```
 j=m+1;
 if m == 0
 % Eq. (3), Ref. [2]
 K(i,j) = 1;
 elseif ((m >= 1) & (n >= (m+1)))
 % Eq. (4), Ref. [2]
 K(i,j) = sqrt((n-m)/(n+m))*K(i-1,j);
 elseif ((m >= 2) & (n >= m))
 % Eq. (5), Ref. [2]
 K(i,j) = K(i,j-1)/sqrt((n+m)*(n-m+1));
 end
 end
end
```

B.3.3 IGRF2000.m

```
function [G,H] = IGRF2000
O=1;
G(O+1,O+0) = -29615; G(O+2,O+0) = -2267;
G(O+1,O+1) = -1728; G(O+2,O+1) = 3072;
h_data(O+1,O+1) = 5186; h_data(O+2,O+1) = -2478;
G(O+2,O+2) = 1672;
G(O+4,O+0) = 935; h_data(O+2,O+2) = -458;
G(O+4,O+1) = 787;
h_data(O+4,O+1) = 272; G(O+3,O+0) = 1341;
G(O+4,O+2) = 251; G(O+3,O+1) = -2290;
h_data(O+4,O+2) = -232; h_data(O+3,O+1) = -227;
G(O+4,O+3) = -405; G(O+3,O+2) = 1253;
h_data(O+4,O+3) = 119; h_data(O+3,O+2) = 296;
G(O+4,O+4) = 110; G(O+3,O+3) = 715;
h_data(O+4,O+4) = -304; h_data(O+3,O+3) = -492;
G(O+5,O+0) = -217; G(O+6,O+0) = 72;
G(O+5,O+1) = 351; G(O+6,O+1) = 68;
h_data(O+5,O+1) = 44; h_data(O+6,O+1) = -17;
G(O+5,O+2) = 222; G(O+6,O+2) = 74;
h_data(O+5,O+2) = 172; h_data(O+6,O+2) = 64;
G(O+5,O+3) = -131; G(O+6,O+3) = -161;
h_data(O+5,O+3) = -134; h_data(O+6,O+3) = 65;
G(O+5,O+4) = -169; G(O+6,O+4) = -5;
h_data(O+5,O+4) = -40; h_data(O+6,O+4) = -61;
G(O+5,O+5) = -12; G(O+6,O+5) = 17;
h_data(O+5,O+5) = 107; h_data(O+6,O+5) = 1;
G(O+6,O+6) = -91;
```

```
G(0+8,0+0) = 25; h_data(0+6,0+6) = 44;
G(0+8,0+1) = 6;
h_data(0+8,0+1) = 12; G(0+7,0+0) = 79;
G(0+8,0+2) = -9; G(0+7,0+1) = -74;
h_data(0+8,0+2) = -22; h_data(0+7,0+1) = -65;
G(0+8,0+3) = -8; G(0+7,0+2) = 0;
h_data(0+8,0+3) = 8; h_data(0+7,0+2) = -24;
G(0+8,0+4) = -17; G(0+7,0+3) = 33;
h_data(0+8,0+4) = -21; h_data(0+7,0+3) = 6;
G(0+8,0+5) = 9; G(0+7,0+4) = 9;
h_data(0+8,0+5) = 15; h_data(0+7,0+4) = 24;
G(0+8,0+6) = 7; G(0+7,0+5) = 7;
h_data(0+8,0+6) = 9; h_data(0+7,0+5) = 15;
G(0+8,0+7) = -8; G(0+7,0+6) = 8;
h_data(0+8,0+7) = -16; h_data(0+7,0+6) = -25;
G(0+8,0+8) = -7; G(0+7,0+7) = -2;
h_data(0+8,0+8) = -3; h_data(0+7,0+7) = -6;
G(0+10,0+0) = -2; G(0+9,0+0) = 5;
G(0+10,0+1) = -6; G(0+9,0+1) = 9;
h_data(0+10,0+1) = 1; h_data(0+9,0+1) = -20;
G(0+10,0+2) = 2; G(0+9,0+2) = 3;
h_data(0+10,0+2) = 0; h_data(0+9,0+2) = 13;
G(0+10,0+3) = -3; G(0+9,0+3) = -8;
h_data(0+10,0+3) = 4; h_data(0+9,0+3) = 12;
G(0+10,0+4) = 0; G(0+9,0+4) = 6;
h_data(0+10,0+4) = 5; h_data(0+9,0+4) = -6;
G(0+10,0+5) = 4; G(0+9,0+5) = -9;
h_data(0+10,0+5) = -6; h_data(0+9,0+5) = -8;
G(0+10,0+6) = 1; G(0+9,0+6) = -2;
h_data(0+10,0+6) = -1; h_data(0+9,0+6) = 9;
G(0+10,0+7) = 2; G(0+9,0+7) = 9;
h_data(0+10,0+7) = -3; h_data(0+9,0+7) = 4;
G(0+10,0+8) = 4; G(0+9,0+8) = -4;
h_data(0+10,0+8) = 0; h_data(0+9,0+8) = -8;
G(0+10,0+9) = 0; G(0+9,0+9) = -8;
h_data(0+10,0+9) = -2; h_data(0+9,0+9) = 5;
G(0+10,0+10) = -1;
H(0+10,0+10) = -8;
G=G*1e-9;
H=H*1e-9;
```

B.3.4 recursion.m

```
%+=================================================================+
%
% Programmers: Carlos Roithmayr Dec 1995
%
% NASA Langley Research Center
% Spacecraft and Sensors Branch (CBC)
% 757 864 6778
% c.m.roithmayr@larc.nasa.gov
%
%+-----------------------------------------------------------------+
%
% Purpose:
% Recursive calculations of derived Legendre polynomials and other
% quantities needed for gravitational and magnetic fields.
%
%+-----------------------------------------------------------------+
%
% Argument definitions:
% repe m Position vector from Earth's center, E*, to a
% point, P, expressed in a basis fixed in the
% Earth (ECF): 1 and 2 lie in equatorial plane
% with 1 in the plane containing the prime meridian,
% 3 in the direction of the north pole.
% The units of length are not terribly important,
% since repe is made into a unit vector.
% nmax Maximum degree of derived Legendre polynomials
% mmax Maximum order of derived Legendre polynomials
% A Derived Legendre polynomials
% ctilde See pp. 4--9 of Ref. [1]
% stilde See pp. 4--9 of Ref. [1]
%
%+-----------------------------------------------------------------+
%
% References:
%
% 1. Mueller, A. C., "A Fast Recursive Algorithm for Calculating
% the Forces Due to the Geopotential", NASA JSC Internal Note
% No. 75-FM-42, June 9, 1975.
% 2. Lundberg, J. B., and Schutz, B. E., "Recursion Formulas of
% Legendre Functions for Use with Nonsingular Geopotential
% Models", Journal of Guidance, Control, and Dynamics, Vol. 11,
```

```
% Jan--Feb 1988, pp. 32--38.
%
%+==================================================================+
function [A,ctilde,stilde] = recursion(repe,nmax,mmax)
clear A;
A=zeros(nmax+3,nmax+3); % A(n,m) = 0, for m > n
R_m = sqrt(repe*repe');
rhat = repe/R_m;
u = rhat(3); % sin of latitude
A(1,1)=1; % "derived" Legendre polynomials
A(2,1)=u;
A(2,2)=1;
 clear ctilde
 clear stilde
ctilde(1) = 1; ctilde(2) = rhat(1);
stilde(1) = 0; stilde(2) = rhat(2);
for n = 2:nmax
 i=n+1;
% Calculate derived Legendre polynomials and "tilde" letters
% required for gravitational and magnetic fields.
% Eq. (4a), Ref. [2]
 A(i,i) = prod(1:2:(2*n - 1));
% Eq. (4b), Ref. [2]
 A(i,(i-1))= u*A(i,i);
 if n <= mmax
% p. 9, Ref. [1]
 ctilde(i) = ctilde(2) * ctilde(i-1) - stilde(2) * stilde(i-1);
 stilde(i) = stilde(2) * ctilde(i-1) + ctilde(2) * stilde(i-1);
 end
 for m = 0:n
 j=m+1;
 if (m < (n-1)) & (m <= (mmax+1))
% Eq. I, Table 1, Ref. [2]
 A(i,j)=((2*n - 1)*u*A((i-1),j) - (n+m-1)*A((i-2),j))/(n-m);
 end
 end
end
```

B.3.5 bfield.m

```
%+==================================================================+
%
% Programmers: Carlos Roithmayr Feb 1997
```

```
%
% NASA Langley Research Center
% Spacecraft and Sensors Branch (CBC)
% 757 864 6778
% c.m.roithmayr@larc.nasa.gov
%
%+------------------------------------------------------------------+
%
% Purpose:
% Compute magnetic field exerted at a point P.
%
%+------------------------------------------------------------------+
%
% Argument definitions:
% repe m Position vector from Earth's center, E*, to a
% point, P, expressed in a basis fixed in the
% Earth (ECF): 1 and 2 lie in equatorial plane
% with 1 in the plane containing the prime
% meridian, in the direction of the north pole.
% nmax Maximum degree of contributing spherical harmonics
% mmax Maximum order of contributing spherical harmonics
% K coefficients that relate Schmidt functions to
% associated Legendre functions.
% A Derived Legendre polynomials
% ctilde See pp. 4--9 of Ref. [1]
% stilde See pp. 4--9 of Ref. [1]
% G, H Tesla Schmidt-normalized Gauss coefficients
% R_mean m Mean radius for International Geomagnetic
% Reference Field (6371.2 km)
% bepe Tesla Magnetic field at a point, P, expressed in ECF
% basis
%
%+------------------------------------------------------------------+
%
% References:
%
% 1. Mueller, A. C., "A Fast Recursive Algorithm for Calculating
% the Forces Due to the Geopotential", NASA JSC Internal Note
% No. 75-FM-42, June 9, 1975.
% 2. Roithmayr, C., "Contributions of Spherical Harmonics to
% Magnetic and Gravitational Fields", EG2-96-02, NASA Johnson
% Space Center, Jan. 23, 1996.
%
```

```
%+------------------------------------------------------------------+
%
% Conversion factors:
% 1 Tesla = 1 Weber/(meter-meter) = 1 Newton/(Ampere-meter)
% = 1e+4 Gauss = 1e+9 gamma
%
%+==================================================================+
function bepe =...
        bfield(repe,nmax,mmax,K,A,ctilde,stilde,G,H, r_total)
% The number 1 is added to degree and order
% since MATLAB can't have an array index of 0.
e1=[1 0 0];
e2=[0 1 0];
e3=[0 0 1];
rmag = sqrt(repe*repe');
rhat = repe/rmag;
u = rhat(3); % sin of latitude
bepe = [0 0 0];
% Seed for recursion formulae
scalar = r_total*r_total/(rmag*rmag);
for n = 1:nmax
% Recursion formula
 scalar = scalar*r_total/rmag;
 i=n+1;
 for m = 0:n
 j=m+1;
 if m <= mmax
 ttilde(i,j) = G(i,j)*ctilde(j) + H(i,j)*stilde(j);
% ECF 3 component {Eq. (2), Ref. [2]}
 b3(i,j) = -ttilde(i,j)*A(i,j+1);
% rhat component {Eq. (2), Ref. [2]}
 br(i,j) = ttilde(i,j)*(u*A(i,j+1) + (n+m+1)*A(i,j));
% Contribution of zonal harmonic of degree n to magnetic
% field. {Eq. (2), Ref. [2]}
 scalar*K(i,j)*(b3(i,j)*e3 + br(i,j)*rhat);
 bepe = bepe + scalar*K(i,j)*(b3(i,j)*e3 + br(i,j)*rhat);
 end
 if ((m > 0) & (m <= mmax))
% ECF 1 component {Eq. (2), Ref. [2]}
 b1(i,j) = -m*A(i,j)*(G(i,j)*ctilde(j-1) + H(i,j)*stilde(j-1));
% ECF 2 component {Eq. (2), Ref. [2]}
 b2(i,j) = -m*A(i,j)*(H(i,j)*ctilde(j-1) - G(i,j)*stilde(j-1));
% Contribution of tesseral harmonic of degree n and order m to
```

```
% magnetic field. {Eq. (2), Ref. [2]}
 bepe = bepe + scalar*K(i,j)*(b1(i,j)*e1 + b2(i,j)*e2);
 end
 end
end
```

B.4 PWM control of RL-circuit

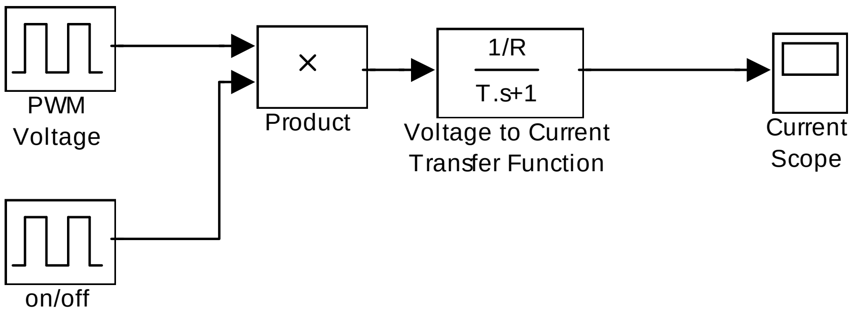
Ripple determination

```
Vpp=3.3;
Rest=16;
duty_cycle=.15;
frac=1:150;
ipp=duty_cycle.*Vpp/Rest.*(1-exp((-1./frac.*(1-duty_cycle))));
i_75=duty_cycle.*Vpp/Rest.*(1-exp((-1./75.*(1-duty_cycle))));
figure(1)
plot(frac,ipp,'k','LineWidth',2);
grid
xlabel('tau/PWM period')
ylabel('ripple current')
frac=50;
duty_cycle=0:0.01:1;
ipp=duty_cycle.*Vpp/Rest.*(1-exp((-1./frac.*(1-duty_cycle))));
figure(2)
plot(duty_cycle,ipp,'k','LineWidth',2);
grid
xlabel('duty cycle')
ylabel('ripple current')
```

PWM Simulink initialization

```
R=16.5;
L=3.45e-3;
T=L/R;
Vinn=3.3;
T_pwm=T/75;
i=0.025;
duty=R*i/Vinn*100;% PWM duty in simulink defined in percentage.
```

Simulink diagram



B.5 Satellite and environment

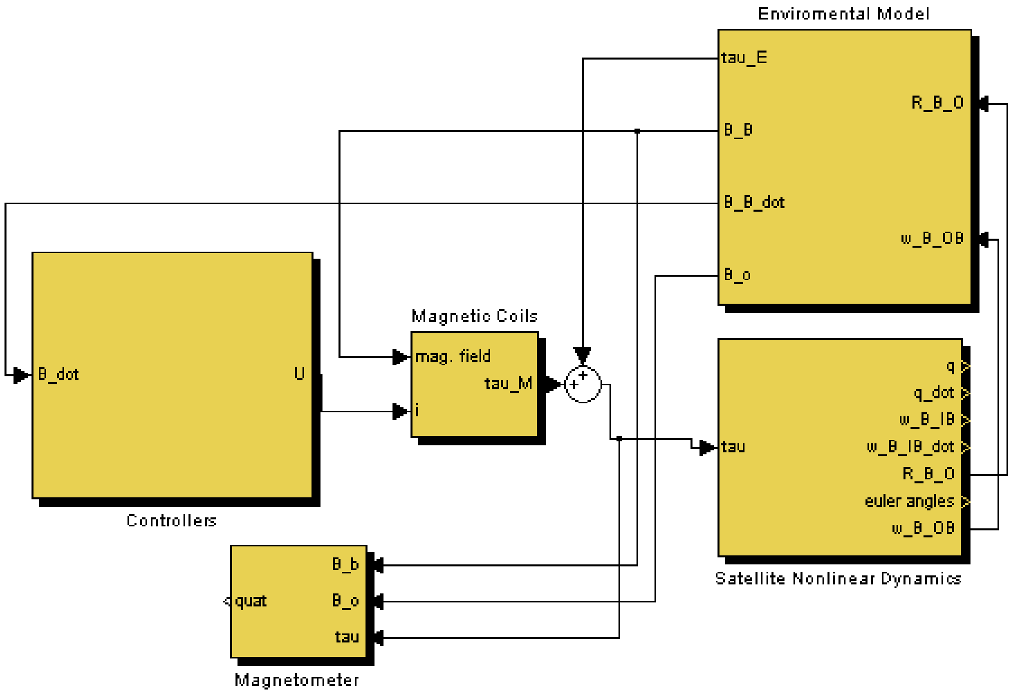
B.5.1 Initialization File

```
%*****************************************************************
% Initialization file for the NCUBE satellite model.
%*****************************************************************
%*****************************************************************
% Inertia matrix
%*****************************************************************
Ix=0.00133; Iy=0.002; Iz=0.00133;
%Ix=3.428; Iy=2.904; Iz=1.275; % without boom
%Ix=80; Iy=100; Iz=30; % boom deployed
InertialMatrix=[Ix 0 0; 0 Iy 0; 0 0 Iz];
I=InertialMatrix;
%*****************************************************************
% Initial values
%*****************************************************************
global r_p;
M_earth = 5.9742e24; % Mass of Earth
omega_o=1.083*10^-3;
h=6.378e6+600.00e3;
r_p=[h 0 0]'; % initial satellite position
% Initial attitude and angular velocities
q_0=euler2q(pi/180*[0 0 0]); % attitude
%w_B_IB_0 = [0 0 0]'; % angular velocity
w_O_IO = [0 -omega_o 0]';
```

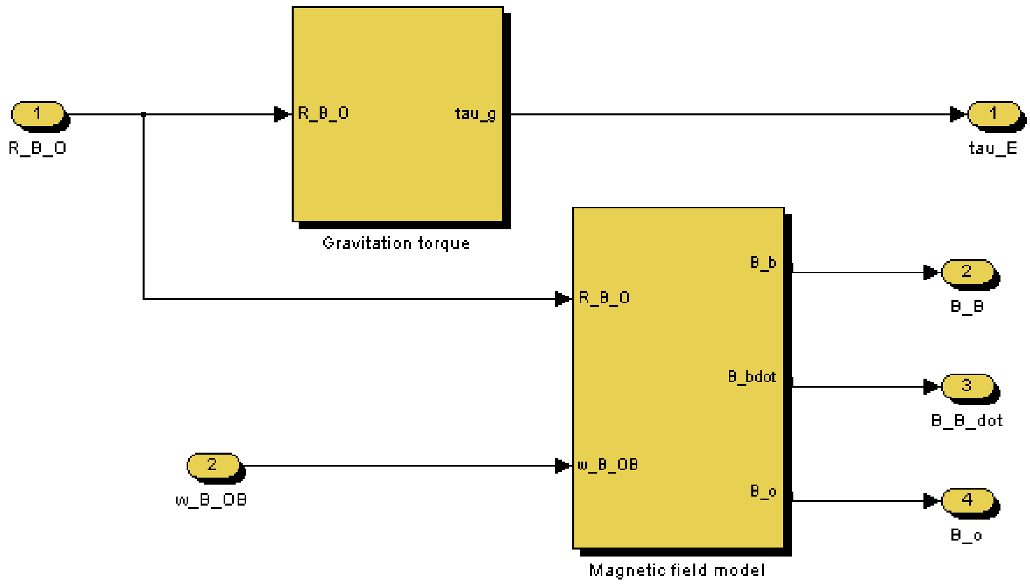
```
R_O_B=Rquat(q_0);
R_B_O=R_O_B';
w_B_OB=[0 0 0]';
c2=R_B_O(:,2);
%w_B_IB_0 = w_B_OB-omega_o*c2;
w_B_IB_0 = [0.2 0.2 0.18]';
w_B_OB=w_B_IB_0+omega_o*c2
%*****************************************************************
% Magnetic field
%*****************************************************************
global G;
global H;
[G,H] = IGRF95;
%*****************************************************************
% Coil parameters
%*****************************************************************
% Number of coil windings
N_x = 30;
N_y = 30;
N_z = 30;
% Coil area [m^2]
A_x = 0.09^2;
A_y = 0.09^2;
A_z = 0.09^2;
% Coil resistance [ohm]
R_x = 48;
R_y = 48;
R_z = 48;
%*****************************************************************
% Controller data
%*****************************************************************
% Detumbling
k = 10e2; % [Am^2s/T]
m_const = [0 0 -0]'; % [Am^2]
A=0.09^2;
N=40;
%*****************************************************************
% Kalman filter parameters
%*****************************************************************
T_f=10
P0=eye(6);
G=eye(6);
R=7e-4*diag([1,1,1]);
```

```
Q=diag([1e-12,1e-12,1e-12,1e-8,1e-8,1e-8]);
```

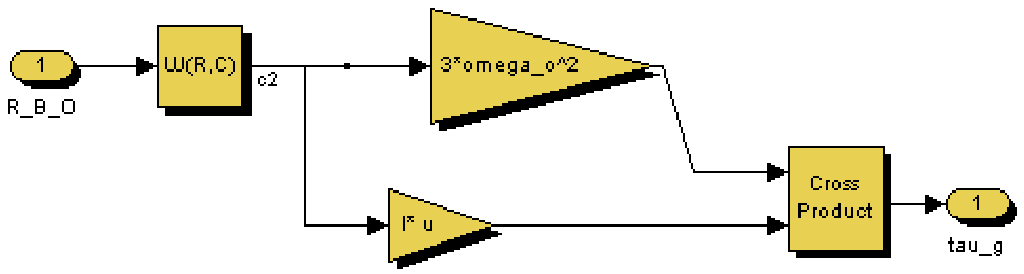
B.5.2 The NCUBE System



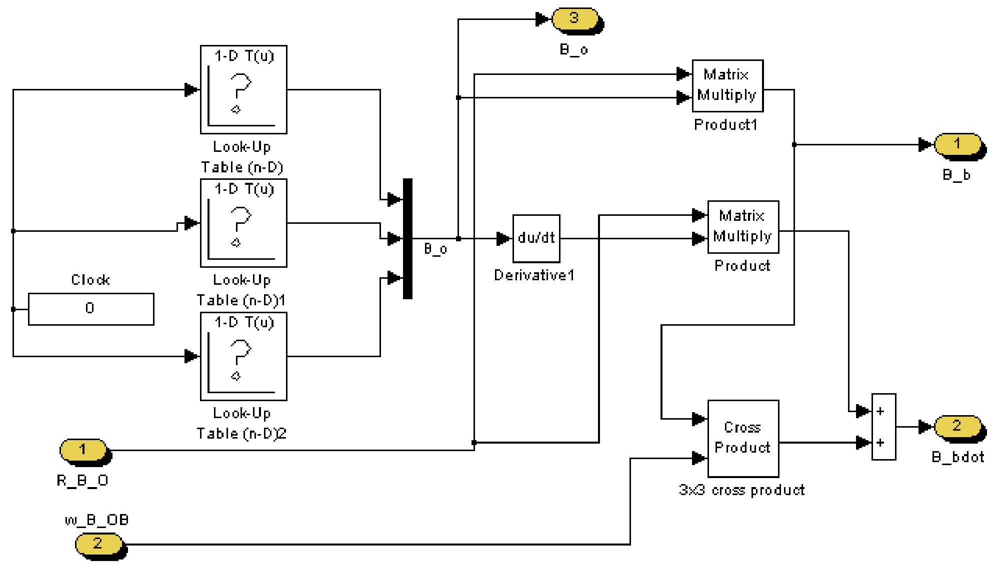
B.5.3 Environmental Model



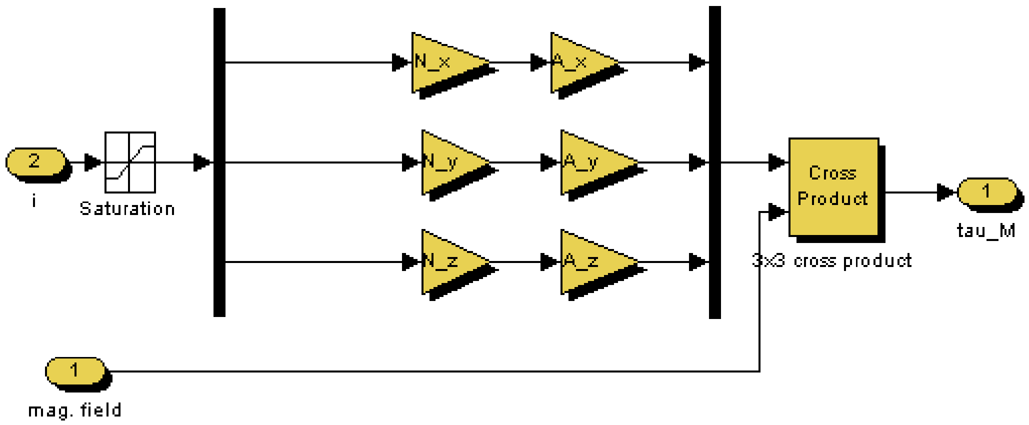
Gravity Torque



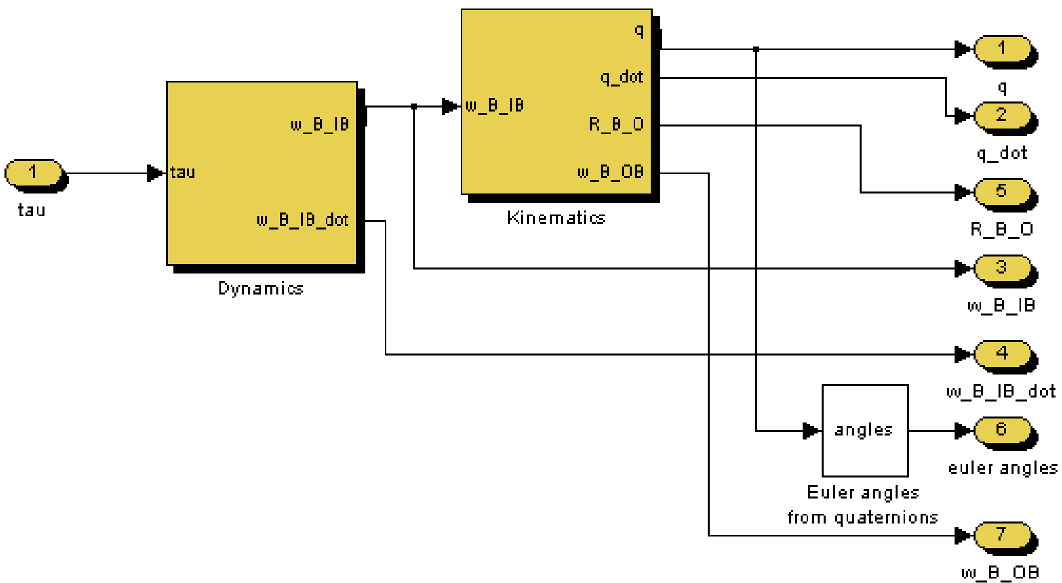
Magnetic Field



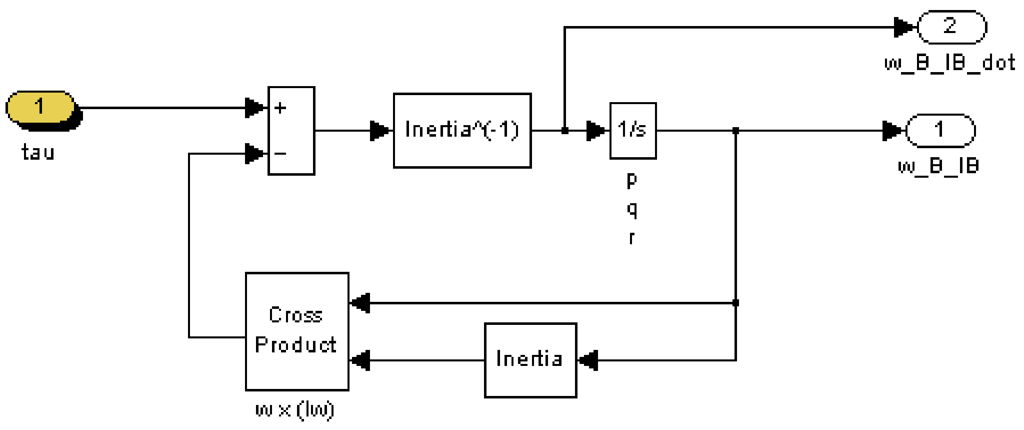
B.5.4 Magnetic Coils



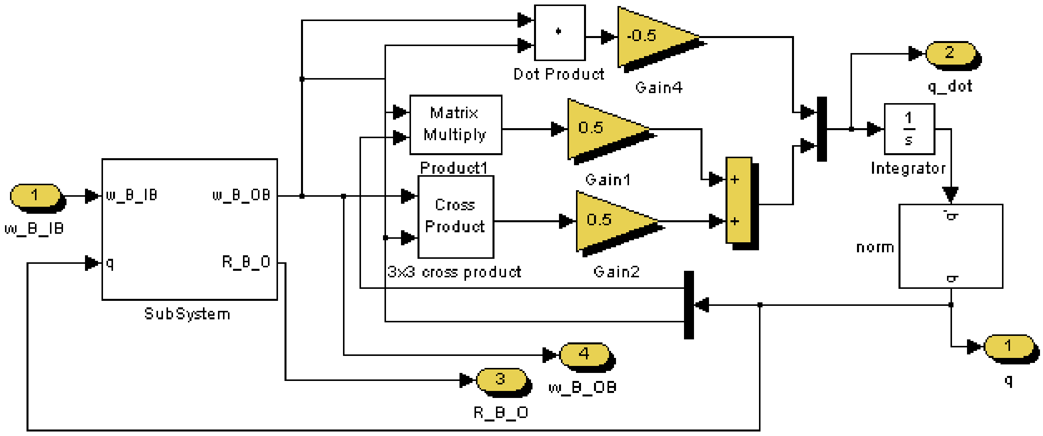
B.5.5 Satellite Nonlinear Dynamics



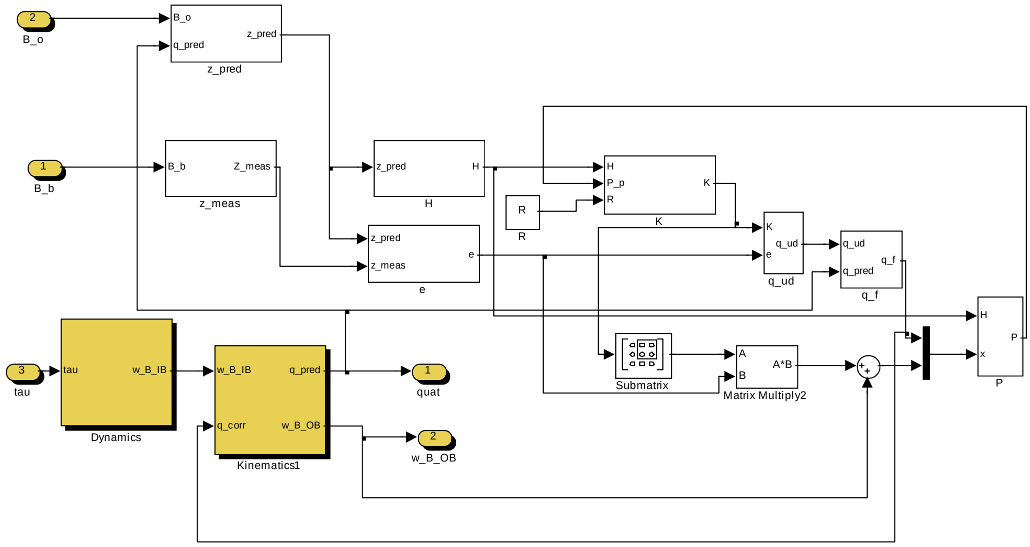
Satellite dynamics



Satellite kinematics



B.5.6 Magnetometer Kalman Filter



Appendix C

Report from the 5th International ESA Conference on Guidance Navigation and Control Systems and the Cubesat workshop

On the 24th and 25th of October I attended the fifth annual ESA conference on GNC. On the 25th I also attended a Cubesat workshop.

The topic of the day on the 24th was "In orbit experience" before lunch and "small satellites" after lunch. Of the more interesting presentations before lunch the EnviSat was, however irrelevant for the Cubesat project, quite intriguing. There was a wide range of impressing imaging and sensing of the earth combined to make valuable research material. The topic of magnet torquers creating magnetic disturbances in the satellite, even when they were not active, which in turn deteriorate the magnetic measurement, was together with some suggested solutions presented. These problems could be relevant, but the solutions are not feasible on a Cubesat.

After lunch professor Bob Twiggs presented the past, the intentions and the future of Cubesats both in the US and worldwide. He emphasized a few points he regards as the most important ones:

- The project should be finished in one year.

The main reason for this is to have the students see the whole life span of a project. If the students are to have hands on experience with systems engineering and real world working experience it must not prolong in time.

- The project should have a customer.

As a part of the real world experience a customer with needs and musts and time limits provide valuable learning.

- All experience should be shared with all other universities

The role model in this thought is the Linux community where all source code is shared. As there is no economical interests in keeping secrets on how the design is made there is little point in not sharing as well. To make it possible for universities to build a Cubesat it might also prove invaluable to have some experience to learn from in order to accomplish tight schedules.

Oliver Montenbruck from the German Space Agency, DLR, held a presentation called "GPS Tracking of Microsatellites - Pcsat Flight Experience" which treated the topic of GPS receivers in space and the results of flying the ORION GPS receiver on the PCsat. He also encouraged the attending GPS manufacturers to either release their source code on their micro chip GPS receivers or to manufacture some without the limitations of altitude and speed. This is necessary if GPS receivers are to be flown on a Cubesat as anything but the main payload as the GPS receivers produced for space is too big and power consuming.

F. Santoni from university of Rome presented "The Nanosatellite Ursa Maior Micropropulsion System", a propulsion system built for Cubesats. They used very much the same system and components used in larger satellites, only scaled down considerably.

In the exhibition outside the conference I spoke to representatives from Zarm and the University of Bremen which produces magnetic torque rods. He was interested in the possibilities of producing torque rods small enough to be used in a Cubesat. There was also a presentation of an attitude determination system combining solar panel data and magnetometers that clearly showed the improvement of utilizing this coarse sun sensor.

On Friday after some less interesting lectures and lunch the Cubesat workshop assembled. The other participants were:

- Bob Twiggs, Stanford University (USA)
- Klaus Shilling, University of Applied Sciences –Weingarten (GER)
- Fabio Santoni, Universita di Roma (ITA)
- Anna Gregorio, University of Trieste (ITA)
- A representative from an other German University
- A representative from ESA
- A group of students from Universita di Roma

Professor Twiggs presented a more technical evaluation of the Cubesat then the one he held the day before. Professor Shilling presented the plans of his university. They include flying the ORION GPS receiver as a payload and building a ground station. The project was not started due to lack of funding. The same was the case with the project on University of Trieste which was in its initialisation. The Roman project was well under way even if it was not very concrete yet.

The NCUBE project has come a lot further and my presentation was well received. The Federated Ground Station Network (J. Cutler, Stanford), already mentioned by professor Shilling, was of particular interest. Mainly because the network enables universities to have a Cubesat program without having to build a ground station and hence the cost will be reduced. The Roman students were also interested in our solar cell manufacturing. At this point professor Twiggs made a remark about his intentions of not only sharing experience but leftovers as well. He had required for free a couple of hundred Gallium Arsenide cells that was not good enough for sale, but good enough for the student satellites. He mentioned also that many of the parts they had used on their satellites had been such leftovers or donated parts.

In the discussions following the presentations the funding of the different projects was the focus of attention. Unlike the NCUBE project the other had not jet been funded. Professor Twiggs' opinion is that the satellites should be funded by the customer, the owner of the payload. It was argued that no customer would pay for this before the Cubesats had proved to work, hence the first projects will need other funding. The ESA representative did not believe that ESA would fund projects entirely but would be willing to organize conferences and help in various ways without funding as well. There was also suggestions that if ESA supported a Cubesat with a small amount of money, other parts would be more willing to fund an "ESA supported project".

The most useful outcome of the conference was Professor Santoni's commitment to make a website gathering all European Cubesat sites and presenting the latest news in pico satellite technology. He also took it upon him to call a new meeting to further discuss the future of Cubesats.

The ESA representative got a copy of all our presentations to show them to the deciding parties of ESA, so that they should have a large array of information before deciding ESAs future involvement in European Cubesat development.

Appendix D

Design review for the ADCS Subsystem

Kjell Magne Fauske, Kristian Svartveit

D.1 Introduction

The primary objective of the ADCS system is to demonstrate that it is possible to estimate the attitude and both actively and passively control the orientation of NCUBE.

The attitude is estimated by using measurements of the Earth's geomagnetic field and currents generated in the solar panels. Varying the currents through three orthogonal electromagnetic coils controls the orientation of the satellite. A gravity gradient boom is used to passively stabilize the satellite.

The most important subsystem functions are

1. Detumble the satellite.
2. Deploy a gravity gradient boom.
3. Estimate attitude, angular velocities and position.
4. Stabilize the satellite.

Step one and two is crucial for the operation of the satellite.

D.1.1 System overview

Figure D.1 shows a diagram of the main hardware components of the attitude determination and control system. Table 1 and Table 2 briefly describe the different components that are connected to the ADCS computer. Note that the boom deployment mechanism is not connected to the ADCS computer. The boom will be deployed directly by a command from the ground station.

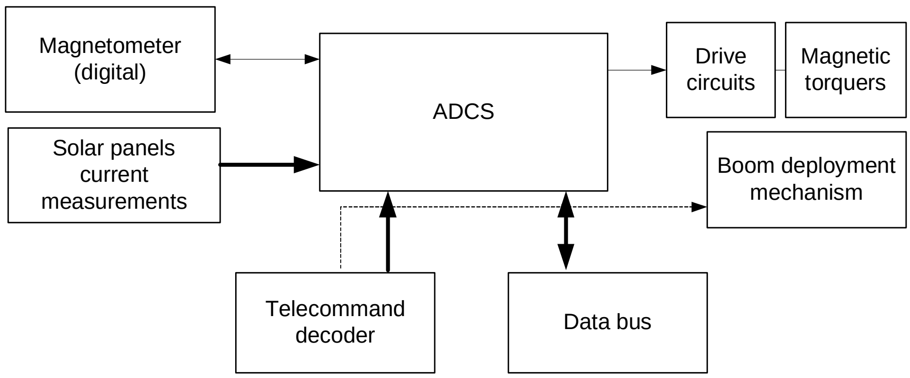


Figure D.1: ADCS block diagram

Table 1 Input

Source	Type	Comment
Telecommand decoder	I^2C	Commands to the ADCS
Data bus	I^2C	Data to and from the ground station
Magnetometer	RS232	16 bit x, y and z magnetic values
Solar panels	Analog	Six analog currents from solar panels
Coil currents	Analog	Currents through the coils are measured

Table 2 Outputs

Destination	Type	Comment
Data bus	I^2C	Status information and state data to ground station
Magnetometer	RS232	The magnetometer has its own command set
Drive circuits	DO	Set points and direction for the three coil currents

D.1.2 Main operation modes

The main operation modes are summarized in Table 3. The operation mode is usually changed by commands received from the Telecommand decoder.

Detumbling

The most important operation mode of the ADCS is the detumbling mode. During detumbling, the initial spin of the satellite is slowed down until the gravity boom can be safely deployed. The controller used for detumbling is a Bdot law $\mathbf{m} = -k\dot{\mathbf{B}} - \mathbf{m}_c$.The controller requires only an estimate of the measured local geomagnetic field's rate of change. No attitude estimation is required.

Attitude estimation

In this mode the satellite's attitude is estimated by reading measurements from the magnetometer and solar panels. These measurements are processed in a state observer. Knowledge of the satellite's position is necessary.

Stabilization

In stabilization mode the actuators are used to control the orientation of the satellite. An estimate of the attitude is necessary for the controllers to calculate the magnetic dipole moment to generate through the electromagnetic coils.

Boom upside-down recovery

This mode is initiated if the satellite is detected to point in the wrong direction. The requirements are the same as in the stabilization mode.

Table 3 Main operation modes

Operation mode	Description
Tumbling (default)	ADCS turned off
Detumbling	Spin is slowed until gravity boom deployment
Attitude estimation	No active control
Stabilization	Active control
Boom upside-down recovery	Turn the satellite if it is pointing upside down

D.2 ADCS hardware interfaces

D.2.1 Edge connector

The ADCS board is connected to the backplane using a 20pin Micronector 200 connector.

D.2.2 Temperature sensor interface

The temperature sensor is used by the power subsystem to monitor the state of the satellite. The sensor must be placed on a suitable place.

D.2.3 Sun sensor interface

Measurements of the currents in the solar panels are used as crude sun sensors. There are six analog current measurements available.

Table 5 Sun sensor interface

Name	Type	Comment
I_A	Analog	Current solar panel A
I_B	Analog	Current solar panel B
I_C	Analog	Current solar panel C
I_D	Analog	Current solar panel D
I_Z	Analog	Current solar panel Z
I_N	Analog	Current solar panel N

D.2.4 Magnetometer interface

The HMR2300 Smart Digital Magnetometer has an RS-232 serial interface, 9600 or 19200 baud. Only three pins are used, RD, TD and GD, see datasheet at http://www.ssec.honeywell.com/magnetic/datasheets/hmr2300.pdf

D.2.5 Magnetorquer interface

The attitude controller calculates a magnetic torque vector. The torque vector is then converted to corresponding currents through each coil. Due to temperature changes, the current must be controlled with a feedback. The easiest way to do this is to drive the coils directly from a microcontroller's digital oututs, DO.

D.3 Software

The purpose of this chapter is to define and describe the functionality of the ADCS software. The ADCS software can be divided into the following main categories:

- Communication.
- Sensor processing
- Attitude estimation

- Magnetic control

D.3.1 Startup

When the ADCS power is turned on, the system goes to idle mode and wait for a command from the Telecommand decoder. Messages and commands from the decoder are handled by interrupts. In idle mode the system uses a minimum of energy and no operations are executed.

The ADCS microcontroller will probably be on all the time. However, the magnetometer will be turned off when power is low or the satellite is communicating with the ground station. When the magnetometer is detected to be off, the microcontroller must go to a mode that consumes a minimum of energy.

D.3.2 Communication and housekeeping

The telecommand decoder sends commands to the ADCS system through an I2C bus. The commands recognized by the ADCS are described in the table below. Note that the ADCS cannot initiate communication with other subsystems. To send or receive data a command has to be received from the Telecommand decoder first.

Table 8 Telecommands

Name	Function
ADCS power save I	No control, measurements on
ADCS power save II	No control, measurements off
ADCS_detumble	Start detumbling mode
ADCS_stabilize	Start stabilization mode
ADCS_invert_boom	Start inverted boom mode
ADCS_off	Turn off ADCS
ADCS_start_data_log	Start to log all signals and states
ADCS_send_data	Download state history.
ADCS_send_status	Download status information
ADCS_reset	reset ADCS
ADCS_orbit_upload	pload orbit parameters
ADCS_mag_upload	upload magnetic field parameters
ADCS_detumble_contr_upload	Upload detumble parameters
ADCS_stabilization_contr_upload	Upload stabilization parameters
ADCS_destab_contr_upload	Upload inverted boom parameters

Data upload

The satellite's position will regularly be uploaded to the ADCS. It will also be possible to upload new controller parameters if necessary. The uploaded parameters must be permanently stored.

```
ADCS_detumble
 The ADCS is set to detumbling mode.
ADCS_stabilize
Start the stabilization mode
ADCS_controller_upload
Upload new controller parameters
ADCS_controller_data_detumble {
k : 16bit
m_c : 16bit
}
ADCS_controller_data_stabilization {
h : 16bit
omega_rb : 16bit
}
ADCS_controller_data_invertedboom {
g : 16bit
}
ADCS_orbit_upload_data {
 epoch_year : 2*8bit
 epoch_day : 11*8bit
 first_time_derivate_mean_motion : 8*8bit
 second_time_derivate_mean_motion : 6*8bit
 bstar_drag_term : 6*8bit
 ephemeris_type : 8bit
 element_number : 5*8bit
 inclination : 7*8 bit
 raotan : 7*8 bit
 eccentricity: 7*8 bit
 argument_of_perigee : 7*8bit
 mean_anomaly : 7*8bit
 mean_motion : 10*8bit
 revolution_number : 5*8bit;
}
```

Data download

The ground station will regularly request state information from the ADCS. Internal state and actuator history will be logged and transmitted to the ground station.

D.3.3 Detumbling mode

The detumbling mode is initiated when the command ADCS_detumble is received. After the boom is deployed we must make sure that the detumbling mode is not entered again by accident. No attitude estimation is required in this mode.

After receiving the command to detumble, we wait until the magnetometer is available. Since the coils and the magnetometer are very close to each other, we must switch between measurement and actuation. The field is measured in 50ms, then

Controller

The controller law is very simple: $\mathbf{m} = -k\dot{\mathbf{B}} - \mathbf{m}_c$

Pseudo code

The pseudo code for the detumbling mode is given below. The sub functions are described in detail in the next subsection.

```
ADCS_detumble{
current_mode = DETUMBLE_MODE
if detumbled = true
then
 change_mode(IDLE)
end
wait_for_magnetometer();
initialize_magnetometer();
while gotonewmode=false
 // Start continous stream of magnetometer readings
 start_magfield_reading();
 //Get at least 2 measurements
 B(k-1) = read_mag_field()
 B(k) = read_mag_field()
 // Stop continuous stream of magnetometer readings
 stop_magfield_reading();
 // Calculate Derivative of magnetic field:
 Bdot =(B(k)-B(k-1))/Sampletime
 //Calculate Magnetic Moment: m = [mx, my, mz]
 m = -k*Bdot-mConstant
 //Calculate current setpoins: i = [ix, iy, iz]
 i = m/(N*A)
```

```
 i = calc_current(m)
 set_current(ix,iy,iz)
 wait(0.5 seconds)
 set_current(0,0,0)
 wait(10*RL_Timeconstant=0.020s)

 // Write data to the log
 write_bdot_log(Bdot_x, Bdot_y, Bdot_z)
 write_i_log(ix, iy, iz)
end
}
```

Sub functions

`wait_for_magnetometer`

Waits for the magnetometer to be switched on. First a status command is sent to the magnetometer. If no answer is received, the magnetometer is assumed to be switched off. The procedure must then wait until the magnetometer is switched on. Two strategies are possible:

1. Regularly poll the magnetometer by sending a status command every 10 seconds.
2. When the magnetometer is switched on, it sends some status data on the RS232 bus. An interrupt can attached to the bus that executes some code when data is received.

It is important that a minimum of energy is used while waiting for the magnetometer to be switched on.

```
wait_for_magnetometer(){
 while not ready
 begin
 result = send_mag_comm(status);
 if result = empty
//magnetometer is switched off. Wait a while
 else ready = true
 end
}
initialize_magnetometer
```

Sends a series of initializing commands to ensure that the magnetometer is initialized properly.

```
initialize_magnetometer(){
 Send_MagCom(*00WE,*00A,*00WE,
 *00R=010,*99WE,*99!BR=S,*00WE,*00TF,*00]R,*00]S)
}
calc_current
```

Calculates the current corresponding to a magnetic moment, with the simple formula

$$i = \frac{\mathbf{m}}{N \cdot A} \tag{D.1}$$

where N = Number of turns in coil and A = Cross section area

```
[ix, iy, iz] = calc_current(mx, my, mz) {
 ix = mx/(N*A);
 iz = my/(N*A);
 iz = mz/(N*A);
}
set_current
```

Sends current set points to coil driving micro controller based on the required currents. This sub routine calculates the duty cycle or width of the pulses on the PWM output. The duty cycle is given as a number between 0 and 1, and represents the fraction of the period that should be high.

```
set_current(IX, IY, IZ){
//Constants:
T// Coil time constant
RX// Coil resistance of coil x
Ry
RZ
V// Voltage over PWM output
i_max// Maximum current from PWM output
duty_max=R*i_max/V// Duty cycle to give maximum current
// Calculating duty cycles for the three PWM output channels
duty_x=IX*RX/V
if duty_x > duty_max
then duty_x = duty_max
end
duty_y=IX*RY/V
if duty_y > duty_max
then duty_y = duty_max
end
duty_z=IX*RZ/V
if duty_z > duty_max
then duty_z = duty_max
```

```
end
Set PWM output duty cycles to [duty_x, duty_y, duty_z]
wait 6*T// Waiting for transients to die
[ix_meas, iy_meas, iz_meas]=Read_current_measurement()
// Reads the current from the three current sensors
// Corecting for uncertanties in coil resistnace.
Set PWM output duty cycles to
[duty_x*IX/ix_meas, duty_y*IY/iy_meas, duty_z*IZ/iz_meas]
RX=duty_x*V/ix_meas
RY=duty_y*V/iy_meas
RZ=duty_z*V/iz_meas
// a timer should be provided to turn off the output
// after a given time, if a stop signal is not received
}
```

D.3.4 Stabilization mode

The stabilization mode can only be initiated after the boom is deployed. The mode consists of attitude estimation and actuation. The output of the attitude estimator is a quaternion , which describes the orientation between the orbit and body reference frame, and the angular velocities of the satellite.

Controller

The stabilization controller is given by

$$\mathbf{m} = -h\boldsymbol{\omega}_{OB}^{B} \times \mathbf{B} \tag{D.2}$$

The controller can however not be activated when the boom is above the horizontal plane. The following control procedure is therefore necessary

The control procedure is:

```
If the boom is below the horizon ( )
 activate the controller
end
```

The pseudo code for the controller law is:

```
m = calc_stabilization_moment(omega, B, e){
// Input:
// omega = [omega1, omega2, omega3] Angular velocities
// B = [B1, B2, B3] Geomagnetic field vector
// Output:
// m = [mx, my, mz] Magnetic moment
```

```
 mx = h*(-omega3*B2 + omega2*B3);
 my = h*( omega3*B1 - omega1*B3);
 mz = h*(-omega2*B1 + omega1*B2);

}


ADCS_stabilization {
wait_for_magnetometer();
while gotonewmode=false
begin
 [q, omega] = estimate_attitude(tbd);
 if log_data then
 write_attitude_log(q, omega);
 write_mfield_log(B);
 end
 if actuate
 begin
 if boom_above_horizon
 m = calc_stabilization_moment(q, B);
 i = calc_current(m);
 actuate(i, time);
 if log_data then write_i_log(q, omega)
 else
 m = [0, 0, 0];
 end
 end
end
}


result = boom_above_horizon(q){
// Input:
// q = [eta, e1, e2, e3]
// Output:
// result = true. Boom points in the right direction
// result = true. Boom points in the wrong direction
 c3 = 1-2(e_1^2+e_2^2);
 if c3 > 0 then
 boom_above_horizon = true
 else
 boom_above_horizon = false
}
```

D.3.5 Inverted boom mode

The inverted boom mode is invoked by a direct command from the ground station. The mode is very similar to the stabilization mode, but a different control law is used.

Controller

The inverted boom mode controller is

$$\mathbf{m} = -g\mathbf{c}_1^B \times \mathbf{B} \tag{D.3}$$

where g = controller gain and

$$\mathbf{c}_1^B = \begin{bmatrix} 1 - 2\left(\varepsilon_2^2 + \varepsilon_3^2\right) \\ 2\left(\varepsilon_1\varepsilon_2 - \eta\varepsilon_3\right) \\ 2\left(\varepsilon_1\varepsilon_3 - \eta\varepsilon_2\right) \end{bmatrix} \tag{D.4}$$

The control procedure in the inverted boom mode is:

```
If the boom is above the horizon ( )
 activate the controller
else change to stabilization mode
```

The pseudo code for the control law is:

```
m=calc_invertedboom_moment(q, B){
 // Input:
 // q = [eta, e1, e2, e3] Quaternion
 // B = [B1, B2, B3] Geomagnetic field vector
 // Output:
 // m = [mx, my, mz] Calculated magnetic moment
 c11 = 1-2*(e2^2+e3^2);
 c12 = 2*(e1*e2-eta*e3);
 c13 = 2*(e1*e3+eta*e2);
 mx = g*(-c13*B2 + c12*B3);
 my = g*( c13*B1 -- c11*B3);
 mz = g*(-c12*B1 + c11*B2);
}

ADCS_invert_boom{
wait_for_magnetometer();
while gotonewmode=false
begin

 [q, omega] = estimate_attitude(tbd);
```

```
 if log_data then
 write_attitude_log(q, omega);
 write_mfield_log(B);
 end
 if actuate
 begin
 if boom_above_horizon = false
 m = calc_invertedboom_moment(q, B);
 i = calc_current(m);
 actuate(i, time);
 if log_data then write_i_log(q, omega)
 else
 change_mode(ADCS_stabilize)
 end
 end
end
}
```

D.3.6 Constants and variables

An overview over the different constants and variables, both those that are updated from the ground station and those that are not.

Orbit Propagator

```
Q0=120;
S0=78;
XJ2=1.082616E-3;
XJ3=-.253881E-5;
XJ4=-1.65597E-6;
XKE=.743699161E-1;
XKMPER=6378.135;
XMNPDA=1440;
AE=1;
//Data updated from groundstation
epoch_year : 2*8bit
epoch_day : 11*8bit
first_time_derivate_mean_motion : 8*8bit
second_time_derivate_mean_motion : 6*8bit
bstar_drag_term : 6*8bit
ephemeris_type : 8bit
element_number : 5*8bit
inclination : 7*8 bit
```

```
raotan : 7*8 bit
eccentricity: 7*8 bit
argument_of_perigee : 7*8bit
mean_anomaly : 7*8bit
mean_motion : 10*8bit
revolution_number : 5*8bit;
```

IGRF model

```
re_earth = 6378137; // Equatorial radius of the Earth
T_earth_exact = 8.6164130e4; // Length of Sideral Day
T_earth = round( T_earth_exact ); // Integer length of Sideral Day
G_const = 6.6720e-11; // Gravitational Constant
M_earth = 5.9742e24; // Mass of the Earth
a_earth = 6.378137e6; // Equatorial radius of the Earth
height = 700.00e3; // Height of the satellite above surface
r_sat = a_earth + height; // Distance from satellite to ECI center
my_g = G_const*M_earth; // Earth Gravitational Constant
w_E = 2*pi/T_earth_exact; // Earth Angular Velocity
w_O = sqrt( my_g/( r_sat^3 ) ); // Satellite Angular Velocity
T_orbit = 2*pi/w_O; // Satellite Orbit Period
v_O = r_sat*w_O; // Satellite Velocity
//Data updated from groundstation
IGRF_gaussian_parameters: 120*16bit
```

Controllers

```
// All constants can be updated from ground station
k:16bit
m_c:16bit
h:16bit
omega_rb :16bit
g:16bit
```

Current set point control

```
Rx// Resistance of coil x
Ry// Resistance of coil z
Rz// Resistance of coil z
T// Time Constant higher then for all coils.
Vmax// Voltage output on PWM
Imax// Maximum current from PWM output
```

ADCS_send_status

```
The information transmitted when status is requested.
state// Which state or mode the ADSC currently is in
detumbled// Whether the detumbling is finnished
magneometer_OK// Whether the magnetometer is functioning
coilX_OK// Whether coil X is working properly
coilY_OK
coilZ_OK
```

Appendix E

Conference puplications

E.1 The 54th International Astronautical Congress, Bremen, Germany

Three axis Attitude Determination and Control System for a picosatellite: Design and implementation

Jan Tommy Gravdahl[1], Amund Skavhaug,
Kristian Svartveit, Kjell Magne Fauske and Fredrik Mietle Indergaard
Email: {Tommy.Gravdahl, Amund.Skavhaug}@itk.ntnu.no, {Svartvei, fauske, indergaa}@stud.ntnu.no
Department of Engineering Cybernetics
Norwegian University of Science and Technology
N-7491 TRONDHEIM
NORWAY

The design and implementation of the Attitude Determination and Control System (ADCS) for a Norwegian picosatellite is presented. The satellite, named Ncube, is based on the CubeSat concept. This means that its size is restricted to a cube measuring 10cm on all sides and that its total mass is restricted to 1kg. Meeting these restrictions represents the main technical challenge of the work. The complete cube includes the payload, ADCS with actuators and sensors, deployable antennas, communication systems, OBDH and power system. Miniaturization is a key approach in order to meet the tight mass budget. The Determination part of the ADCS is solved by integrating measurements from a three-axis magnetometer with current measurements from the solar panels in a Kalman filter. A novel approach is used to employ the solar panes as a crude sun sensor. The Control part is solved by using a combination of magnetic coils and

[1]Contact author, Email: Tommy.Gravdahl@itk.ntnu.no, Phone: +47 73 59 43 93, fax: +47 73 59 43 99

gravity boom. The control system operates in one of two modes: 1) Detumbling and 2) Stabilization. The control laws are derived using Lyapunov theory, and stringent stability proofs are given. The gravity boom is realized using measurement tape, and a boom release policy guaranteeing release in the right direction will be presented. Simulations of both detumbling, boom deployment and stabilization are presented. The Ncube satellite project is a cooperation between several Norwegian educational, research and indutrial environments. The payload is an automatic identification system, AIS, which is as a mandatory system on all larger ships. It transmits identification and position data messages on the 162 MHz maritime VHF band. The satellite prototype is under construction and launch is planned for the second half of 2003.

E.2 17th AIAA/USU Conference on Small Satellites, Utah, USA

nCube: The first Norwegian Student Satellite

Åge-Raymond Riise, B. C. H. Samuelsen, N. Sokolova,
H. D. Cederblad, J. Fasseland, C. O. Nordin[1]

J. Otterstad, K. M. Fauske, O. E. N. Eriksen, F. M. Indergaard,
K. Svartveit, P. E. Furebotten, E. T. Sæther[2]

[1]Narvik University College (HIN)
Institute for Computing, Electronics and Spacetehnology
N-8514 Narvik, Norway
(+47) 7696 6501
agerr@student.hin.no

[2]Norwegian University of Science and Technology (NTNU)
O.S. Bragstads Plass 2B
N-7491 Trondheim, Norway
(+47) 7359 4472

Advisor:
Dr. Egil Eide[2]
Department of Telecommunications
eide@tele.ntnu.no

ABSTRACT

nCube is a picosatellite complying with the CubeSat standard. It is built completely by students in their final year of their Master education in different Norwegian Institutes. The cross institutional project is mainly sponsored by Norwegian space related industry. The satellite is due to launch March 2004 from Dnepr in Ukraine

The concept incorporates use of a miniaturized version of an Automatic Identification System receiver which will be uploaded the coordinates of reindeer herds, making Norwegian Agricultural College able to track them. The satellite will also be able to surveillance regular marine traffic with certain filter options.

nCube is equipped with instruments to determine the attitude based both on solar cell lighting conditions and measurements on the earth magnetic field. Two techniques of controlling the attitude are implemented; by the use of magnetic coils, and gravity gradient stabilization. Communication with the satellite is achieved by the use of AMSAT frequencies in the amateur band and the AX.25 protocol. The project has built is own ground station, which is situated in Narvik City N 68.26 E 17.25, an additional station will be built at Longyearbyen Svalbard and the ground stations will be added to the Federated Ground Network.

TABLE OF CONTENTS

1. INTRODUCTION

On mission from The Norwegian Space Center and Andøya Rocket Range, four Norwegian universities and educational institutes have since 2001 participated in a program to develop a picosatellite known as nCube.

The four educational institutes involved in the project are Narvik University College, Norwegian University of Science and Technology, Agricultural University of Norway, and University of Oslo.

After the initial phase of the project, several work packages were distributed among these institutes; Mechanical Structure, Power System, Attitude Determination and Control System (ADCS), Payload, Space Communication System (COM), and Ground Segment (GSEG).

The main mission of the satellite is to demonstrate ship traffic surveillance from a LEO satellite using the maritime Automatic Identification System (AIS) recently introduced by the International Maritime Organization (IMO) [1]. The AIS system is based on VHF transponders located onboard ships. These transponders broadcast the position, speed, heading and other relevant information from the ships at regular time intervals. The main objective of the satellite is to receive, store and retransmit at least one AIS-message from a ship. Another objective of the satellite project is to demonstrate reindeer herd monitoring from space by equipping a reindeer with an AIS transponder during a limited experimental period. This part of the project is conducted by the Agriculture Univierstiy of Norway. In addition, the satellite should maintain communications and digipeater operations using amateur frequencies. A third objective is to demonstrate efficient attitude control using a combination of passive gravity gradient stabilization and active magnetic torquers.

By letting students gain first hand experience with space mission design, we hope to stimulate further growth of the already fast growing Norwegian space industry.

The satellite will be placed in a low earth sun synchronous orbit with a perigee of approximately 700km, and as circular as possible. The inclination will be close to 98 degrees. The launch is scheduled to April 2004 from Dnepr, Russia.

2. System Overview

During the early phases of the project it was realized that we needed to develop a system architecture that allowed the four universities to design and test their systems themselves with a minimum number of interfaces to the other systems. Therefore, the basic system architecture does not contain a centralized CPU, but instead, we use a pipelined structure where each subsystem contains their own on board data handlers (OBDH). Figure 1 shows a block diagram of the system architecture. The Terminal Node Controller serves as the communications interface to the VHF receiver and the UHF and S-band transmitters. All telecommands are validated by the Telecommand Decoder who forwards the instructions to each subsystem using the I^2C Telecommand Bus. The main subsystems are the AIS receiver payload, the ADCS system and the Power Management Unit. The Data Selector is used to connect the different subsystems to the TNC during transmission down to the ground station. By using this architecture, it is possible to test and verify each subsystem independently during the implementation phase. It is also possible to turn off each subsystem to save power.

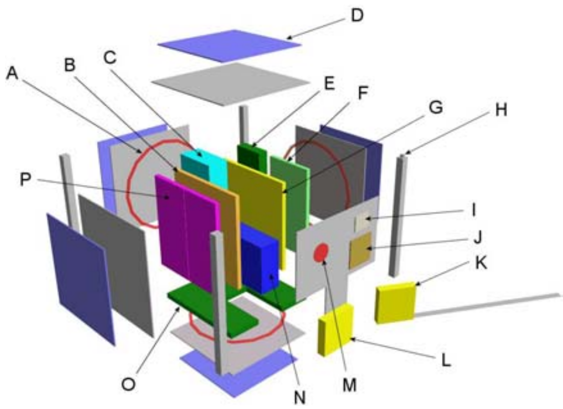


Figure 1. Exploded view of nCube subsystems

Table 1. nCube subsystems

A	Magnetic coils	I	RJ-45 connector
B	VHF receiver & TNC	J	S-band patch antenna
C	Battery pack	K	UHF antenna housing
D	Solar panels	L	VHF antenna housing
E	Magnetometer	M	Flight pin
F	ADCS	N	Gravity boom housing
G	AIS & OBDH	O	Power supply backplane
H	Deployment switch	P	UHF & S-band transmitters

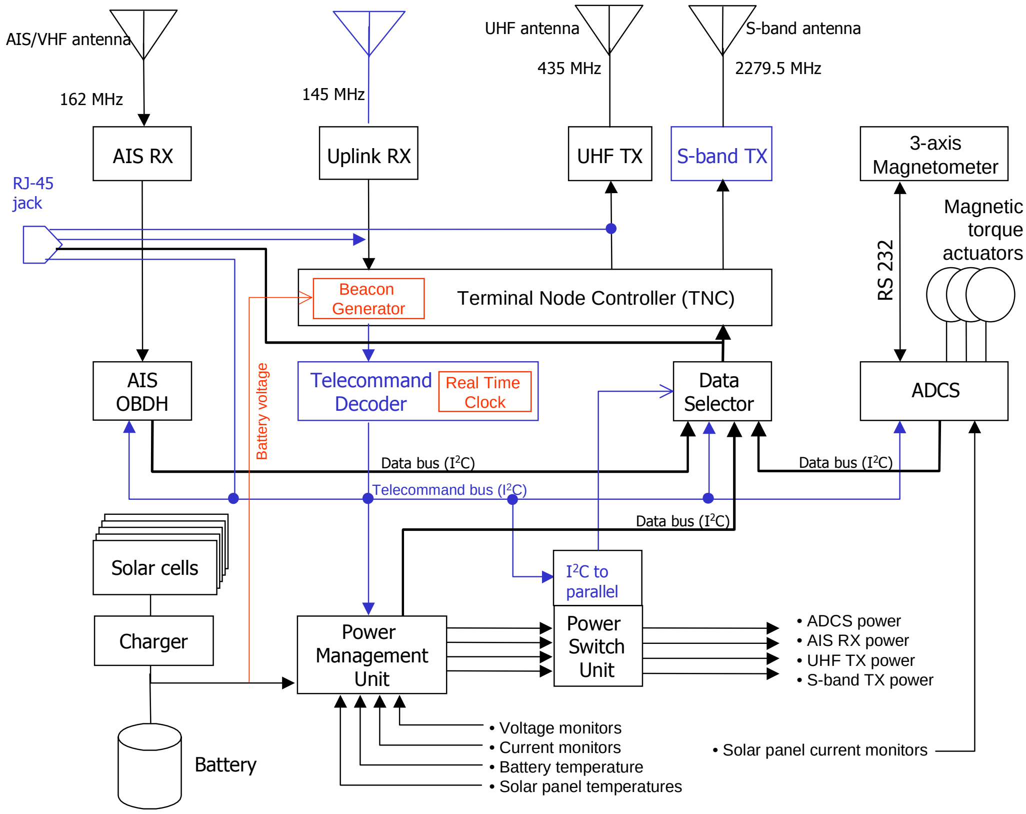


Figure 2. Satellite system architecture.

3. MECHANICAL STRUCTURE

The mechanical structure is designed according to the CubeSat specifications developed by California Polytechnic State University and Stanford University's Space Systems Development Laboratory [2]. For attitude stabilization, the satellite contains a 1.5 meter long deployable gravity gradient boom consisting of steel measuring tape and a counterweight of 40 grams at the outer end. The gravity gradient boom also serves as a VHF antenna for the payload described in Section 7.

One of the side panels, the nadir surface, and houses two deployable VHF/UHF monopole antennas made of steel measuring tape, an S-band patch antenna, the deployable gravity gradient boom, and an I/O interface for ground support. Figure 2 shows a photo of the nadir surface where one of the two antenna containers has been released to the open position. During launch, the monopole antennas are stowed inside the antenna containers until the containers are opened and the antennas are released.

The release mechanism consists of a nylon line that keeps the antenna containers and gravity boom in place. A nichrome wire is used to melt the nylon causing the antennas to rapidly uncoil. The same materials and techniques are used for the gravity gradient boom release mechanism. The antenna deployment is done automatic after the satellite is launched from the P-POD, and the gravity boom is released by a telecommand from the ground station.

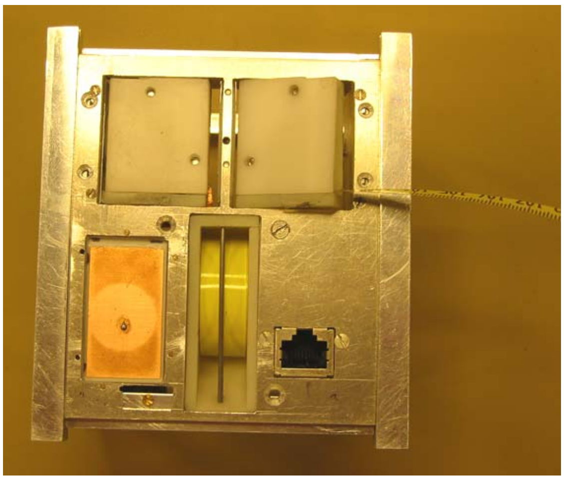


Figure 3. Photo of the nadir surface of nCube

A kill switch is implemented in the design. This switch should physically switch all power off in the satellite, so when stacked in the launch pod, no error should cause a malicious early deployment of booms and antennas, and in the same time conserves power for the early stages of the space mission.

4. POWER SUPPLY SYSTEM

Since the mission endurance is expected to be at least 3 months, using dry cell batteries would not be sufficient for delivering electrical power to the satellite. Due to the weight constraints, the power system will use commercial off the shelf Lithium Ion batteries found in most handheld devices today. These batteries will be precharged before launch such that the satellite can execute initial operations such as detumbling, antenna deployment, and gravity gradient boom deployment. Five of the satellite satellite's six surfaces will be covered by monocrystaline solar cells that are manufactured by Institute for Energy Technology (IFE), Norway. These cells are used to both power the satellite and to charge the batteries to prepare the satellite for the eclipse portion of the orbit. Figure 3 shows a block diagram of the power supply system.

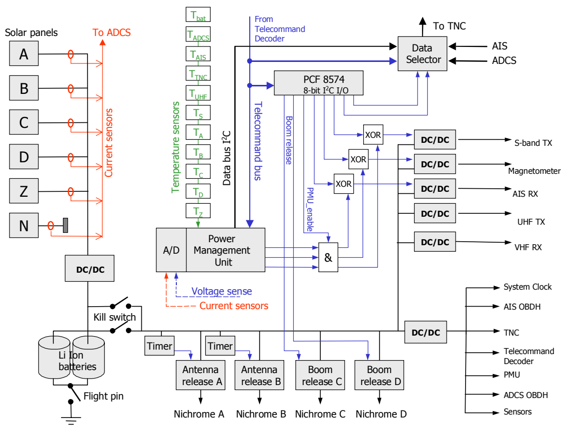


Figure 4. Power supply subsystem.

The power system is equipped with its own micro controller which is able to autonomously power subsystems in a predetermined prioritized order. The only subsystem able to override the power system is the Telecommand Decoder described in the COM section. The COM system is always powered. The different subsystems have different power demands, and require different voltages. The power subsystem internally operates within the voltage range of a typical Lithium Ion cell, 3.7 to 4.2 volts, and all peripheral equipment is interfaced with a set of DC/DC converters adapting to the voltage demand. The Power Management Unit monitors current consumption, battery voltages and temperatures of critical system components during operation.

5. ATTITUDE DETERMINATION AND CONTROL SYSTEM, ADCS

Early after the launch vehicle places the nCube in orbit, the satellite will have a certain amount of rotation about its center of gravity relative to earth. The goal of the ADCS is to point the satellite with the Nadir side towards earth in order to use the broad band antenna.

The attitude is determined by the use of a Honeywell HMR2300 digital three-axis magnetometer inside the satellite. The magnetometer measures the earth's magnetic field. This measurement is compared with an estimate obtained from the International Geomagnetic Reference Field, IGRF, in the Kalman Filter. In addition, the current levels, (I_x, I_y, I_z), from the individual solar panels will be monitored to get the vector towards the sun in body coordinates given as $sun^B_{meas} = \begin{bmatrix} X & 0 & 0 \\ 0 & Y & 0 \\ 0 & 0 & Z \end{bmatrix} \cdot \begin{bmatrix} I_x \\ I_y \\ I_z \end{bmatrix}$, where X, Y and Z are ±1 depending on whether the solar cells on the positive or negative side of the satellite delivering current, and hence points towards the sun. The sun vector in body coordinates is again compared with an estimate of the sun vector based on the day of the year in the Kalman Filter.

The attitude determination is done in a standard quaternion Kalman Filter. Updating the filter with measurements is however done with a scheme suggested by Psiaki [5] where the innovation process is defined as $\nu = B_{meas} \times B_{estimate}$ instead of the usual $\nu = B_{meas} - B_{estimate}$, where both magnetic field vectors are normalized. This innovation is then proportional to the sine of the rotation angle between the two. In addition the update is done using the quaternion product, again considering the difference as a rotation, with a built in quaternion normalization given by $q_{updated} = q_{estimate} \otimes \begin{bmatrix} K \cdot \nu \\ \sqrt{1 - |K \cdot \nu|^2} \end{bmatrix}$, where K is the Kalman filter Gain. The coarse sun sensor made by the solar cells is in essence the same as the magnetometer; A reference sensor giving a vector to be compared with some known vector. This means that they can be treated in the same way in the Kalman Filter. The observation matrix with both sensors will be $H = \begin{bmatrix} 2S(B_{meas}) \\ 2S(sun_{meas}^{B}) \end{bmatrix}$, where S() gives the skew symmetric form of a vector.

Attitude control is primarily achieved by two basic principles:
Gravity gradient stabilization; A gravity gradient boom is deployed and moves the center of gravity so if the rotation are within certain limits, the energy stored in rotation is converted to a nutation like oscillation inside the new systems body cone. The vector of the boom and its counterweight will be rotating around a vector pointing directly towards the center of the earth. If this oscillation can be dampened, it is possible to control the attitude of the satellite such that the nadir surface points towards the earth within limits of ±10 degrees. This is sufficient for antenna pointing.

This dampening can be achieved by direct interaction with the earth's own magnetic field using three magnetic torque coils located inside the satellite. By permitting a current to flow through these coils, a given force vector interacting with the earth's magnetic field can be produced. The currents are pulse width modulated using a stepper motor controller as PWM driver.

Two different schemes for attitude control laws will be used. Firstly, before the gravity boom is released, the satellite must be detumbled. The simple detumble controller is based on the time derivative of the magnetic field. The torque made by the magnetic coils is given by $\mathrm{m} = -\mathrm{k}\dot{\mathrm{B}} - \mathrm{m_c}$, where k and m is constants, and $\dot{\mathrm{B}}$ is obtained by numerical time derivation of the magnetometer measurement. When the satellite is detumbled and the boom is released, a more accurate control law utilizing the attitude measurement from the Kalman Filter is applied.

Figure 5. Attitude control system block diagram.

6. PAYLOAD

The main purpose of nCube is to monitor marine traffic and to track reindeer herds in the Norwegian mountain plateaus, where some of them will be equipped with transponders.
Tracking is based on the Automatic Identification System (AIS), proposed by the International Maritime Organization (IMO), which is specified in IEC-61993 [1].
nCube will receive, filter and forward specific AIS-messages to the Ground Station. Each message contains a 30-bit identifier (MMSI), position, timestamp, velocity, heading and course, in addition to cyclic checksum and flags. The format is following the HDLC-standard, except for extra the 24-bit preamle, used for synchronization of the receiving GMSK modem.

nCube will contain a specially developed AIS VHF receiver shown in Figure 5, using the CMX586 GMSK modem chip to demodulate the Gaussian Minmum Shift Keyed signal. An Atmel AVR 8-bit RISC micro controller, running at 8 MHz will process received data, and store them in an internal EEPROM. The micro controller can be set to store only messages sent from a specific MMSI, to reduce storage use and downlink capacity. More information about the implementation can be found in [3].

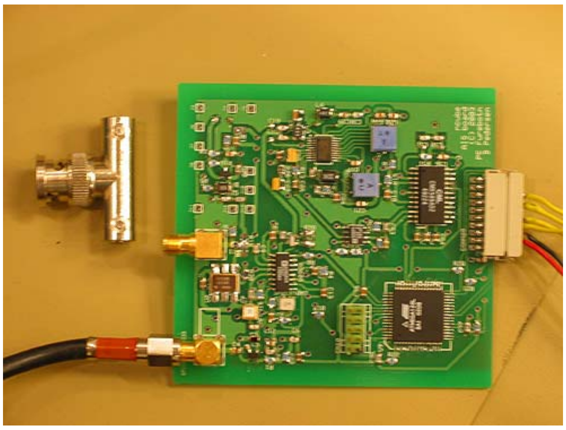


Figure 5. Miniaturized AIS VHF receiver.

7. COMMUNICATION SYSTEM

The communications system is based on using amateur radio frequencies in the VHF and UHF frequency bands. In addition, an S-band transmitter, which originally was developed for sounding rockets, is included for downloading the AIS data. The communications uses the AX.25 protocol with either 1200 bps or 9600 bps data rate. The UHF transmitter has an output power of 0.3W, while the S-band transmitter can output as much as 0.8W to the S-band patch antenna. Monopole antennas with almost omni directional radiation patterns are used for VHF and UHF allowing communications to the satellite even if the ADCS subsystem is not used.

A very simple telemetry format is chosen for monitoring the battery voltage of the satellite. By modulating the carrier wave with an audio tone that is proportional to the battery voltage, any radio amateur can monitor the satellite health without AX.25 equipment. It is also possible to request full telemetry of the housekeeping data from the satellite using the AX.25 protocol. During periods with no scientific or experimental use of the payload, the TNC of the satellite will be open for digipeating (relaying) messages from radio amateurs. This feature is however available only as long as there is enough power in the satellite battery.

Telecommand system description Atmel ATmega 32L microcontroller etc .

8. GROUND SEGMENT

The nCube project builds and operates its own ground station (GS). The GS is based on using standard amateur radio equipment that supports the AX.25 protocols using amateur radio bands covering 144-146 MHz for the uplink and 432 - 438 MHz for the downlink. Using amateur bands is desirable, because it is relatively easy for the students to be granted a license to operate on these frequencies and also because many other student satellites are within these frequencies. One goal with selecting these frequencies is to cooperate with other satellite projects GS's in the future to provide a 24/7 connection ability through a GS abstraction level, defined by an API proposed by the Federated Groundstation Network group[4].

The antenna rig consists of four 19-element 70 cm crossed yagis for the downlink, and two 9-element 2m crossed yagis from Tonna France. The amateur radio transceiver is a Kenwood TS-2000. The antenna is controlled by an azimuth rotator OR2800, an elevation rotator MT 3000, combined with a RC 2800 rotor controller, all from M2.inc, USA. The server application, logger/viewer, estimator and terminal emulator are currently run on an IBM ThinkPad A31p laptop located in the remote operations center of HIN. All software is developed by students, with a commercially available counterpart in backup.

The ground station currently consists of the following elements or subsystems as shown in Figure 6. A real-time satellite position estimator, a two-axis commercial off the shelf game controller, a hardware mono-pulse tracker (or simulator), a terminal emulator, a central server application validating different connections, a data logger and viewer, a hardware antenna positioner, a transceiver, and finally an optional digital wattmeter monitoring standing wave ratio and signal power from the transmitter.

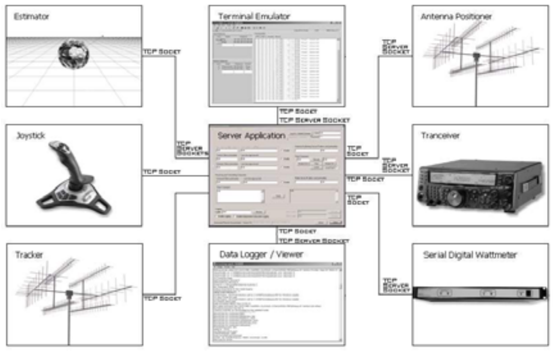


Figure 6. Ground station system overview.

All equipment or subsystems of the ground station are connected through the server application by TCP and IP, and hence allowing the different parts of the complete setup individually to be anywhere internet is accessible. By adding the TCP functionality, we hope to achieve great flexibility when operating ground stations on two or more locations and cooperating in FGN.

The antennas of the GS are presently located on the roof of a building at Narvik University College. The location was selected after a simulation of average 24h contact times at different university locations in Norway. Narvik City is the northernmost among these participating institutes, and hence achieved most contact time for LEO satellites in near polar orbits.

The satellite will be tracked by using a real-time satellite position estimator, which utilizes two-line element set data from NORAD to calculate the satellite's position and orbit, as well as predicting satellite passes for a number of days in advance. This information is then displayed on-screen using OpenGL graphics. It will calculate azimuth and elevation data necessary to track the satellite from any given location in the world, and relay this data to the antenna rotor controller program.

The most desirable location for a GS in Norway is Longyearbyen on Svalbard/Spitsbergen. The very high latitude of 78 degrees is ideal for supporting polar orbiting satellites. Preparations are made to implement a ground station at Svalbard Satellite Station that is operated by KSAT during 2003/2004. This ground station will be totally automated and connected to FGN allowing other universities to connect to their satellites.

9. TEST PROGRAM

The nCube satellite will be tested in a balloon at 20 km altitude from Andøya Rocket Range in June 2003. The purpose of this test is to verify the functionality of the subsystems over a 3 hour period during the balloon flight. In this way we plan to test a complete operations scenario before launch into orbit. After completed test, the satellite is released from the balloon by remote control and returns to the ground using a parachute.

Extensive testing of the gravity boom release mechanism under micro gravity conditions will be performed in the ESA Student Parabolic Flight Campaign in France in July 2003. The main objective of these tests is to study the behavior of the gravity gradient boom and to measure the moments of inertia after the boom is released.

During Fall 2003, the satellite will undergo environmental tests such as thermal vacuum, vibration, and other required tests before launch.

10. Future Development

The nCube satellite has gained significant interest among Norwegian universities and research institutions. There are already initial plans for payload of the next nCube satellite. Hence, the current nCube architecture and system platform will serve as a basis for future satellites.

An additional ground station will be located at SvalSAT, Longyearbyen giving other universities access to their polar orbiting satellites that uses amateur frequencies for communications.

11. Conclusions

The project have involved many essential tasks regarding development of structures and electronics for use in space, the participants have also gained experience in cross institutional cooperation and project administration. It has been a valuable experience for young students preparing for a role in national and international space activities. It will also probably show the benefit for the contributing industries of investing in educational efforts as a part of research and development.

Acknowledgments

The nCube project has been supported by several Norwegian companies such as Kongsberg Seatex AS [6], Telenor ASA [7], Kongsberg Satellite Services (KSAT) [8] and Andøya Rocket Range (ARR) [9].

References

[1] http://www.iala-aism.org/

[2] http://www.rocketrange.no/ncube/

[3] Cubesat Specifications website

[4] J. Cutler, P. Linder ,A Fox, , “A federated ground station network”, NAG3-2579. http://swig.stanford.edu/pub/publications/SpaceOps2002-T2072.pdf

[5] M. Psiaki. Three axis attitude control determination via Kalman filtering of magnetometer data, Journal of guidance, control and dynamics, Vol13, May-June 1990

[6] www.seatex.no

[7] www.telenor.com

[8] www.ksat.no

[9] www.rocketrange.no

Appendix F

Magnetometer data sheets

Honeywell

Magnetic Products

SMART DIGITAL MAGNETOMETER HMR2300

FEATURES

- Microcontroller Based Smart Sensor
- Low Cost and Easy To Use—Just Plug and Read
- Range of $\pm$2 Gauss—<70 μGauss Resolution
- High Accuracy over $\pm$1 Gauss—<0.5%FS
- Output Rate Selectable—10 to 154 Samples/Sec.
- Three-Axis Digital Output—BCD ASCII or Binary
- RS-232 or RS-485 Serial Output—9600 or 19200

APPLICATIONS

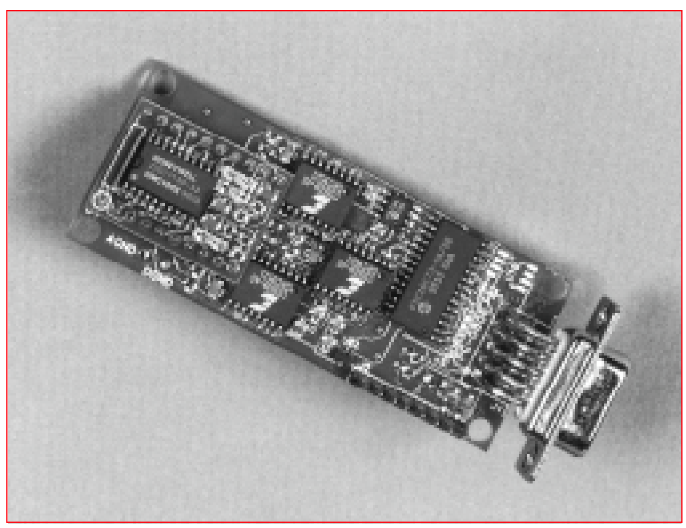
- Compassing—Avionics and Marine
- Remote Vehicle Monitoring (Roll/Pitch/Yaw)
- Process Control
- Laboratory Instrumentation
- Anomaly Detection
- Traffic and Vehicle Detection
- Security Systems

GENERAL DESCRIPTION

Honeywell's three-axis smart digital magnetometer (HMR) detects the strength and direction of a magnetic field and communicates the x, y, and z component directly to a computer. Three independent bridges are oriented to sense the x, y, and z axis of a magnetic field. The bridge outputs are then converted to a 16-bit digital value using an internal delta-sigma A/D converter. A command set is provided (see Table 1) to configure the data sample rate, output format, averaging and zero offset. An on-board EEPROM stores any configuration changes for next time power-up. Other commands perform utility functions like baud rate, device ID and serial number. Also included in the HMR magnetometer is a digital filter with 50/60 Hz rejection to reduce ambient magnetic interference.

A unique switching technique is applied to the permalloy bridge to eliminate the effects of past magnetic history. This technique cancels out the bridge offset as well as any offset introduced by the electronics. The x, y, and z digitized data is sent out as a series of bytes, either after an ID match is received from the control processor, or as a continuous data stream. The data is serially output at either 9,600 or 19,200 baud, using the RS-232 or RS-485 standard, for serial input to most personal computers. The RS-485 standard allows connection of up to 32 devices on a single wire pair up to 4,000 feet in length. An HMR address can be stored in the on-board EEPROM to assign one of thirty-two unique ID codes to allow direct line access. An internal microcontroller handles the magnetic sensing, digital filtering, and all output communications eliminating the need for external trims and adjustments. Standard RS-485 or RS-232 drivers provide compliant electrical signalling.

Honeywell's magnetoresistive magnetometers provide an excellent means of measuring both linear and angular position and displacement. Low cost, high sensitivity, fast response, small size, and reliability are advantages over mechanical or other magnetometer alternatives. With an extremely low magnetic field sensitivity and a user configurable command set, these sensors solve a variety of problems in custom applications. The HMR2300 is available either as a circuit board with an optional 9-pin connector or in an aluminum enclosure with a 9-pin connector. Possible applications include compassing, remote vehicle monitoring, process control, laboratory instrumentation, anomaly detection, traffic and vehicle detection, and retail security systems.



1 Gauss (G) = 1 Oersted (in air), 1G = 79.58 A/m
1G = 10E-4 Tesla, 1G = 10E5 gamma
ppm - parts per million

OPERATING SPECIFICATIONS—Table 1

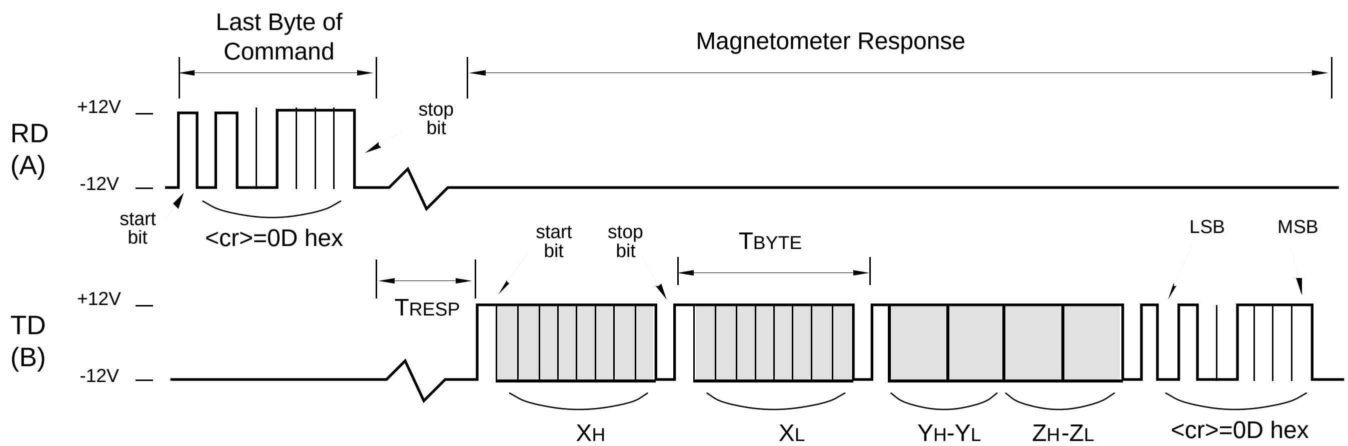
Characteristic	Conditions	Min	Typ	Max	Unit
Supply Voltage	Pin 9 referenced to pin 5	6.5		15	Volts
Supply Current	Vsupply=15V, with S/R=ON		27	35	mA
Operating Temperature	Ambient	-40		85	°C
Storage Temperature	Ambient, unbiased	-55		125	°C
Field Range	Full scale (FS) - total applied field	-2		+2	Gauss
Linearity Error	Best fit straight line (at 25°C) ±1 Gauss ±2 Gauss		0.1 1	0.5 2	%FS
Hysteresis Error	3 sweeps across ±2 Gauss @ 25°C		0.01	0.02	%FS
Repeatability Error	3 sweeps across ±2 Gauss @ 25°C		0.05	0.10	%FS
Gain Error	Applied field for zero reading		0.05	0.10	%FS
Offset Error	Applied field for zero reading		0.01	0.03	%FS
Accuracy	RSS of all errors (at 25°C) ±1 Gauss ±2 Gauss		0.12 1	0.52 2	%FS
Resolution	Applied field to change output	67			µGauss
Temperature Effect	Coefficient of gain Coefficient of offset (with S/R ON)		-600 ±114		ppm/°C
Power Supply Effect	From 6 -15V with 1G applied field		150		ppm/V
Vibration (operating)	5 to 10Hz for 2 hours 10Hz to 2kHz for 30 min.		10 2.0		mm g force
Max. Exposed Field	No perming effect on zero reading			10	Gauss
Weight	Board only In Aluminum Enclosure - extended - flush base		28 98 94		grams

TIMING SPECIFICATIONS—Table 2

Characteristic	Conditions	Min	Typ	Max	Unit
TRESP	Timing Diagrams (Figs. 1,2,3) *dd command (dd=Device ID) *ddP *ddRST *ddC *99 command (exceptions below) *ddQ *99Q	1.9	2 3 6 40 2 + (dd x 40) 2 + (dd x 80) 2 + (dd x 120)	2.2 3.2 6.2 60 2 + Typ 2 + Typ 2 + Typ	msec
TDELAY	Timing Diagram (Fig. 3) *dd command (dd=Device ID) *99 command	39	40 dd x 40	41 2 + Typ	msec
TBYTE	Timing Diagrams (Figs. 1,2) 9600 19200		1.04 0.52		msec
TSTARTUP	Power Applied to end of Start-Up message		50	80	msec

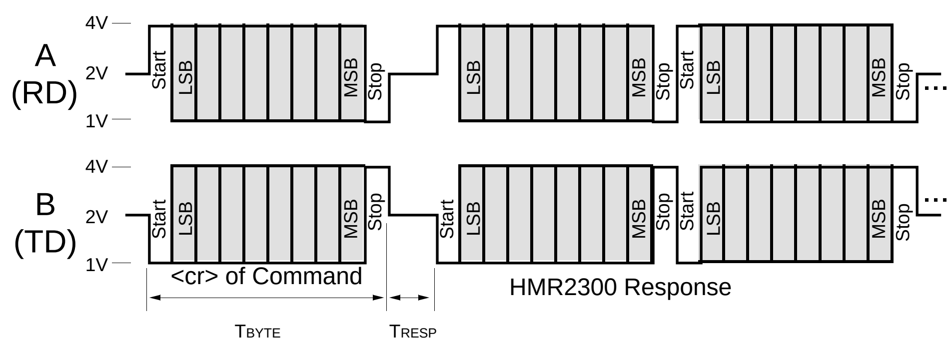
RS-232 COMMUNICATIONS—Figure 1

Timing is not to scale



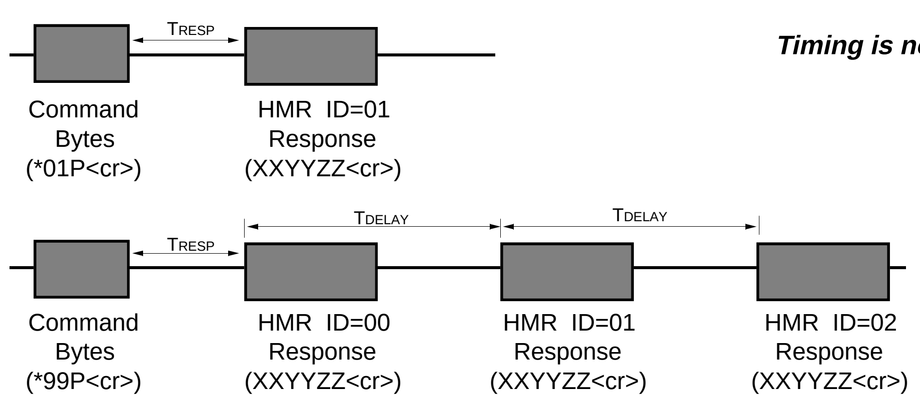
RS-485 COMMUNICATIONS—Figure 2

Timing is not to scale



GLOBAL ADDRESS (*99) DELAY—Figure 3

Timing is not to scale



COMMAND INPUTS—Table 3

A simple command set is used to communicate with the HMR. These commands can be typed in through a standard keyboard while running any communications software such as Terminal in Windows®.

Command	Inputs (1)	Response (2)	Bytes(3)	Description
Format	*ddWE *ddA *ddWE *ddB	ASCII_ON ¬ BINARY_ON ¬	9 10	ASCII - Output readings in BCD ASCII format. (default) Binary - Output readings in signed 16 bit binary format.
Output	*ddP *ddC ESC	{x, y, z reading} {x, y, z stream} {stream stops}	7 or 28 ... 0	P=Polled - Output a single sample. (default) C=Continuous - Output readings at sample rate. Escape key - Stop continuous readings.
Sample Rate	*ddWE *ddR=nnn	OK ¬	3	Set sample rate to nnn where: nnn= 10, 20, 25, 30, 40, 50, 60, 100, 123, or 154 samples/sec (default=20 sps)
Set/Reset Mode	*ddWE *ddTN *ddWE *ddTF *ddWE *ddT	S/R_ON ¬ S/R_OFF ¬ {Toggle}	7 8 7 or 8	S/R mode: TN -> ON=automatic S/R pulses (default) TF -> OFF=manual S/R pulses *ddT toggles command. (default=ON)
Set/Reset Pulse	*dd]S *dd]R *dd]	SET ¬ RST ¬ {Toggle}	4 4 4	] character - single S/R:]S -> SET=set pulse]R -> RST=reset pulse Toggle alternates between SET and RESET pulse.
Device ID	*99ID= *ddWE *ddID=nn	ID=_nn ¬ OK ¬	7 3	Read device ID (default ID=00) Set device ID where nn=00 to 98
Baud Rate	*99WE *99!BR=S *99WE *99!BR=F	OK ¬ BAUD=_9600 ¬ OK ¬ BAUD=_19,200 ¬	14 14	Set baud rate to 9600 bps. (default) Set baud rate to 19,200 bps. (8 bits, no parity, 1 stop bit)
Zero Reading	*ddWE *ddZN *ddWE *ddZF *ddWE *ddZR	ZERO_ON ¬ ZERO_OFF ¬ {Toggle}	8 9 8 or 9	Zero Reading will store and use current reading as a negative offset so that the output reads zero field. *ddZR toggles command. (default=OFF)
Average Readings	*ddWE *ddVN *ddWE *ddVF *ddWE *ddV	AVG_ON ¬ AVG_OFF ¬ {Toggle}	7 8 7 or 8	The average reading for the current sample X(N) is: Xavg = X(N)/2 + X(N-1)/4 + X(N-2)/8 + X(N-3)/16 + ... *ddV toggles command. (default=OFF)
Re-enter Response	*ddWE *ddY *ddWE *ddN	OK ¬ OK ¬	3 3	Turn the "Re-enter" error response ON (*ddY) or OFF (*ddN). OFF is recommended for RS-485 (default=ON)
Query Setup	*ddQ	{see Description}	62-72	Read setup parameters. default: ASCII, POLLED, S/R ON, ZERO OFF, AVG OFF, R ON, ID=00, 20 sps
Default Settings	*ddWE *ddD	OK ¬ BAUD=_9600 ¬	14	Change all command parameter settings to factory default values.
Restore Settings	*ddWE *ddRST	OK ¬ BAUD=_9600 *or* BAUD=_19,200	14 16	Change all command parameter settings to the last user stored values in the EEPROM.
Serial Number	*dd#	SER#_nnnn ¬	22	Output the HMR2300 serial number.
Software Version	*ddF	S/W_vers:_ nnnn ¬	27	Output the HMR2300 software version number.
Hardware Version	*ddH	H/W_vers:_ nnnn ¬	19	Output the HMR2300 hardware version number.
Write Enable	*ddWE	OK ¬	3	Activate a write enable. This is required before commands: Set Device ID, Baud Rate, and Store Parameters.
Store Parameters	*ddWE *ddSP	DONE ¬ OK ¬	8	This writes all parameter settings to EEPROM. These values will be automatically restored upon power-up.
Too Many Characters	Wrong Entry	Re-enter ¬	9	A command was not entered properly or 10 characters were typed after an asterisk (*) and before a <cr>.
Missing WE Entry	Write Enable Off	WE_OFF ¬	7	This error response indicates that this instruction requires a write enable command immediately before it.

(1) All inputs must be followed by a <cr> carriage return, or Enter, key. Either upper or lower case letters may be used. The device ID (dd) is a decimal number between 00 and 99. Device ID=99 is a global address for all units.

(2) The "¬" symbol is a carriage return (hex 0D). The "_" symbol is a space (hex 20). The output response will be delayed from the end of the carriage return of the input string by 2 msec (typ.), unless the command was sent as a global device ID=99 (see TDELAY).

Sample Rate (sps)	ASCII		Binary		f_{3dB} (Hz)	Notch (Hz)	Command Input Rate - min. (msec)
	9600	19200	9600	19200			
10	*yes*	*yes*	*yes*	*yes*	17	50/60	20
20					17	50/60	20
25					21	63/75	16
30					26	75/90	14
40					34	100/120	10
50					42	125/150	8
60					51	150/180	7
100	DATA				85	250/300	4
123	INVALID				104	308/369	3.5
154					131	385/462	3

Parameter Selections verses Output Sample Rate—Table 4

DATA FORMATS

The HMR2300 transmits each x, y, and z axis as a 16-bit value. The output data format can either be 16-bit signed binary (sign + 15-bits) or binary coded decimal (BCD) ASCII characters. The command *ddA will select the ASCII format and *ddB will select the binary format.

The order of output for the binary format is: Xhi, Xlo, Yhi, Ylo, Zhi, Zlo. The binary format is more efficient for a computer to interpret since only 7 bytes are transmitted. The BCD ASCII format is easiest for user interpretation but requires 28 bytes per reading. There are limitations on the sample rate based on the format and baud rate selected (see Table 2). Examples of both binary and BCD ASCII outputs are shown below for field values between ±2 Gauss.

Field (Gauss)	BCD ASCII Value	Binary Value (Hex) High Byte	Binary Value (Hex) Low Byte
+2.0	30,000	75	30
+1.5	22,500	57	E4
+1.0	15,000	3A	98
+0.5	7,500	1D	4C
0.0	00	00	00
-0.5	- 7,500	E2	B4
-1.0	-15,000	C3	74
-1.5	-22,500	A8	1C
-2.0	-30,000	8A	D0

Output Readings—Table 5

Binary Format: 7 bytes

X_H | X_L | Y_H | Y_L | Z_H | Z_L | <cr>

X_H = signed high byte, x axis
X_L = low byte, x axis
<cr> = carriage return (Enter Key), Hex code = 0D

The binary characters will be unrecognizable on a monitor and will appear as strange symbols. This format is best when a computer is interpreting the readings.

ASCII Format: 28 bytes

SN | X_1 | X_2 | CM | X_3 | X_4 | X_5 | SP | SP | SN | Y_1 | Y_2 | CM | Y_3 | Y_4 | Y_5 | SP | SP | SN | Z_1 | Z_2 | CM | Z_3 | Z_4 | Z_5 | SP | SP | <cr>

The ASCII characters will be readable on a monitor as signed decimal numbers. This format is best when the user is interpreting the readings.

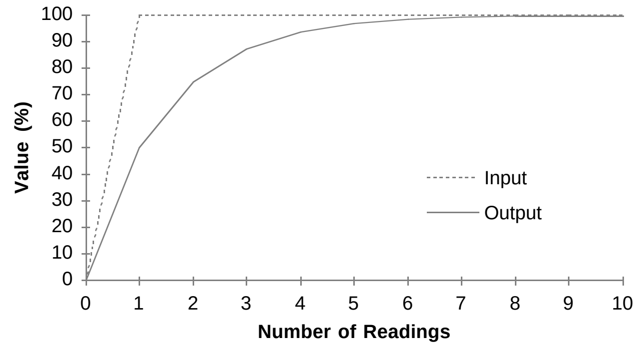
<cr> = carriage return (Enter Key), Hex code = 0D
SP = space, Hex code = 20
SN (sign) = - if negative, Hex code = 2D
SP if positive, Hex code = 20
CM (comma) = , if leading digits are not zero, Hex code = 2C
SP if leading digits are zero, Hex code = 20
X_1, X_2, X_3, X_4, X_5 = Decimal equivalent ASCII digit
X_1, X_2, X_3 = SP if leading digits are zero, Hex code = 20

SET/RESET and AVERAGE COMMAND

The set-reset function generates a nominal 3.5 Amp pulse, or ≈80 Oe field, to each sensor to realign the permalloy magnetization. This yields the maximum output sensitivity for magnetic sensing. This current pulse is generated inside the HMR2300 and consumes less than 1 mA typically. The Set/Reset Mode command (*ddT, or *ddTN, for S/R=ON) activates an internal switching circuit that flips the current in a 'Set' and 'Reset' condition. This cancels out any temperature drift effects and ensures the sensors are operating in their most sensitive region. Fluctuations in the magnetic readings can be reduced by using the Average Readings command: *ddV, or *ddVN, for AVG=ON. This command provides a low pass filter effect on the output readings that reduces noise due to S/R switching and environmental magnetic effects. See the figures at right for the impulse and step response of the Average Readings function.

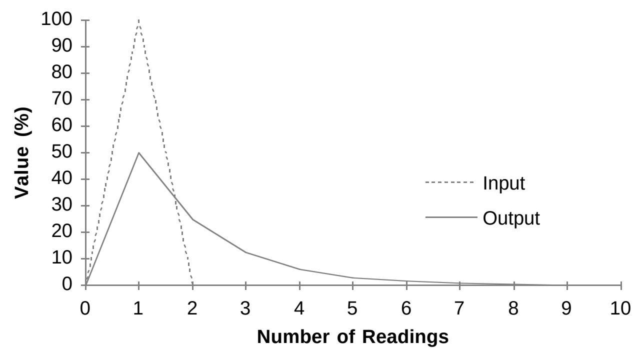
Switching the set-reset state is not required to sense magnetic fields. A single Set (or Reset) pulse will maximize the output sensitivity and it will stay that way for months or years. To turn off the internal switching, enter the command: *ddT, or *ddTF, for S/R=OFF. In this state the sensors are either in a Set or Reset mode. If the hybrid is exposed to a large magnetic field (>20 Gauss), then another set pulse is required to maximize output sensitivity.

In the Set mode, the direction of sensitive axis are shown on the package label and Board Dimensions drawing. In the Reset mode, the sensitive field directions are opposite to those shown. By typing '*dd]' the user can manually activate a Set, or Reset, pulse. The S/R commands (*dd], *ddS, *ddR) can be used during the continuous read mode to flip between a Set and Reset state. Note that the first three readings immediately after these command will be invalid due to the uncertainty of the current pulse to the sensor sample time.



Average Reading Response to an Input Step

Figure 4



Average Reading Response to an Impulse Input

Figure 5

DEVICE ID

The Device ID command (*99ID=nn) will change the HMR2300 ID number. A Write Enable (*ddWE) command is required before the device ID can be changed. This is required for RS-485 operation when more than one HMR2300 is on a network. A Device ID=99 is universal and will simultaneously talk to all units on a network.

ZERO READING COMMAND

The Zero Reading command (*ddZR) will take a magnetic reading and store it in the microprocessor. This value will be subtracted from subsequent readings as an offset. The Zero Reading will be terminated with another command input (*ddZR) or a power down condition. This is useful for setting a reference heading direction or nulling the earth's field before anomaly detection.

BAUD RATE COMMAND

The Baud Rate command (*dd!BR=F or S) will change the HMR2300 baud rate to either Fast (19,200 baud) or Slow (9600 baud). A Write Enable (*ddWE) command is required before the baud rate can be changed. The last response after this command has been accepted will be either BAUD=9600 or BAUD=19,200. This will indicate to the user, or computer, to change to the identified new baud rate before communications can resume.

DEFAULT and RESTORE SETTINGS

The Default Settings command (*ddD) will force the HMR2300 to all the default parameters. This will not be a permanent change unless a Store Parameter command is issued. The Restore Settings command (*ddRST) will force the HMR2300 to all the stored parameters in the EEPROM.

OUTPUT SAMPLE RATES

The sample rate can be varied from 10 samples per second (sps) to 154 sps using the R= command. Each sample contains an x , y, and z reading and can be outputted in either 16-bit signed binary or binary coded decimal (BCD) ASCII. The ASCII format shows the standard numeric characters displayed on computers. Some sample rates may have restrictions on the format and baud rate used due to transmission time constraints.

There are 7 bytes transmitted for every reading in binary format and 28 bytes per reading in ASCII format. Transmission times for 9600 baud are about 1 msec/byte and for 19,200 baud are about 0.5 msec/byte. The combinations of format and baud selections verses the sample rates are shown in Table 2. The default setting of ASCII format and 9600 baud will only transmit correctly up to 30 sps.

Note: The HMR2300 will output at higher data rate settings but the readings may be incorrect and will be at a lower output rate than selected.

For the higher sample rates (>60 sps), it is advised that computer settings for the Terminal Preferences be set so a line feed (LF) is NOT added to the inbound data. This slows down the reception of data and it will not be able to keep up with the incoming data stream.

INPUT SIGNAL ATTENUATION

Magnetic signal being measured will be attenuated based on the sample rate selected. The bandwidth, defined by the 3dB point, is shown in Table 2 for each sample rate. The default rate of 20 sps has a bandwidth of 17 Hz. The digital filter inside the HMR2300 is the combination of a comb filter and a low pass filter. This provides a linear phase response with a transfer function that has zeros in it.

When the 10 or 20 sps rate is used, the zeros are at the line frequencies of 50 and 60 Hz. These zeros provide better than 125 dB rejection. All multiples of the zeros extend throughout the transfer function. For example, the 10 and 20 sps rate has zeros at 50, 60, 100, 120, 150, 180, ... Hz. The multiples of the zeros apply to all the sample rates against the stated notch frequencies in Table 2.

COMMAND INPUT RATE

The HMR2300 limits how fast the command bytes can be received based on the sample rate selected. Table 2 shows the minimum time between command bytes for the HMR2300 to correctly read them. This is usually not a problem when the user is typing the commands on a keyboard. The problem could arise from a computer program outputting command bytes (characters) too quickly.

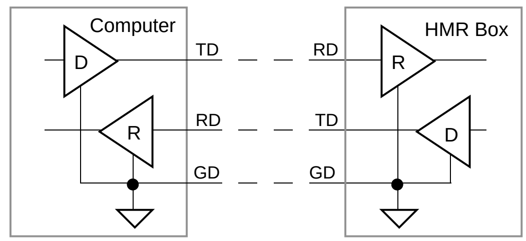
DATA COMMUNICATIONS

The RS-232 signals are single-ended unidirectional levels being sent and received simultaneously (full duplex). One signal is from the PC (Tx) to the HMR (Rx) and the other is from the HMR (Tx) to the PC (Rx). When a logic one is being transmitted, either the Tx or Rx line will drive ≈-7 volts referenced to ground. For a logic zero, the Tx or Rx line will drive ≈-7 volts below ground. Since the signals being transmitted are dependent on an absolute voltage level, this limits the distance of transmission due to line noise and signal loss—typically 60 feet.

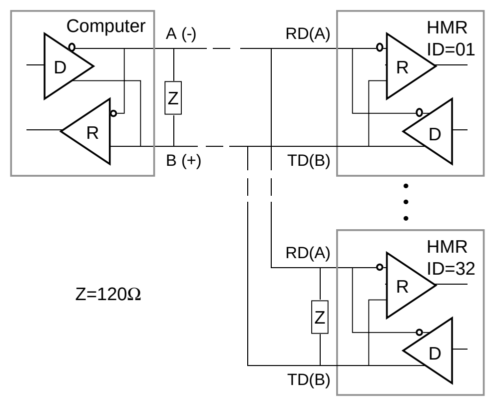
When using RS-485, the signals are balanced differential levels. That is, when a logic one is being transmitted, the Tx line will drive ≈1.5 volts higher than the Rx line. For a logic zero, the Tx line will drive ≈1.5 volts lower than the Rx line. The signals being transmitted are not dependent on an absolute voltage level on either Tx or Rx but rather a difference voltage. This allows signals to be transmitted in a high noise environment or over very long distances where line loss may otherwise be a problem—typically 4,000 feet. Note that the RS-485 line must be terminated at both ends with a 120Ω resistor.

Note: When the HMR2300 is in a continuous read mode on the RS-485 bus, it may be necessary to enter several escape keys to stop the readings. If the computer taking the readings can detect a carriage return code and send the escape code immediately after it, then a systematic stop readings will occur. If an operator is trying to stop readings using the keyboard, then several (if not many) escape key entries must be given, because the RS-485 lines share the same wires for transmit and receive. If an escape key is entered during the time data is sent from the HMR2300, then the two will produce an erroneous character that will not stop the data stream. Only when the escape key is pressed during the time the HMR2300 is not transmitting will the data stream stop.

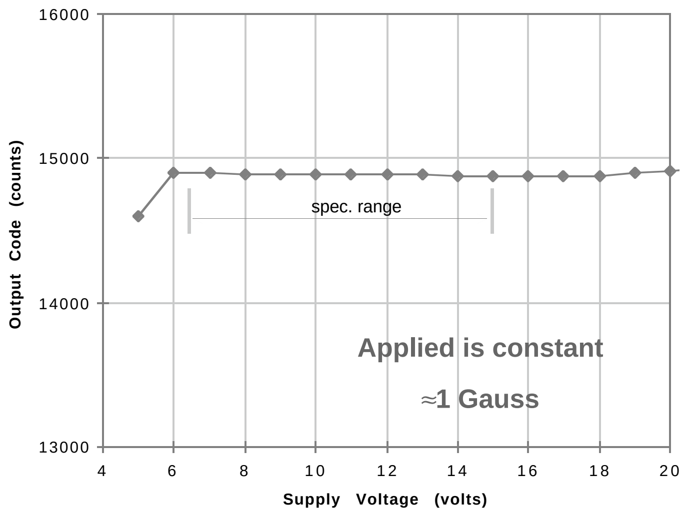
There is a safety feature built into the HMR2300 to allow multiple units to respond to a global address command on the RS-485 bus. When a command is sent with the global address (*99P), each HMR2300 will automatically delay their response by its device ID number times TDELAY. This is for operation on the RS-485 bus to allow each response to follow each other so two units do not transmit at the same time. This delay also applies to single units operating on the RS-232 bus.



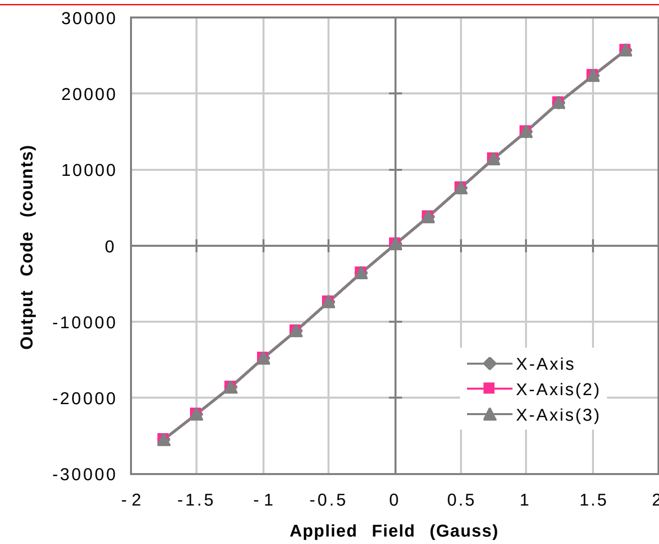
RS-232 Unbalanced (Single-Ended)—Figure 6



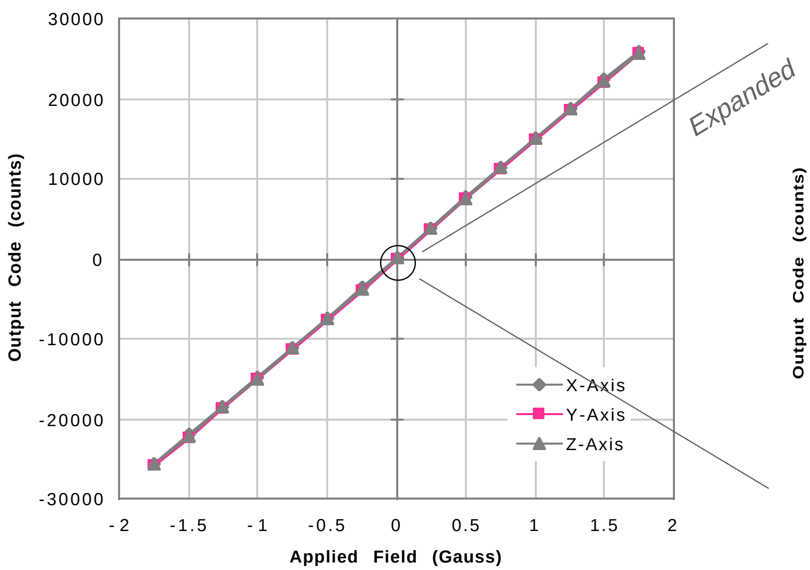
RS-485 Balanced Multiport—Figure 7



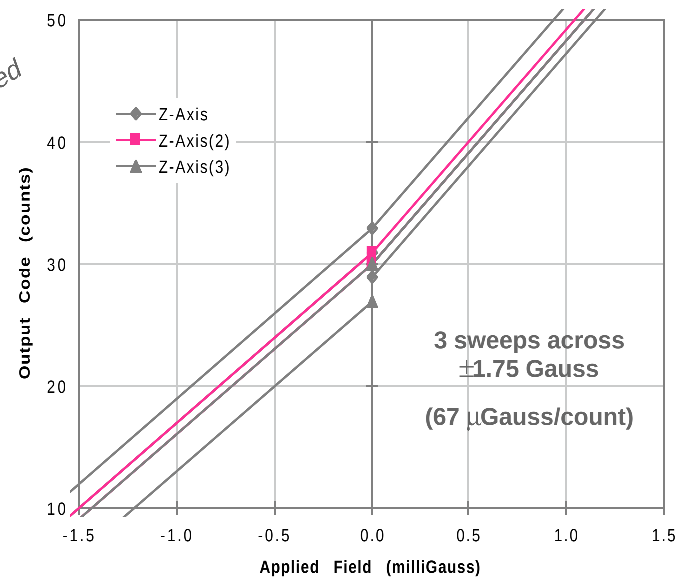
Supply Voltage Effect on Readings—Figure 8



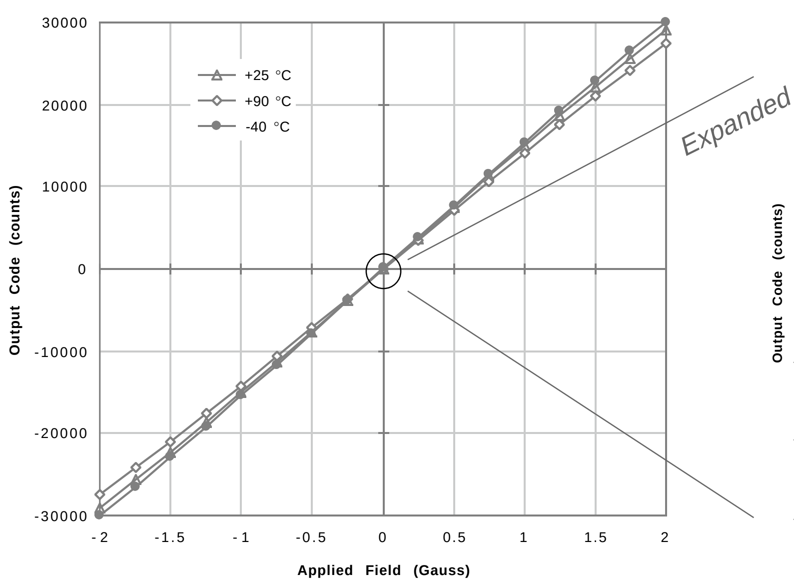
Three Sweeps of X-Axis Transfer Curve—Figure 9



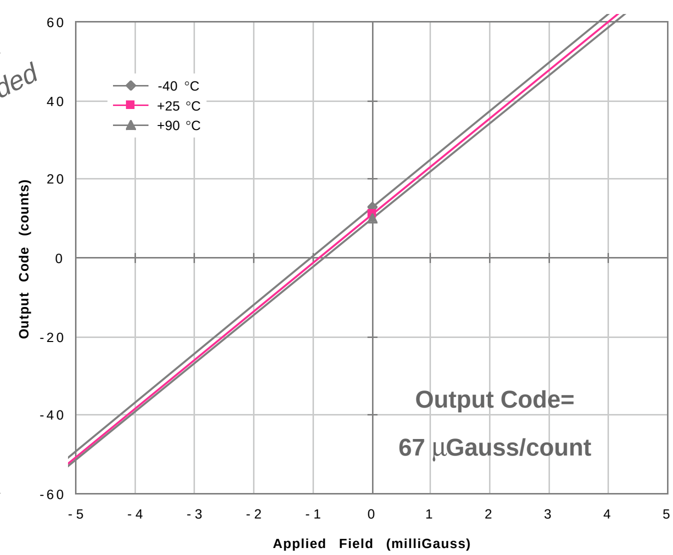
Transfer Curves for X, Y, and Z Axis—Figure 10



Magnified Z-Axis Showing Hysteresis—Figure 11

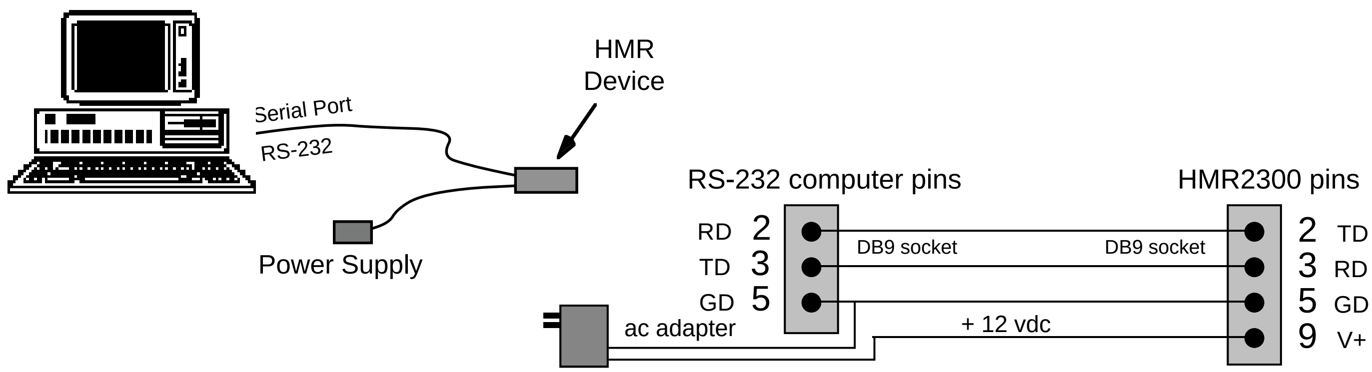


Single-Axis Temperature Effects—Figure 12

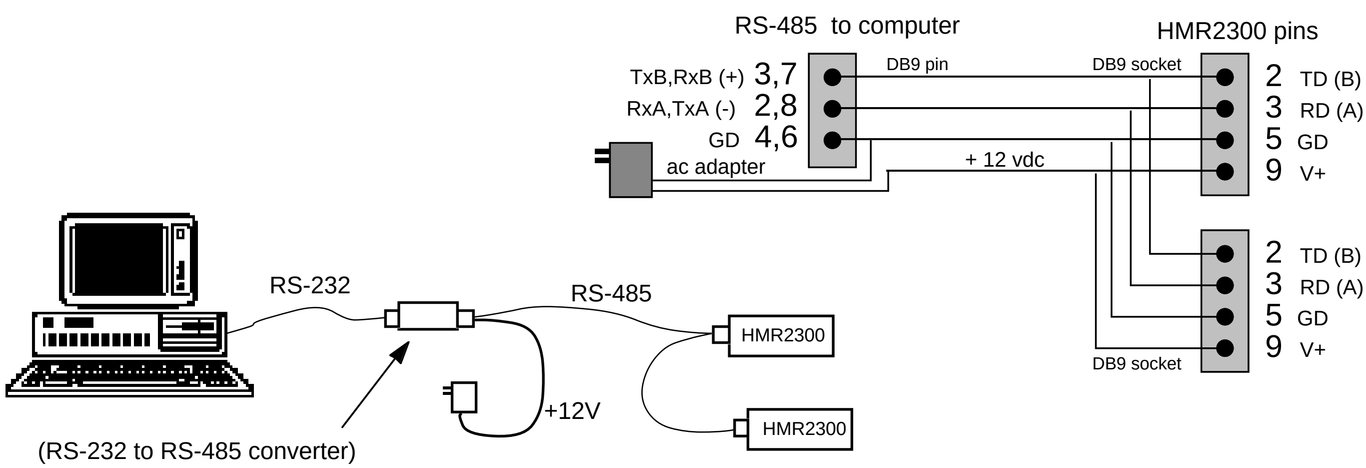


Magnified-Axis Showing Temp. Effects—Figure 13

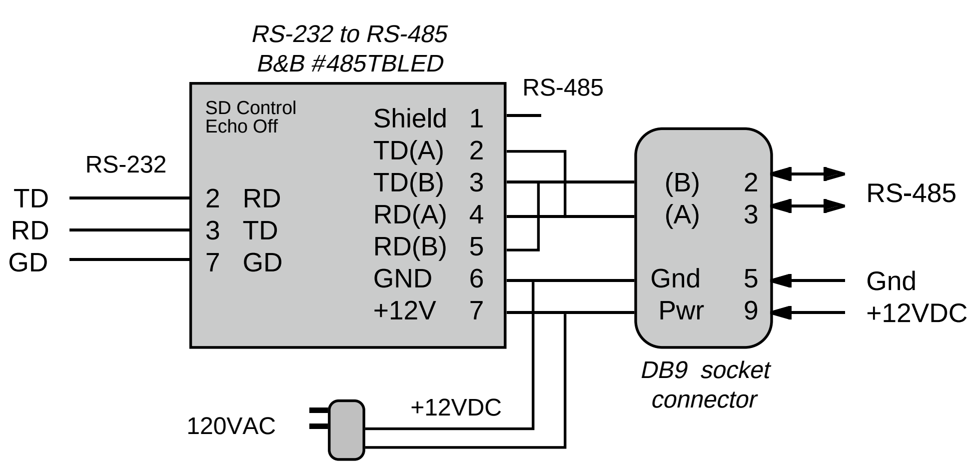
COMPUTER RS-232 TO HMR2300 CONNECTION— Figure 14



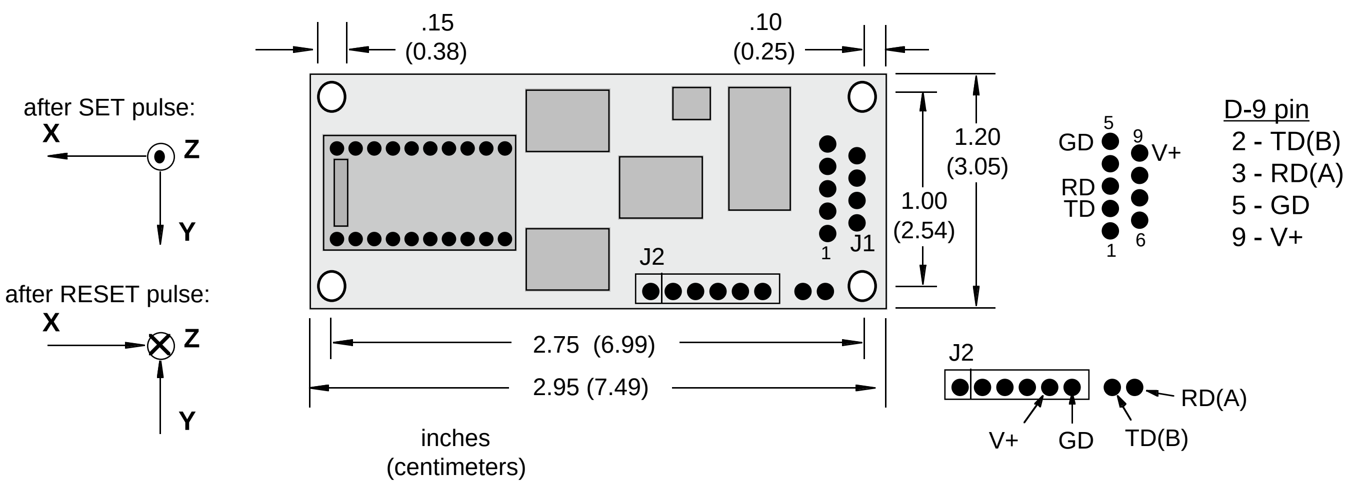
COMPUTER RS-485 TO HMR2300 CONNECTION—Figure 15



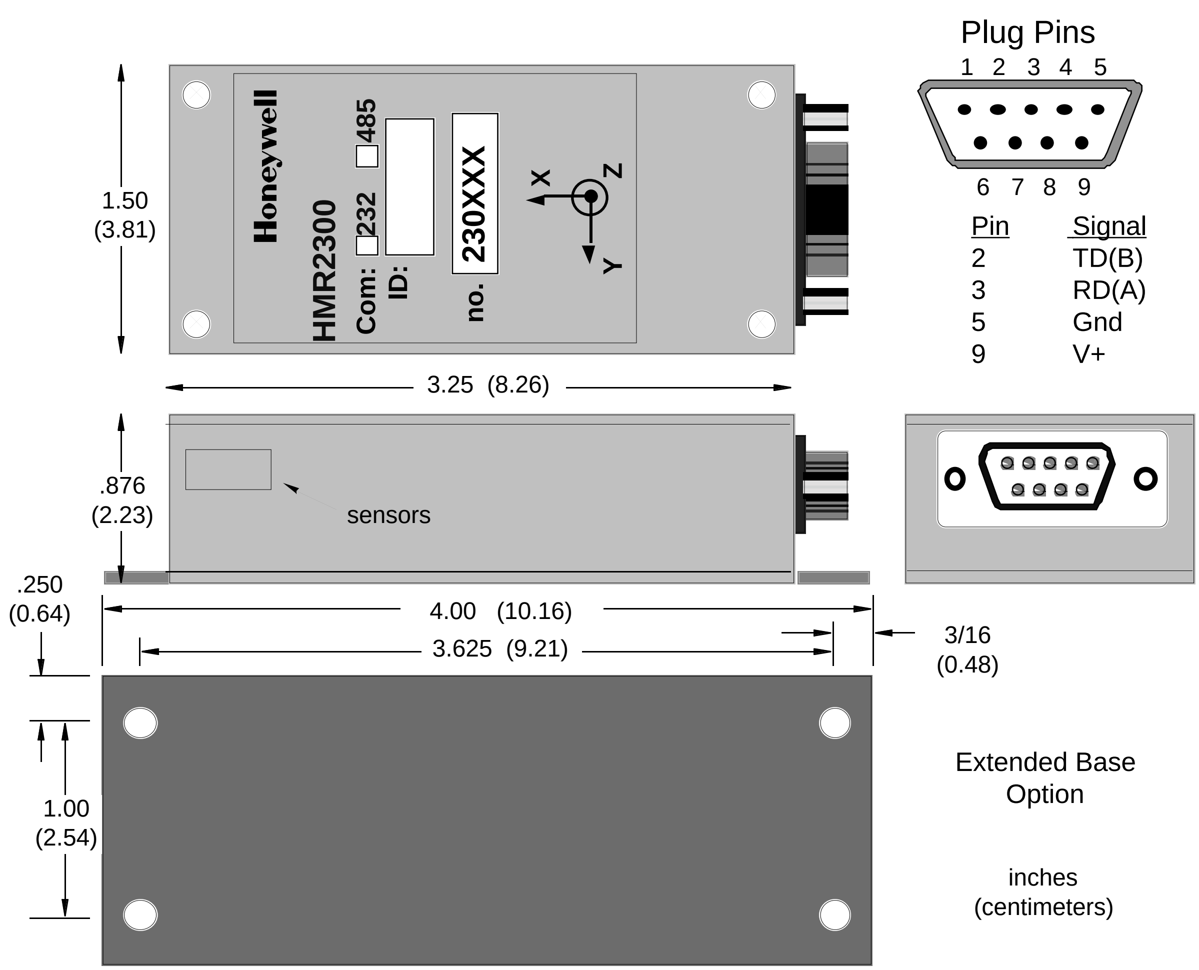
RS-232 TO RS-485 CONVERTER—Figure 16



BOARD DIMENSIONS AND PINOUT—Figure 17



CASE DIMENSIONS AND PINOUT—Figure 18



APPLICATIONS PRECAUTIONS

Several precautions should be observed when using magnetometers in general:

• The presence of ferrous materials—such as nickel, iron, steel, cobalt—near the magnetometer will create disturbances in the earth's magnetic field that will distort x, y, z field measurements.

• The presence of the earth's magnetic field must be taken into account when measuring other x, y, z, fields.

• The variance of the earth's magnetic field must be accounted for in different parts of the world. Differences in the earth's magnetic field are quite dramatic between North America, South America and the Equator region.

• Perming effects on the HMR box need to be taken into account. If the HMR box is exposed to fields greater than 10 Gauss (or 10 Oersted), then the box must be degaussed. The result of perming is a high zero field output code that exceeds specification limits. Degaussing magnets are readily available from local electronics outlets and are inexpensive. If the HMR box is not degaussed, severe zero field offset values will result.

HANDLING PRECAUTIONS

The HMR Magnetometer measures fields within 2 Gauss in magnitude with a 0.1 milliGauss resolution. Computer floppy disks (3.5" diskettes) store data with fields strengths approximately 10 Gauss. This says that the HMR Magnetometer is at least ten times more sensitive to stray magnetic fields than common floppy disks! Therefore, treat the HMR Magnetometer at least as good as your computer disks by keeping them away from motors, monitors and magnets. Even though they don't erase data like a floppy disk would, they will pick up, and retain, these magnetic fields that may interfere with x, y, z axis measurements.

NON-FERROUS MATERIALS

Materials that do not affect surrounding magnetic fields are: copper, brass, gold, aluminum, some stainless steel, silver, tin, silicon and any nonmetallic material.

ORDERING INFORMATION

Customer Service Representative
(612) 954-2888 fax: (612) 954-2582
E-Mail: clr@mn14.ssec.honeywell.com

ORDERING INFORMATION

Example: HMR2300-D00-232

HMR	2	3	00-	D	00-	232
Honeywell Magnetoresistive	Family	Axis	Type	Connector	Box	COM Std.

Family	2—16-bit Delta Sigma Data Conversion
Axis	3—Three-Axis
Type	00— ±2Gauss, -40 to 85°C
Connector	N—No connector D—Nine-pin D-Sub Connector
Box	00—No Box, Board Only 20—Aluminum 3.25"x1.50"x1.13", flush base 21—Aluminum 4.00"x1.50"x1.13", extended base
COM Std.	232—RS-232 Communication Standard 485—RS-485 Communication Standard

Honeywell

Helping You Control Your World

900139 Rev. G
6-97