
Development of CubeSat: Attitude Determination System.

MEng in Mechatronics Engineering

By

Josef Da Silva Bernardo Neto



University of Glasgow

ENG5041P: Individual Project 5
2169252N

First Supervisor: Dr. Matteo Ceriotti
Second Supervisor: Dr. Kevin Worrall

A report submitted to the University of Glasgow in
accordance with the requirements of the degree of
MASTER OF ENGINEERING in the School of Engineering.

January 11, 2019
Scotland, Glasgow

Abstract

With the advancements in technology, CubeSats have been a catalyst in shaping the field of space technology and the way in which satellites are viewed. Hence, their applications are innumerable, from providing students and researchers with knowledge on development of satellites, to also being used in the military sector. Therefore, the purpose of this report is to show the development of a CubeSat, the focal point will be to present the Attitude Determination System of a CubeSat (ADS). This project uses sun sensors based on less expensive photodiodes and an inertial measurement unit sensor to measure the orientation of the CubeSat. The photodiode used was the OPT101 Monolithic photodiode. Each photodiode integrates a transimpedance amplifier with an operating voltage of 2.7 to 36V and quiescent current of 120 μ A.

This project employs three sun sensors at three different sides of the CubeSat. Each side represents a single axis (x, y, z) and each individual axis will produce body angles susceptible to the incident light falling at the sensitive area of the sun sensor and therefore the sun vectors can be obtained. To perform the experiment, sun simulator flashlight of 3000W was used. The sun simulator was maintained at the fixed location while the CubeSat was moved around the sun simulator. The body angle acquired when the sun sensor points directly to the source light is 90°. The sun sensor captured the ambient light which introduces roughly 12°.

The inertial measurement unit sensors are common device to measure the orientation of a satellite with respects to a reference frame. The roll, pitch and heading angles were calculated using Euler algorithm. In addition, a Kalman filter was used to reduce the unsatisfactory noise into the system. The experiment was well developed and has shown promising results which introduces a new research area for future students at the University of Glasgow. This project compiles hardware and software development.

Acknowledgement

I would like to thank God I am alive. I would also like to express my deep gratitude to Dr. Matteo Ceriotti and Dr. Kevin Worrall for supervising me throughout this project. I would also like to thank Mr. Andrew Philip, Mr. Tom O'Hara and Mr. Alan Yuill for helping me answering some technical questions related to the project. My gratitude for Sonangol Ep. for providing me a scholarship to study in UK and experience higher-level education. I would like to thank Miss Andrea Amado and Mr. Treasure Nkawu for proof reading my work. I am grateful for my family and friends support.

To my dear parents

Ama-vos

Table of Content

	Pages
Abstract	i
Acknowledgement	ii
List of figures	vii
List of tables	viii
Acronyms	ix
Mathematic Expression and Symbols	x
Chapter 1	
1. Section 1: Introduction and Background Theory	1
1.0. Introduction	1
1.1. Project Scope	2
1.2. Motivation	2
1.3. Background	3
1.3.1. Pumpkin CubeSat	3
1.3.2. Modular Chassis system	3
1.3.3. Motherboard	4
1.3.4. Development Board	4
Chapter 2	
2. Section 2: Literature Review	5
2.0. Attitude determination system	5
2.1. Coordinate system	5
2.1.1. Inertial reference frame	5
2.1.2. Body reference frame	5
2.1.3. Orbit reference frame	6
2.2. ADS Sensors	6
2.2.1. Sun sensor	7
2.2.2. Inertial Measurement Unit Sensor (IMU)	14
2.2.2.1. Gyroscope	14
2.2.2.2. Accelerometer	15
2.2.2.3. Magnetometer	16
2.3. Rotation Matrix	17

2.4.	Euler Angles	18
2.5.	Kalman filter	20
2.6.	Conclusion	23

Chapter 3

3.	Section 3 : Hardware and Software	24
3.0.	ADS Hardware	24
3.1.	Power Management	28
3.2.	Software programming	29
3.3.	MSP430 Analogue to digital Converter	29
3.4.	Telemetry	31
3.5.	Protocols of communication	32
3.5.1.	Inter-integrated circuit (I2C) protocol	32
3.5.2.	One-wire protocol	34
3.6.	Sun sensor hardware	36
3.6.1.	Software Flow diagram	38
3.7.	Inertial Measurement unit (IMU) Hardware	40
3.7.1.	Software diagram flow	42
3.8.	IIR Digital Control	43
3.9.	Conclusion	44

Chapter 4

4.	Section 4: Methodology and Results	45
4.0.	Experiment description	45
4.1.	Sun Sensor -Results and Discussion	46
4.2.	ADC 12 bit to Voltage	48
4.2.1.	Methodology	48
4.2.2.	Results and Discussion	48
4.3.	Body Angles	50
4.3.1.	Methodology	50
4.3.2.	Results and discussion	50
4.4.	IMU sensor	52
4.4.1.	Methodology	52
4.4.2.	Results and Discussion	52
4.5.	Euler Angles - Result and Discussion	54

4.6.	Kalman filter- Results and Discussion	55
4.7.	Conclusion	57
 Chapter 5		
5.	Section 5: Overall Conclusion and Future work	58
5.0.	Overall Conclusion and Future work	58
5.1.	Future work	58
References		59
Appendix 1		63
Appendix 2		66

List of Figures

Figure 1: 3D FRAME OF THE CUBESAT	8
Figure 2: SUN SENSOR DETECTING INCIDENT ANGLES	10
Figure 3: SINGLE SLIT (POST, ET AL., 2013)	11
Figure 4: CONFIGURATION SUN SENSOR (Post, et al., 2013)	11
Figure 5: TWO SENSORS AT THE SAME PLANE (Wertz, 1990)	12
Figure 6: MAGNETIC NORTH POLE	16
Figure 7: FRAME OF ORIENTATION	17
Figure 8: : EULER ANGLES AT REFERENCE FRAME	18
Figure 9: KALMAN FILTER (Aliabadi & Sharbafi, 2010)	20
Figure 10: BLOCK DIAGRAM OF THE ENTIRE SYSTEM	24
Figure 11: MAIN BOARD PCB DESIGN	26
Figure 12: MAIN BOARD SCHEMATIC	27
Figure 13: Power management diagram	28
Figure 14: ADC BLOCK DIAGRAM SETUP IN GRACE	29
Figure 15: EXTENDED ADC BLOCK DIAGRAM IN GRACE	30
Figure 16: BLUETOOTH DEVICE	31
Figure 17: UART SETUP	31
Figure 18: I2C INTERFACE BETWEEN BOARDS	32
Figure 19: DATA FRAME	33
Figure 20: I2C BUS	33
Figure 21: ONE-WIRE CONNECTION	34
Figure 22: PRE-DEFINED SIGNAL	35
Figure 23: SUN SENSOR SCHEMATIC	36
Figure 24: RELATIVE RESPONSE AGAINST INCIDENT ANGLE (Texas Instruments, 2003)	37
Figure 25: SOFTWARE DIAGRAM FOR THE SUN SENSOR	38
Figure 26: PMOD NAV IMU SENSOR	40
Figure 27: BLOCK DIAGRAM FOR THE IMU SUBSYSTEM	41
Figure 28: SOFTWARE DIAGRAM FLOW FOR IMU SENSOR PROGRAMMING	42
Figure 29: LOWPASS FILTER	43
Figure 30: EXPERIMENT APPARATUS SETUP	45
Figure 31: SUN SENSOR AT THE X-AXIS	47
Figure 32: SUN SENSOR AT THE Y-AXIS	47
Figure 33: SUN SENSOR AT THE Z-AXIS	47
Figure 34: VOLTAGE OF THE SUN SENSOR AT THE X-AXIS	49
Figure 35: VOLTAGE OF THE SUN SENSOR AT THE Y-AXIS	49
Figure 36: VOLTAGE OF THE SUN SENSOR AT THE Z-AXIS	49
Figure 37: BODY ANGLES OF THE CUBESAT AS IT ROTATES AROUND THE SUN SIMULATOR	51
Figure 38: GYROSCOPE AT X-AXIS	52
Figure 39 : ACCELERATION AT Y-AXIS	52
Figure 40: ACCELERATION AT X-AXIS	52
Figure 41: ACCELERATION AT Z-AXIS	52

Figure 42: GYROSCOPE Z-AXIS	53
Figure 43:GYROSCOPE Y-AXIS	53
Figure 44: MAGNETOMETER Y-AXIS	53
Figure 45: MAGNETOMETER Z-AXIS	53
Figure 46: MAGNETOMETER X-AXIS.....	53
Figure 47: EULER ANGLES	55
Figure 48: ROLL ANGLES.....	56
Figure 49: PITCH ANGLES	56
Figure 50: HEADING ANGLES	57
Figure 51:MOTHERBOARD	63
Figure 52:MODULAR CHASSIS STRUCTURE	63
Figure 53:DEVELOPMENT BOARD	63
Figure 54: HARDWARE OF ENTIRE SYSTEM.....	64
Figure 55: SUN SENSOR WITH TEMPERATURE SENSOR	64
Figure 56: SUN SENSOR FIXED COORDINATES	65
Figure 57: BOARD WITH IMU, IMU, AND PPM CONNECTORS AT LEFT	65

List of Tables

Table 1: Sensitivity ranges.....	40
Table 2: Body angles and sun vectors	51

List of Acronym

UCS- Union of Concerned Scientists.

UNOOSA - United Nations Office for Outer Space Affairs.

NASA – National aeronautics and space administration.

PPM – Pluggable processor modules.

C&DH – Command and Data Handling.

TT&C - Telemetry, Tracking and Command.

CSK BUS – CubeSat kit Bus.

DB – Development Board.

ADC – Analogue to Digital Conversion.

ADC12 – 12-bit Analogue to Digital Conversion.

ADS – Attitude Determination System.

ADCS - Attitude Determination and control System.

FOV – Field of view

CCD – charge coupled device

CMOS- Complementary metal – oxide semiconductor.

CDs – CubeSat Design Specification

GPIO – General purpose input and output

DOF – Degree of Freedom

DB-MSP430 – Development Board (MSP430)

MSE – Mean Square Error

Mathematic Expression and Symbols

1. i, j, k : Cartesian Basic vector.
2. n, n_i, n_j, n_k : Normal vector, normal vector at the i vector, normal vector at j and k axis.
3. I_i, I_{max} : Current at i cartesian vector, maximum current.
4. S : Sun vector or Solar vector.
5. $\theta, \alpha_1, \alpha_2$: Body angles.
6. $\Delta I, I_1, I_2$: Difference in Current, current at plane 1, current at plane 2.
7. V_{max} : Maximum voltage
8. $\tilde{V}_i, \tilde{V}_j, \tilde{V}_k$: Vectors voltage.
9. ω_{ib}^b : The angular velocity at inertial body frame.
10. R^{bn} : The rotation matrix from the moving frame to the body frame.
11. ω_{ie}^n : The angular velocity of the earth frame earth in relation to the inertial frame
12. ω_{en}^n : The angular velocity of the earth
13. ω_{nb}^b : The transport rates
14. a_x, a_y, a_z : Acceleration at x-axis, acceleration at y-axis, acceleration at z-axis
15. a_{ii}^n : The linear acceleration of the sensor
16. a_{nn}^n : Acceleration for the navigation
17. $2\omega_{ie}^n \times \vartheta_n^n$: Coriolis acceleration.
18. $\omega_{ie}^n \times \omega_{ie}^n \times p^n$: Centrifugal acceleration.
19. ψ, θ, ϕ : Heading, Pitch, Roll angles.
20. R_x, R_y, R_z : Rotation matrices.
21. \mathcal{X}_k : State vector
22. B : Control-input model
23. \mathcal{U}_k : Control input
24. \mathcal{W}_k : Process noise
25. $\begin{bmatrix} \theta \\ \theta_b \end{bmatrix}$: Angle and bias dependent on the measurement from gyroscope and accelerometer.
26. Δt : Unbiased rate times
27. P_k : Error covariance matrix
28. F : State transition model
29. Q_k : Covariance matrix at current time
30. $\tilde{\mathcal{Y}}_k$: Innovation

31. z_k : The observation
32. z_k : The observation
33. H : The observation model to map the a priori state into the observed space
34. S_k : the innovation covariance
35. $P_{k|k-1}$: Priori error covariance matrix
36. R : The Measurement covariance matrix
37. $var(v)$: Measurement noise variance
38. K_k : Kalman gain at time k
39. $\hat{x}_{k|k}$: Posteriori state
40. $\hat{x}_{k|k-1}$: Priori state
41. $P_{k|k}$: Posteriori error covariance matrix
42. I : Identity matrix
43. Q_θ : The accelerometer variance
44. $Q_{\hat{\theta}}$ the variance of the bias

CHAPTER 1

Section 1: Introduction and Background Theory

1. Introduction

The purpose of this report is to present the design and result on the Attitude determination system of a CubeSat. The report examines different methods of performing the orientation of the CubeSat and introduces a the most reliable and cost-effective technique. This report contains 5 chapters. The first chapter aims to provide a theoretical background related to the project. The second chapter examines approaches from different literatures. The third chapter discuss the hardware and software integrated in this project. Fourth chapter shows the methodology used to perform this experiment among with the results attained and discussed. The last section details the conclusion of the project. The report will also recommend future works and critical areas of the systems, that would require further calibration for an improved performance of the system.

Satellite is space-based machinery, designed and launched into orbit that revolves around a planet or the sun for various technological and space research purposes (Howell, 2017). Satellite application meets a broad range of industries, education and geographies (Pelton, et al., 2017) . For Instance, communication satellites are used in mobile wireless or fixed wireless communications to transmit and receive radio signals from an orbiting satellite to a ground station or another terrestrial position (Elbert, 2004). These satellites endorse high speed communication and provide excellent connectivity (Luka, et al., 2011) . For example, satellites provide facilitation in appraising an exact location or position desired (Jin, 2015). There are substantial benefits associated with satellites orbiting higher than the earth's atmosphere. A peculiar benefit can be the collection of data. Data is collected at a faster rate than instruments on the ground and they can detect the earths' surface better than telescopes (May, 2017). In spite of the great advantages of large satellites, there are some costs attained within their production. Large satellites could be unreliable and extremely expensive to be manufactured in a complete model, and for them to be launched into space. Large Satellites are also difficult to maintain (Sweeting, 2018).

"Small satellites are a state of mind rather than defined simply by physical parameters"

(Sweeting, 2018)

To overcome the disadvantages of larger satellites, small satellites such as CubeSats have been developed. In 1999, Professor Bob Twiggs from Stanford university and Professor Jordi Puig-Suari from California Polytechnic State University proposed a reference design for a small satellite (Toorian, et al., 2005). The purpose was to allow graduate students to design, build, test and operate partial capabilities of larger satellites to reduce budget and time constraints. Thus, to provide greater sustainability in education and space research (Toorian, et al., 2005).

An important aspect about reducing the size of large satellite, consists of reducing the cost of deployment, minimizing of risks and lowering the amount of work that would be required to build and launch larger satellites (Baker & Worden, 2008). CubeSats have been integrated for different space missions, such as in governmental, educational and military sectors, enabling current and future interplanetary missions to be performed with reduced financial assets, in comparison to larger satellites, likewise providing a cost-effective prospect for science investigations and advanced mission concepts (Chin, et al., 2017). CubeSats are less expensive to launch into space with contrast to larger satellites. Furthermore, certain space agencies have special programmes to fund the deployment of CubeSats into space for University students (Foust, 2015).

1.1. Project Scope

The scope of this project consists on developing an Attitude Determination system for a CubeSat. There are different methods of determining the attitude of a CubeSat. A distinctive method is to use sensors to sense the orientation or position of the CubeSat such Inertial Measurement Unit (IMU) sensors, horizon sensors, star trackers and other sensors (Farhat , et al., 2013). This project aims to perform an experiment on Attitude of a CubeSat, with significantly reduced, financial assets using photodiodes for a low-price sun sensor and an IMU sensor to perceive the orientation of the CubeSat.

1.2. Motivation

Satellites are required to point to a certain location or direction. Different methods have been developed to orientate and stabilize the satellite into a desired direction. However, some of the methodologies used to solve this issue are expensive and difficult to implement for educational purposes (Daffalla , et al., 2016). The motivation of this project was found in its cost effectiveness. To intentionally designs and present an unexpansive system to perform the attitude determination of a CubeSat.

1.3. Background

1.3.1. Pumpkin CubeSat kit

The CubeSat kit encompasses hardware and software support to operate as groundwork for a CubeSat mission, from introductory development to final deployment. CubeSats can be designed and manufactured with different sizes. The size is categorized by Unit class. Standard CubeSats is 1U size that is 10cmx10cmx10 cm, with 1 to 1.33kg per unit (Crook & Sakoda, 2009). The pumpkin CubeSats kit is 1U CubeSat. The kit brings a modular chassis system, a flight module that supports an ultra-low-power MSP430F2618 single chip with less than 100 mW, a development Board and a pluggable processor Modules (PPM). Miscellaneous parts such as spacers, switches and Universal Serial Bus (USB) cable, programming adapter for FM430 flight module, TI'S parallel MSP430 JTAG FET programmer with cables, code composer studio (CCS V4.1) software CD-ROM and benchtop +5 power supply. The 1U flight module can oblige up to 4 to 16-bit PC104-compliant modules (Pumpkin , 2003).

1.3.2. Modular Chassis System

The pumpkin chassis is a space proven structure and it has demonstrated reliability in space application by a long list of CubeSats already launched (Smitha, et al., 2010). In this manner, pumpkin structure considers a monocoque approach (Alemayehu & Rezene, 2018). Monocoque approach is a structural method by which loads are sustained by an external skin object. The monocoque approach facilitates into maximizing the internal volume (Lis, 2013). The chassis walls are simple to assembly, ten 3x5mm stainless steel flathead screws, which are used to maintain the complete structure connected. Due to its monolithic aspects, it has enough strength to resist external or internal forces in compression and in rotation (*Appendix 1*). The structure is manufactured from 5052-H32 aluminium sheet metal, hard-anodized and Alodine to meet the CubeSat Design Specification (CDs) equipment (Cote, et al., 2011).

1.3.3. Motherboard

The motherboard can support different processors, its applicable for CubeSat Command and Data Handling (C&DH), Telemetry, Tracking and Command (TT&C), mass storage and battery (*Appendix 1*). It provides the general purpose of a low-power computing in PC/104-size form factor input and output pins (GPIO) allowing the integration of other circuit boards to the motherboard. The motherboard has an open architecture, it admits pluggable processor Modules (PPM) with different power levels and input and output (I/O) prerequisite. A Pluggable Processor Modules is a miniature hardware module that connects a microprocessor with its input and output peripherals. The motherboard supports a wide range of transceivers and power sources. Each pin has low quiescent current with less than 10 μ A. It integrates 3V lithium backup battery (BR1225), USB 2.0 device interface to PC, external DC power and to charge the battery, an MMC/SD card sockets for mass storage. All on-board peripherals input/output are level shifted and zero-power-isolated to connect with a pluggable processor module at any input or output voltage from +1.65V to +5.5V (Pumpkin, Inc., 2012).

1.3.4. Development Board

The development board is a laboratory platform to conduct in-house experiment. It provides to novices a training platform for developing, debugging and testing (*Appendix 1*). The development Board has all the functionalities of the motherboard. However, some features were added to suit laboratory experiments. Therefore, the development board has in line jumpers for circuit isolation and current measurements, it provides +5V and +3V power supply, reset switch to reset the board, LEDs indicator for the on and off status, two 104-pin¹ CSK Bus connectors, a large area for measurements and configurable RS-232 serial port with DB-9 connector. The development board is electrically programmable. The board allows the changes of different voltage levels (Pumpkin, Inc, 2012).

¹ Double connectors with 52 pins each row.

CHAPTER 2

Section 2: Literature Review

2.0. Attitude determination system

Attitude is the orientation of a body with respects to a certain frame (Acharya, 2014). Attitude determination system (ADS) is the module of the satellite responsible for stabilizing and orientating the satellite towards an assigned direction (Crowell, 2011). ADS is an essential part of the satellite; its functionality can affect the performance of the CubeSat. An ADS module integrates different sensors to determine the orientation of the satellite. Magnetometers, GPS, Gyroscope, horizon and sun sensors are the sensors frequently used to obtain the orientation of the CubeSat (Bezold, 2013) (Theil, et al., 2017). There are different reasons associated with the application of an ADS module in satellites, this could be due to CubeSats being powered through solar panels (Bugryniec, 2016), it is needed to ensure that the solar panels is pointing towards the sun to acquire the most power required, space telescopes demand an attitude determination module to provide an unobstructed view of a certain zone of space (Lyndon 6. Johnson Spaca Center , 1984) .

2.1. Coordinate systems

2.1.1. Inertial reference frame

Newton's first law states that an object either remains at rest or continuously moves at a constant velocity, unless another external force acts upon it, and this is described as inertia (Croce, 2005). Therefore, inertial frame of reference is when an object remains either at rest or continues at a constant velocity unless another external force acts upon the body (Harrison, 2002). In this project the orientation of the CubeSat is found with respects to the Inertial reference frames.

2.1.2. Body reference frame

Body frame is a fixed frame with the CubeSat. It has its origin at the centre of the mass of a satellite (*Appendix 1*). When the satellite moves, the body frame changes (Pillet, 2017). The orientation of the satellite in space is derived with respects to the orbit frame which would be a reference frame of the satellite in space (Bender, 2011) . The body angle of the CubeSat are the angles between the orbit frame reference and inertial frame reference.

2.1.3. Orbit reference frame

Orbit frame is the frame that rotates with the earth centred frame. It depends on the orientation of the orbit. For a circular orbit the tangent component is perpendicular to the radius vector (Bona & Canuto, 2002).

2.2. ADS sensors

In section 2.0, it was mentioned some sensors used to determine the orientation of the CubeSat. For instance, one of the most effective methods is to measure the attitude of the CubeSat by tracking the stars with regards to a reference frame. For several years, star trackers have been applied in satellites technology to obtain the attitude of satellites (Liebe, 1995). Star trackers are usually divided into three categories: star scanners, gimbaled star trackers and fixed head star trackers (Smith, 2017). Fixed head star trackers track the capabilities of the star over a constrained field of view (FOV). On the other hand, star scanners provide searching and sensing, whilst gimbaled star trackers use mechanical action to search for the star (Smith, 2017).

Another sensor frequently used for attitude determination is the horizon sensor. They sense points on the earth's horizon, it has been applied to many earth – orbiting satellites. Horizon sensors are designed to perform a measurement in uniform infrared wavelengths because, it is difficult to obtain accurate measurements in visible wavelengths (Wessels, 2018). They sense the infrared radiation range. There are two types of horizon sensors; static horizon sensors and scanning horizon sensors. Static horizon sensor points at the fixed direction and they are restricted to small pitch and roll angles or attitude range, while scanning sensors scans a small field of view across the earth. In comparison to star trackers that find the satellite position in a reference frame, horizon sensors sense a point on the earth horizon without reference frame (Stengel, 2016). Global Positioning Systems (GPS) are often fused with IMU to provide an accurate orientation (Farrel & Barth, 1999). Sun sensors were designed and manufactured for this project, the next section will give an overview of the theory concentrated on sun sensors.

2.2.1. Sun sensor

Sun sensors are usually applied in satellites to inform the position of the CubeSat with respects to the sun (Virk, 2012). There are two common types of sun sensors, analogue and digital sun sensors. Analog sun sensors output continuous function of the sun angles. On the other hand, Digital sun sensors output discrete values as function of the angle incident (NASA, 1970). Due to the properties of the sun, it is possible to design a sensor that is light dependent. Sun sensors are mainly employed in satellite to protect sensitive equipment such as star trackers, to supply a reference for on-board attitude determination and control system and to locate the solar power arrays (Svartveit, 2003).

The sun sensors attached at the CubeSat provides the body angles, the body angles of the CubeSat are calculated in reference to the sun. Field of View (FOV) is the diameter that the light is allowed to fall. The body angles are directly related to the field of view of the sensor. The configuration and structure of the sun sensors affect the field of view of the sensor. The field of view depends on the application of the sensors. 180° field of view would have a resolution of several degrees less than an arc second (Carmona, 2017). The selection of sun sensor is directly related to the field of view, angular resolution, its accuracy and stability, output characteristics and its durability. The most common form of sun sensors is based on the photocells and they output an electric current which is directly proportional to the intensity of the sun light falling on the photocell (Manickam & Ranjan, 2013).

The output of the sun sensor gives the average direction of the intensity of the light in the photocell. Equation (1) shows that the electrical current is directly proportional to the light. In the absence of light, the current (I_i) produced by the light falling in the sun sensor will be approximately 0A. However, if the sun sensor is directed towards the light source, the sun sensor will output the maximum current (Springmann, 2013), (Post, et al., 2013):

$$I_i = \begin{cases} I_{max}(n_i, s) \\ 0 \end{cases} \quad (1)$$

By finding the body angles, it does not describe the full orientation of the CubeSat, to give a clearer meaning to the output of the sun sensor, the sun vector must be deduced. The dimension of the vector depends on the quantity of the sun sensor around the CubeSat. A single sun sensor will produce one axis vector. Using three or six sun sensors orthogonal to one another, 3D vectors can be deduced. From figure 1, 'n' represents the vector direction of the sun sensor pointing to the normal, so it can be presumed that the inverse normal $n_{-i} = -n_i$ so:

$$I_{-i} = \begin{cases} I_{max}(n_{-i}, s) \\ 0 \end{cases} \quad (2)$$

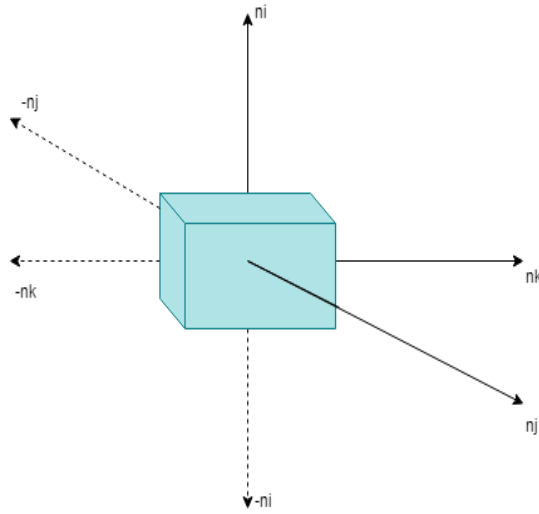


FIGURE 1: 3D FRAME OF THE CUBESAT

The difference between the two outputs currents ($I_i - I_{-i}$) will be equal to the maximum current $I_{max}(n_i, s)$, note that the current is at the maximum when the light is directed towards the sun. Based on this idea, the current output when the light falls on the other sides of the CubeSat can be found. For cartesian basic vector K.

$$I_k = \begin{cases} I_{max}(n_k, s) \\ 0 \end{cases} \text{ and } I_{-k} = \begin{cases} I_{max}(-n_k, s) \\ 0 \end{cases} \quad (3)$$

The difference between the two vectors will be the maximum current in k-direction $I_{max}(n_k, s)$

$$I_j = \begin{cases} I_{max}(n_j, s) \\ 0 \end{cases} \text{ and } I_{-j} = \begin{cases} I_{max}(-n_j, s) \\ 0 \end{cases} \quad (4)$$

Similarly, the difference between the two vectors will be the maximum current in J-axis $I_{max}(n_j, s)$. Note if the normal Cartesian vectors n_i , n_j and n_k are not coplanar i.e. if the vectors are not parallel to the same plane or lie on the same plane (Markley & Crassidis, 2014). The vector directed from the CubeSat to the Sun or light source (S) can be found. Hence, taking in consideration six sensors with normal $\pm n_i$, $\pm n_k$ and $\pm n_j$ obtain (Markley & Crassidis, 2014):

$$\begin{bmatrix} I_i - I_{-i} \\ I_k - I_{-k} \\ I_j - I_{-j} \end{bmatrix} = I_{max} \begin{bmatrix} n_i \cdot S \\ n_k \cdot S \\ n_j \cdot S \end{bmatrix} \quad (5)$$

In consequence of this, S for six sensors becomes:

$$S = \frac{(I_i - I_{-i})(n_j \times n_k) + (I_k - I_{-k})(n_i \times n_j) + (I_j - I_{-j})(n_i \times n_k)}{I_{max}[n_i \cdot (n_j \times n_k)]} \quad (6)$$

If three sensors are used, which is the case for this project, the vector directed from the CubeSat to the sun or to the light source can be expressed as shown in the Equation (7). Note that as three sensors are being used to deduce the Equation (7), this means, the presence of the light beam will only be sensed in three sides of this CubeSat which is also enough to express 3D vectors, this equation is particularly important as it aids to identify the body frame angles in terms of the current produced by the sun sensor as the light falls on the sun sensor, which will be discussed further in the report:

$$S = \frac{(I_i)(n_j \times n_k) + (I_k)(n_i \times n_j) + (I_j)(n_i \times n_k)}{I_{max}[n_i \cdot (n_j \times n_k)]} \quad (7)$$

To design the sun sensor, it is important to take into consideration the configuration and field of view of the sun sensors. The sun sensor for this project is analogue sensor, Analogue sun sensors also known as cosine detectors are sensors based on the variation of the output current of solar cell (Wertz, 1990), if the design of the sun sensor is based on 180° FOV or less, the same concept could be applied to find the output current from the photocell i.e. cosine detector.

Although the determination of the current requires that the sensitive area of the sensor is directly proportional to the cosine of the angle of incident light. Figure 2 shows the model based on a single sun sensor. Note from the first concept introduced that the normal vector represents the maximum current or when the sensor is directly pointing to the sun.

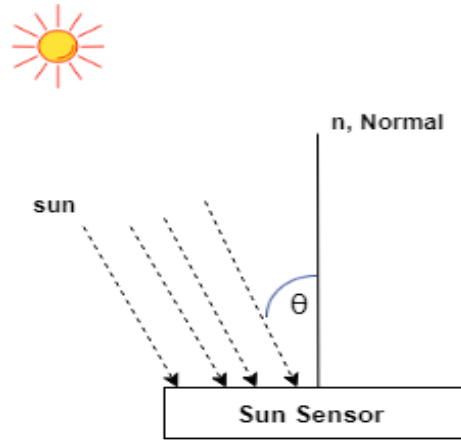


FIGURE 2: SUN SENSOR DETECTING INCIDENT ANGLES

The sun sensor will output a variation of the current depending on the light introduced on the sensitive area of the sensor. As the light falls into the sensor, it will produce an energy flux (E) across the sun sensor surface with area ' dA ' and a normal ' n '. Thus, using basic trigonometry, it is possible to deduce that:

$$I(\theta) = I_{max} \cdot \cos \theta \quad (8)$$

It is evident from equation (8), that if the CubeSat directs the sun sensor towards the sun simulator, the body angle (θ) will be equal to the FOV, assuming that the FOV is of 180° the equation becomes:

$$I(\theta) = I_{max} \quad (9)$$

Therefore, it can be assumed that the sun sensor is directed towards the sun, it is pointing to a Unitarian normal direction as:

$$\cos \theta = 1 \quad (10)$$

As Aforementioned, the configuration of the sun sensor is crucial if a higher degree of freedom was required. A common method is to use a slit. The slit will not only provide the possibility of adding more degrees of freedom (DOF), but it will also reduce the field of view of the sun sensor, which is recommended due to albedo and glint effect. Although different configuration could be considered, the concept of the sun sensor originates from the same hypothesis. Figure 3, shows a sun sensor with a single slit, denote that the light falling in the sun sensor will be limited by the size of the slit and it also reduce the field sensor.

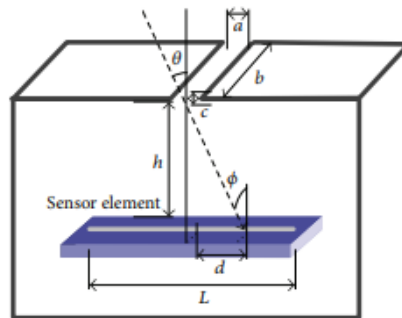


FIGURE 3: SINGLE SLIT (POST, ET AL., 2013)

This configuration provides only a single axis sensing. As the slit is placed vertically centred to the sensor, the light will be limited 45 degree to -45 degree the total field of view is of 90 degree (Post, et al., 2013).

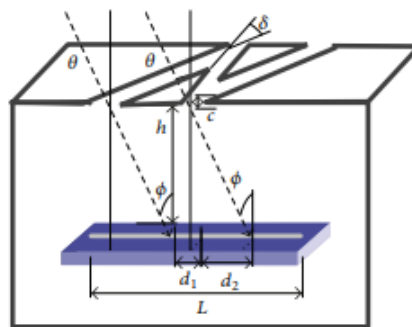


FIGURE 4: CONFIGURATION SUN SENSOR (POST, ET AL., 2013)

Figure 4 shows a sun sensor with Z configuration. In this case, the light is no longer limited by a single slit, therefore it is possible to obtain two-axis with this configuration. The Z configuration covers two 90° with a pyramidal volume of view. This configuration may be acquired easily in CubeSat, due to the external structure, but it remains dependent on the mission pointing purpose (Post, et al., 2013). An alternative way of obtaining two-axis would be to use two sun sensors at the same surface of the CubeSat, using a sun sensor based on linear photocell, it is possible to place one at horizontal and vertical axis. Refer to (Springmann, 2013)

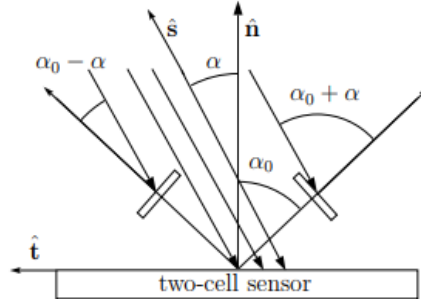


FIGURE 5: TWO SENSORS AT THE SAME PLANE (WERTZ, 1990)

From the body diagram, it is possible to determine the angle in each specific plane separated by the normal vector. This method uses the sun sensors tilted at a certain angle α_0 relative to the normal vector \mathbf{n} (Wertz, 1990). The sun vector \mathbf{S} is projected into planes \mathbf{n} and then the intensity of a current alters accordingly (Markley & Crassidis, 2014):

$$I_1(\alpha) = I_{max} \cos(\alpha_0 - \alpha) \quad (11)$$

And

$$I_2(\alpha) = I_{max} \cos(\alpha_0 + \alpha) \quad (12)$$

The difference between the currents:

$$\Delta I = I_2 - I_1 \quad (13)$$

$$\Delta I = I_{max} [\cos(\alpha_0 + \alpha) - \cos(\alpha_0 - \alpha)] \quad (14)$$

Using the trigonometric identity: $\cos(\alpha_0 + \alpha) = \cos(\alpha_0) \cdot \cos(\alpha) + \sin(\alpha_0) \sin(\alpha)$

$$\Delta I = 2I_{max} \sin \alpha_0 \sin \alpha \quad (15)$$

To note that from figure 5, it has three-unit vectors, two normal ' \mathbf{n}_1 '² and ' \mathbf{n}_2 '³ acting at the different planes and ' \mathbf{t} ' acting at t-plane. It can be deduced that the sun vector \mathbf{S} is a vector with

² It is located at the left side of the plane as one of the sensors will be tilted at the left, refer to figure 5, it is parallel to \mathbf{n} .

³ It is located at the right side of the plane as one of the sensors will be tilted at the right, refer to figure 5, it is parallel to \mathbf{n} .

respects to $\mathbf{n1}$, $\mathbf{n2}$, \mathbf{t} . so that $\mathbf{S} = \{\mathbf{n1}, \mathbf{n2}, \mathbf{t}\}$ (Markley & Crassidis, 2014). The orientation of the CubeSat is determined by the body frame angles (Wertz, 1990).

The sun sensor vector can be determined from the two measured angles by assuming that the component at the direction of $\mathbf{n1}$ is equal to one. Then it is clearer that if the components are in $\mathbf{n2}$ plane, \mathbf{S} will be the ratio of the tangent of both angles, and at \mathbf{t} component direction \mathbf{S} will be equal to the tangent of the angle made in the $\mathbf{n1} - \mathbf{t}$ plane. Then sun sensor vector can be derived as (Markley & Crassidis, 2014):

$$\mathbf{S} = [1 \quad \frac{\tan \alpha_1}{\tan \alpha_2} \quad \tan \alpha_1]^T \quad (16)$$

Thence the sun vector needs to be normalized. Hence, normalized sun vector (\mathbf{S}) can finally be calculated as (Markley & Crassidis, 2014):

$$\mathbf{S} = \frac{\begin{bmatrix} 1 & \tan \alpha_1 / \tan \alpha_2 & \tan \alpha_1 \end{bmatrix}}{\sqrt{\mathbf{S}^T \mathbf{S}}} \quad (17)$$

Therefore, if the CubeSat has three sun sensors, each side a single sensor, the sun or solar vector which depends on the body angles can be mathematically calculated by (Cutler & Springmann, 2012):

$$\vec{S} = \begin{bmatrix} \frac{\tilde{V}_i}{V_{max,i}} & \frac{\tilde{V}_j}{V_{max,j}} & \frac{\tilde{V}_k}{V_{max,k}} \end{bmatrix} \quad (18)$$

Note that the equation describes the sun vector in terms of voltage, however this can be easily converted to current using ohm's law.

2.2.2. Inertial Measurement Unit Sensor (IMU)

Inertial measurement unit sensor also known as IMU Sensor are frequently used in satellites to acquire the absolute orientation of the satellite. These sensors usually provide 6-axis or 9-axis orientation. IMU sensor integrates 3-axis accelerometer, 3-axis gyroscope and 3-axis magnetometer (Tran, 2017). A great advantage of IMU sensor is that they are small in size which makes easy to implement in any technology.

2.2.2.1. Gyroscope

Gyroscope is a device that measures the angular velocity of a body in respect to the inertial frame. They are usually based on the Micro-Electro-Mechanical systems (MEMS). A small resonating mass attached to the MEMS is shifted as the angular velocity changes. Newton's law of conservation of momentum states that the total momentum of an object or body is constant in closed system (Tamir, 2013). The gyroscope obeys Newton's law of conservation of momentum. The output motion of the object is converted into electrical signals which is amplified to make data readable by microprocessor. Gyroscopes have usually 2-axis or 3-axis, each channel will measure a single axis. To select a gyroscope, one must take into account, its sensitivity range and application. For instance, the maximum angular velocity expected should not be greater than the maximum range of the gyroscope. The angular velocity can be mathematically expressed as (Kok, et al., 2017):

$$\omega_{ib}^b = R^{bn}(\omega_{ie}^n + \omega_{en}^n) + \omega_{nb}^b \quad (19)$$

ω_{ib}^b is the angular velocity, R^{bn} is the rotation matrix from the moving frame to the body frame, ω_{ie}^n is the angular velocity of the earth frame earth in relation to the inertial frame (Kok, et al., 2017). The angular velocity of the earths' frame is around 7.29×10^{-5} rad/sec. ω_{nb}^b is the transport rate which non -zero if the moving frame is not expressed static in relation to the earth. Gyroscopes tends to drift overtime, and this may be a drawback for this type of devices. To solve this, a low pass or high pass filter is usually implemented.

2.2.2.2. Accelerometer

The accelerometer is a device that measure the acceleration of body moving at a certain frame. There is various type of accelerometer, and they can measure one, two or three orthogonal axis. This is usually described by m/s^2 or g-force where 1g is equivalent to 9.8 m/s^2 , accelerometer can either sense dynamic and static acceleration. Static acceleration comprises gravity, while dynamic acceleration has an abrupt start or stop motion. Due to its static characteristic an accelerometer can provide the orientation of moving body in relation to the earth's surface. The accelerometer will measure all the force acting upon a body (El-Fatratry, 2003). If an object is positioned at standstill on the earth the only force applied to this body will be the gravitational force (Kok, et al., 2017). The force provokes an object with a mass to vary its velocity that is to accelerate ($F = m.a$). If an object is placed at the Z direction facing the ground the acceleration at the Z direction will be equal to 1g or 9.81 m/s^2 . An object at the standstill state on earth reference frame, the total acceleration is 1g (El-Fatratry, 2003):

$$a_x^2 + a_y^2 + a_z^2 = 1g \quad (20)$$

With contrast to the gyroscope, the accelerometer is accurate in the long term and can provide precise pitch and roll orientation of the object. However, this device alone cannot provide the heading or yaw orientation, it is for this reason this project uses the IMU sensor. The linear acceleration of the sensor a_{ii}^n the its relationship with the required acceleration (a_{nn}^n) for the navigation purposes can be described as (Kok, et al., 2017):

$$a_{ii}^n = a_{nn}^n + 2\omega_{ie}^n \times \vartheta_n^n + \omega_{ie}^n \times \omega_{ie}^n \times p^n \quad (21)$$

$\omega_{ie}^n \times \omega_{ie}^n \times p^n$ represents the centrifugal acceleration and $2\omega_{ie}^n \times \vartheta_n^n$ is the Coriolis acceleration. The centrifugal acceleration is absorbed by the gravity vector in many cases (Kok, et al., 2017). When selecting the IMU sensor, the sensitivity range of the accelerometer should be selected according to the application of the device (Chaudhry, et al., 2018). A reduced full-scale sensitivity range means a more sensitive output, so the accelerometer will provide more precise readings at low scale sensitivity range.

2.2.2.3. Magnetometer

Magnetometer is a device that sense the magnetic field's strength point at a certain direction. This is usually used to determine the heading angle due to the limitation of accelerometers to provide heading or yaw angles and the gyroscope drifts characteristic. Similar to gyroscope and accelerometer, magnetometer devices also offer three axis measurements. The magnetometer is used to improve the orientation determination. Figure 6 shows the direction of the magnetic field around the earth.

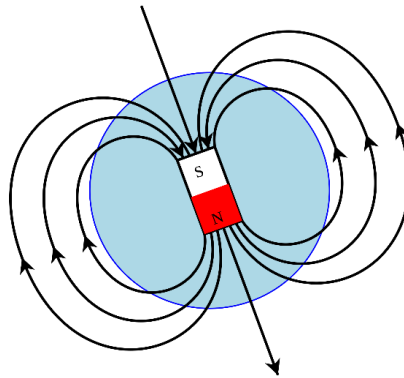


FIGURE 6: MAGNETIC NORTH POLE

The magnetic field of the earth can be looked as a magnet bar. From figure 6, it can be seen that the field line of the magnetic dipole originates from south pole and ends at the north pole. These two poles are known as magnetic poles, note that the field lines will change in both direction and strength depending on the orientation of the earth (Gunnarsdóttir, 2012). The earth's rotating axis defines the north and south poles. The poles are defined geographically which are used for map references. However, there is a small discrepancy between the geographic poles and the actual magnetic poles (Gunnarsdóttir, 2012). This discrepancy is known as the declination angle. In Glasgow the declination angle is of 2.6 °N.

2.3. Rotation Matrix

To perform the orientation of CubeSat using the IMU and sun sensor, rotation matrix that describes the dynamic of the CubeSat must be deduced (Kuipers, 1999). The rotation matrix helps to understand the translation or rotation a body will perform as it changes from different frame system or from the reference frame to a different frame. These angles are usually described as pitch, roll and heading or yaw. (see figure 7).

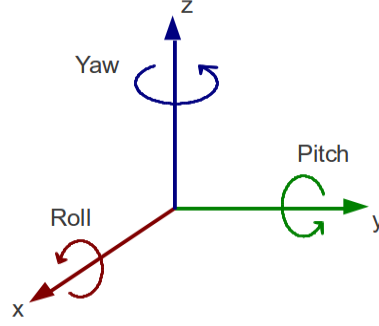


FIGURE 7: FRAME OF ORIENTATION

Denote from the body frame above, the heading (yaw) will be an angle that rotates around the z-axis and at z-axis the rotation matrix can be described as:

$$R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(22)

A pitch angle is the angle around the y-axis. The rotational matrix at the y-axis can be described as follow:

$$R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

(23)

A roll angle is the angle that rotates about the x axis. The rotational matrix at the x-axis can be described as follow:

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

(24)

Note that by pitching the CubeSat, that is by rotating the CubeSat at the Y-direction a 2D rotation will be developed between X and Z coordinates and Y will be constant. An absolute orientation can be found by having different rotational matrices being pre-multiplied or post-multiplied by other rotational matrices depending on the application. The product of this matrices depending on its order can define the Euler angles, this will be discussed at the next section. Different results should be expected by changing the order these matrices are multiplied.

2.4. Euler Angles

The orientation of the CubeSat can be described as a series of three rotations from the reference frame to any moving frame. Euler angles define the 3D orientation of the object with relationship to a global frame system. Denote from the figure 8 below, the blue line represents the reference frame, the red line is the frame that moves with the CubeSat. Recalling rotation matrix, Euler angles can provide the information on how the CubeSat is moving by composing 3X3 rotation matrix (Equations 22 -24). Figure 8, note that the green line represents the magnetic pole, so the Euler angles of moving body are found by taken in consideration the magnetic pole frame.

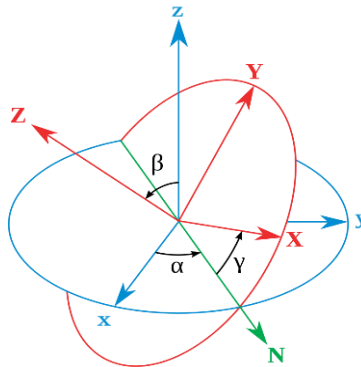


FIGURE 8: EULER ANGLES AT REFERENCE FRAME

The Euler angles are commonly denoted as ψ or γ , ϕ or α and θ or β . For this project will use the follow notation ψ (heading), θ (pitch) and ϕ (roll), these angles represent the Euler angles. The rotation matrix can be described in terms of Euler angles as follow:

Rotation of ψ around x axis is expressed as:

$$R_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \psi & -\sin \psi \\ 0 & \sin \psi & \cos \psi \end{bmatrix} \quad (25)$$

Rotation of θ around y-axis is expressed as:

$$R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (26)$$

Rotation of ϕ around z-axis is expressed as:

$$R_z(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (27)$$

The general rotation matrix is the product of the three matrices and the order of its multiplication, will influence the result. This project uses the XYZ coordinate system. There are twelve possible sequences of rotation (Kuipers, 1999). Note that although the Euler angles is totally similar to the rotation matrix. Yet, the final rotation is about the moving reference frame:

$$Euler(\phi, \theta, \psi) = Rot(a, \phi)Rot(o, \theta)Rot(a, \psi) \quad (28)$$

$$= \begin{bmatrix} \cos \phi \cos \theta \cos \psi - \sin \phi \sin \psi & -\cos \phi \cos \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \sin \theta & 0 \\ \sin \phi \cos \theta \cos \psi + \cos \phi \sin \psi & -\sin \phi \cos \theta \sin \psi + \cos \phi \cos \psi & \sin \phi \sin \theta & 0 \\ -\sin \theta \cos \psi & \sin \theta \sin \psi & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (29)$$

The **a**, **o** and **n** are the axis of moving frame, refer to figure 8, the red axis. Following Dr. McGookin's derivation, the deduction of the Euler angles can be presented as (Kuipers, 1999):

$$\phi = \tan^{-1}\left(\frac{a_y}{a_x}\right) \text{ or } \phi = \tan^{-1}\left(\frac{-a_y}{-a_x}\right) \quad (30)$$

$$\theta = \tan^{-1}\left(\frac{a_x \cos \phi + a_y \sin \phi}{a_z}\right) \quad (31)$$

$$\psi = \tan^{-1}\left(\frac{-n_x \sin \phi + n_y \cos \phi}{-o_x \sin \phi + o_y \cos \phi}\right) \quad (32)$$

2.5. Kalman filter

Kalman filter offers an efficient recursive method to estimate the state of a process, so that the mean of the squared error (MSE) can be reduced. Kalman filter is powerful tool due to its different aspects. This filter consists of two stages: measurement update and time update. This means that the filter is competent to dynamically perform calculation and update its measurement. The Kalman filter works by producing an optimal estimate of the system state based on the measurements (Lauszus, 2012).

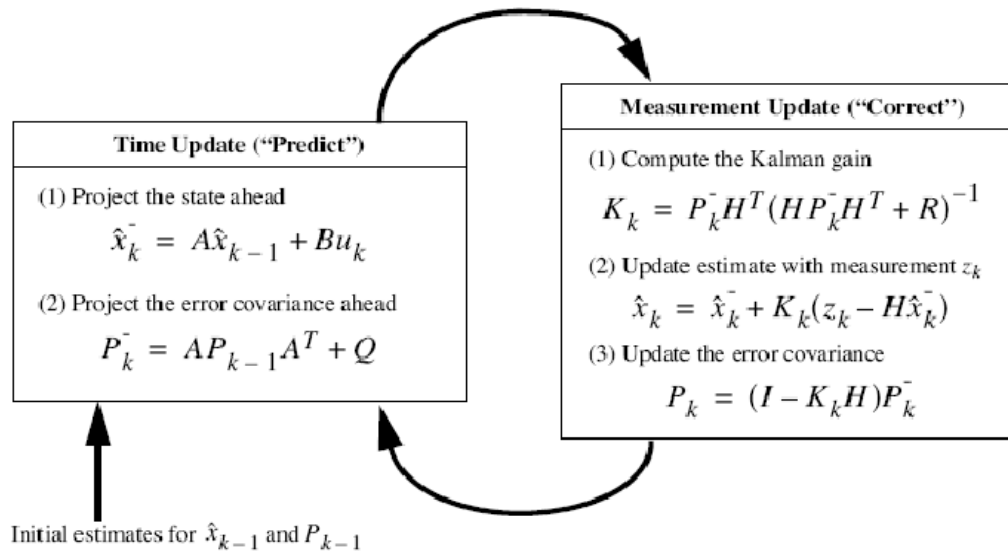


FIGURE 9: KALMAN FILTER (ALIABADI & SHARBAFI, 2010)

The Kalman filter uses a form of feedback control, this can be seen in the figure 9, it estimates the process state for a certain time and acquires noisy as feedback. Note, that the time update equations will be in charge to predict the current state and error covariance estimation, to get the priori estimation for the next step process. Priori estimation indicates the estimate of the state matrix at real time k which depends on the previous state of the system and the estimate states before it (Welch & Bishop, 2006) (Lauszus, 2012).

On the other hand, the measurement update equations are in charge to provide the feedback so that a new measurement is merged into a priori estimate, by doing this, an improvement will be noted in the posteriori estimate. The posteriori estimate is the estimation of the state at a certain time k based on the experience and observation. This paper is concerned with the implementation of the filter rather than derivation of the filter. Refer to (Lauszus, 2012) and (Welch & Bishop, 2006) for the derivation of the filter. To perform the Kalman filter, it is necessary to know the noise of the input to the filter. The noise in the system has to be Gaussian distributed and have an average zero. To implement the filter, the first step describes the state of the system equation at a certain time k . \mathcal{X}_k is state vector at time k , F is the state transaction model matrix applied to the previous time, \mathcal{W}_k is the associated white noise.

$$\mathcal{X}_k = F\mathcal{X}_{k-1} + B\mathcal{U}_k + \mathcal{W}_k \quad (33)$$

$$\begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_{k|k-1} = \begin{bmatrix} \theta + \Delta t(\dot{\theta} - \dot{\theta}_b) \\ \dot{\theta}_b \end{bmatrix} \quad (34)$$

Refer to (Lauszus, 2012) for the derivation of the state vector. Note that from the equation 33, that the priori estimate ($\dot{\theta}_{k|k-1}$) will be equal to the estimate of the previous state ($\dot{\theta}_{k-1|k-1}$) added the rate of time (Δt).

$$P_k = FP_{k-1|k-1}F^T + Q_k \quad (35)$$

$$\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} P_{00} + \Delta t(\Delta t \cdot P_{11} - P_{01} - P_{10} + Q_\theta) & P_{01} - \Delta t P_{11} \\ P_{10} - \Delta t P_{11} & P_{11} + Q_{\theta_b} \Delta t \end{bmatrix} \quad (36)$$

Then second step compiles the estimation of the priori error covariance matrix (P_k) and this error depends on the previous error covariance matrix.

$$\tilde{y}_k = z_k - H\hat{x}_{K|K-1} \quad (37)$$

$$\tilde{y}_k = z_k - \theta_{K|K-1} \quad (38)$$

To update the measurement, Equation (38) was used, this is the difference between the mean measurement and the a priori state. \tilde{y}_k is named the innovation in many applications. The large the noise measurement is the large is the innovation covariance S_k , can be calculated using Equation (40).

$$S_k = HP_{k|k-1}H^T + R \quad (39)$$

$$S_k = P_{00_{k|k-1}} + \text{Var}(v) \quad (40)$$

This equation will attempt to predict the accuracy of the measurement depending on a priori error covariance matrix and measurement covariance matrix. The Kalman gain is an important component in Kalman filter since it allows one to change accordingly. The Kalman gain indicates how accurate the innovation is, it is mathematically described as:

$$K_k = P_{k|k-1}H^T S_k^{-1} \quad (41)$$

$$K_k = \frac{\begin{bmatrix} P_{00} \\ P_{10} \end{bmatrix}_{k|k-1}}{S_k} \quad (42)$$

To update the posteriori estimate ($\hat{x}_{k|k}$) of the current state, it used:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k \quad (43)$$

$$\begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_{k|k} = \begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_{k|k-1} + \begin{bmatrix} K_0 \tilde{y} \\ K_1 \tilde{y} \end{bmatrix}_k \quad (44)$$

The last step consists in updating a posteriori error covariance matrix ($P_{k|k}$) and this can be mathematically described as:

$$P_{k|k} = (I - K_k H) P_{k|k-1} \quad (45)$$

$$\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k-1} - \begin{bmatrix} K_0 P_{00} & K_0 P_{01} \\ K_1 P_{00} & K_1 P_{01} \end{bmatrix} \quad (46)$$

2.6. Conclusion

This section aimed to expound the literature reviewed used to compile this project. The sun sensor and IMU 10 axis Pmod Nav were chosen to sense the orientation of the CubeSat. The sun sensor was selected amongst other sensors because they are less complex to produce in-house and its components are accessible. The Pmod Nav has advantages in comparison to other inertial measurements unit sensor because it brings 10 axis and one of the axes can provide the altitude of the satellite which could be used for further development of this project in an area which requires information regarding the altitude of the satellite. All the theories described in the literature review and background were used to design and develop the technical part of the project. The next chapters will focus on the application of these theories to build the hardware and software of this project.

CHAPTER 3

Section 3: Hardware and Software Development

3.0. ADS Hardware

The hardware for the CubeSat is composed of two microprocessors, the MSP430 and Arduino Uno Board. The MSP430F2618 from Texas is used by the development board (see Chapter 1, Section 1.4.), it performs the algorithm for the sun sensors via 12-bit analogue to digital converter channels. The MSP430F2618 reads the analogue input of the sun sensors and converts it into digital values. In section 3.3, of this chapter it is illustrated how the ADC Setup is performed in the MSP430. Figure 10 shows the block diagram for the entire system.

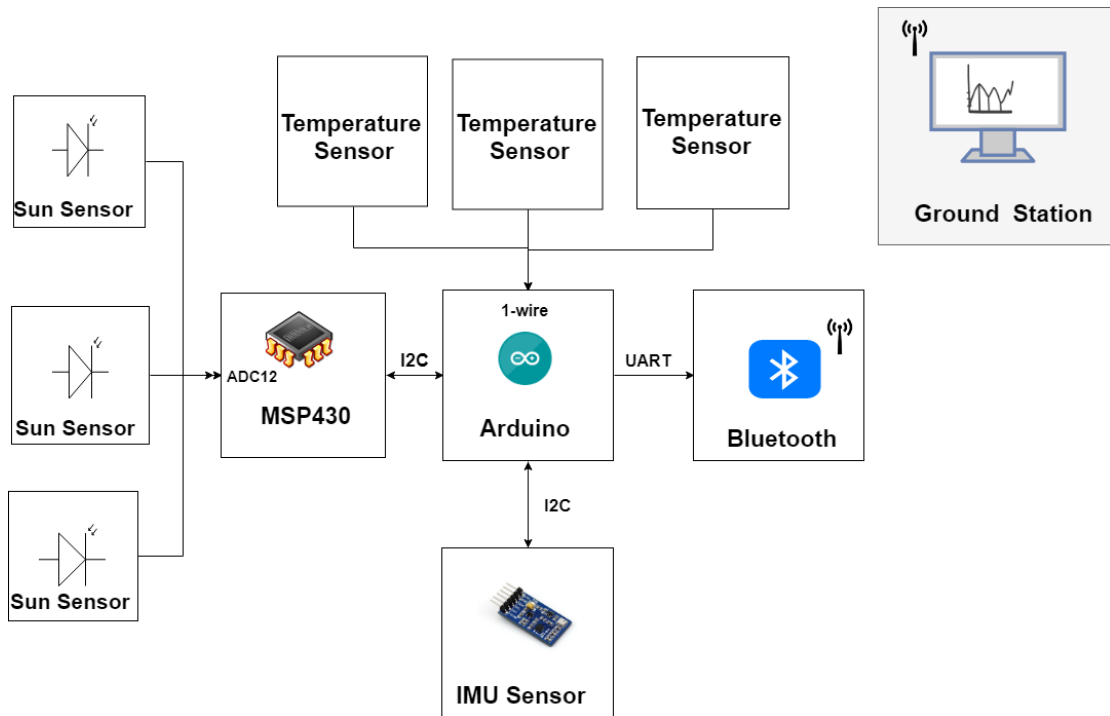


FIGURE 10: BLOCK DIAGRAM OF THE ENTIRE SYSTEM

The Arduino Uno Board is an additional board to the entire system that interfaces with IMU sensor, Bluetooth device and temperature sensors. The MSP430 Development Board from pumpkin has been configured in a windows 7 machine, in consequence of this, it may not be possible to send serial in some windows 10 PCs through USB 3.0 Port. However, it may work well for a system that has USB 2.0 Port. In order to transfer serial data from the board to PC, an Arduino board is interfaced with development board through I2C protocol.

The Arduino board is a microcontroller based on the ATmega328, which can easily be replaced by a self-made board alongside an oscillator. The board contains 14 digital input/output pins which 6 of them can be used as PWM (Pulse Width Modulation), and it has six analogue pins. Arduino Uno uses 16 MHz Crystal Oscillator, and a USB connection to maintain the communication between the Arduino, and to also power the Arduino board from a PC.

The Arduino board operates with 5V, but it is recommended to use an input voltage of 7 to 9V with the limit voltage between 5 to 20V. The Arduino Board has 40mA DC current at each input and output pins, whereas the 3.3V supply pin has 50mA. Additionally, this microcontroller has 32KB of flash memory. Furthermore, the Arduino connects the IMU Sensor via I2C protocol and Bluetooth device via UART Communication protocol. In addition to this, it contains the temperature sensors which are connected to the Arduino via one-wire communication. The entire system was designed to be supplied from either the development board (MSP430F2618) or Arduino board. However, it is strongly recommended that the system should be powered from the development board, due to power instability from Arduino which can eventually provoke an unexpected data result.

Figure 12 illustrates the schematic of the entire system and shows how the Arduino communicates with the development connectors which are directly connected to the pluggable processor Module (PPM). The block diagram shows in a clear manner, how each module is interfaced. To note that, from both figure 10 and figure 12, it can be verified that both microprocessors are interfaced via inter-integrated circuit protocol. The Bluetooth is connected with the ground station (PC) via Universal Asynchronous Receiver/Transmitter protocol (UART). Note from the Figures 11 and 12, each component is placed into the board through connectors, each component's footprint was created specifically to be accommodated in this project. Figure 11 and 12 shows two connectors each with 52 pins, the connector J43 has the pins for the I2C protocol, P3.2 and P3.1 from Port 3 and connector J44 has eight input/output pins connected into it. Three pins P6.0, P6.1, and P6.2 from Port 6 are the ADC 12 bits channels, pins PPM_5V is directly connected to Arduino input voltage pin and it is responsible to power the entire system.

The entire system is common grounded to the analogue ground from the development board to ensure that the reference voltage (GND) is the same throughout the entire board; this prevents disturbance at the signals. To accommodate the main board (*appendix 1*) into the development board another connector had to be added as extension had to be added. Figure 11 is the PCB design of the main board with Arduino.

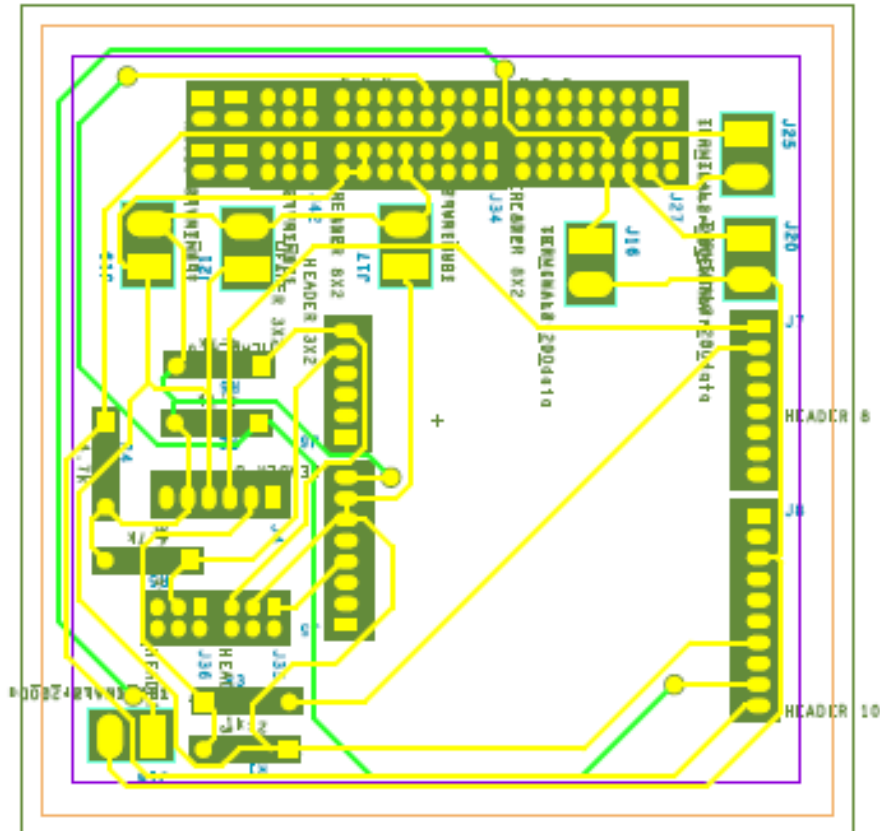


FIGURE 11: MAIN BOARD PCB DESIGN

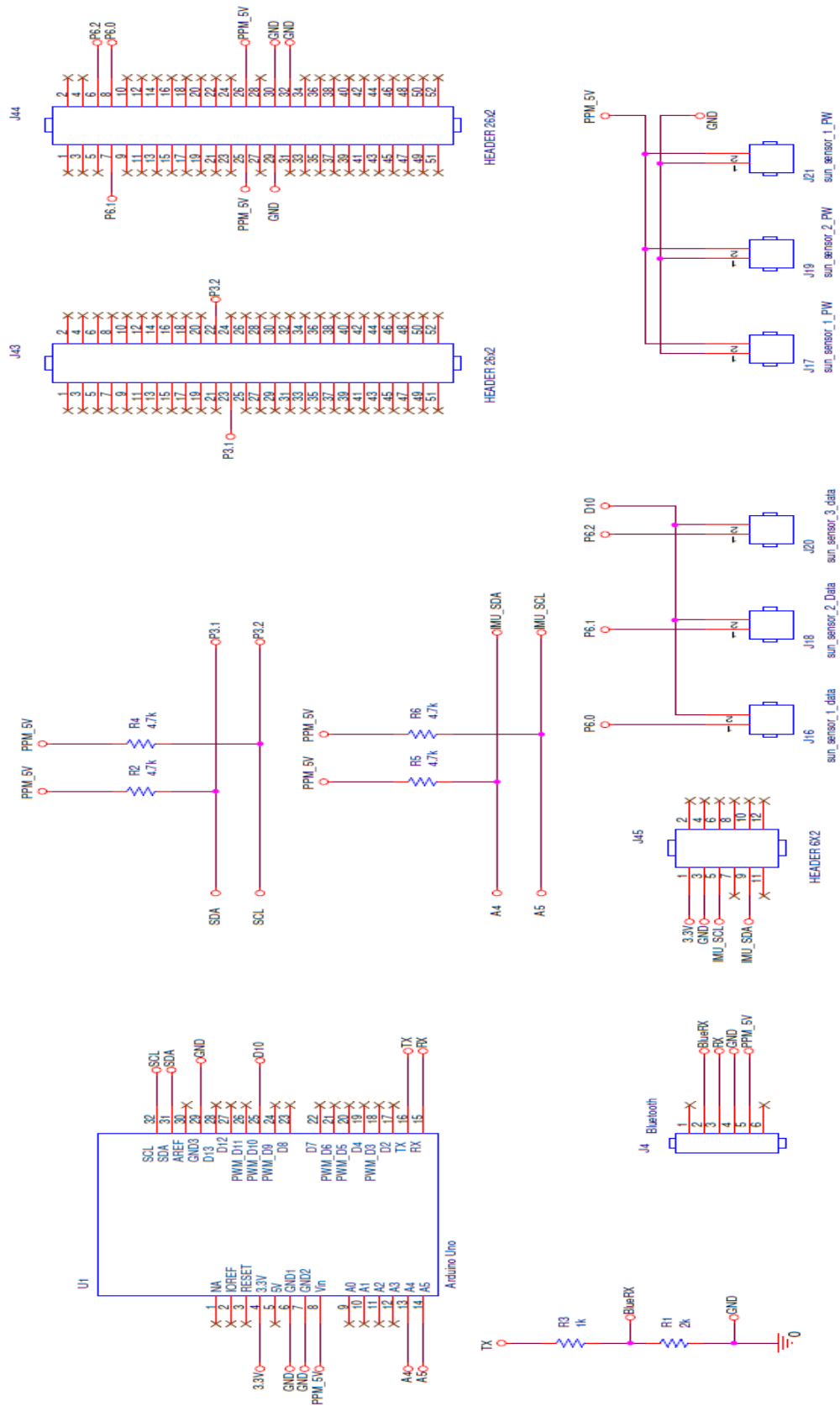


FIGURE 12: MAIN BOARD SCHEMATIC

3.1. Power Management

The diagram below illustrates the power distribution in the entire system. The power source is obtained from 230V socket, which is then regulated through an AC/DC switching Adaptor Model GS18A09. This AC to DC switching Adaptor takes as input range 100 to 240.0V_{AC}, 0.5A and approximately 50 to 60 HZ. The AC to DC switching Adaptor outputs 9.0V_{DC} and 2.0 A, 18 watt as maximum power supplied. The development Board takes 7 to 25V_{DC}, (max 25 V_{DC} and 4 A) and this voltage can be regulated in three different voltage levels, 5V, 4V and 3.3V. The selection of the voltage level depends in the application or requirement of the project. The 5V is supplied from the development board to the sun sensor as reference voltage, 5.58 V is supplied to Arduino Board which distributes 5V to the Bluetooth Module ,3.3V to the Inertial Measurement Unit. Although the temperature sensor was connected to Arduino through one-wire protocol, it is powered through the development board.

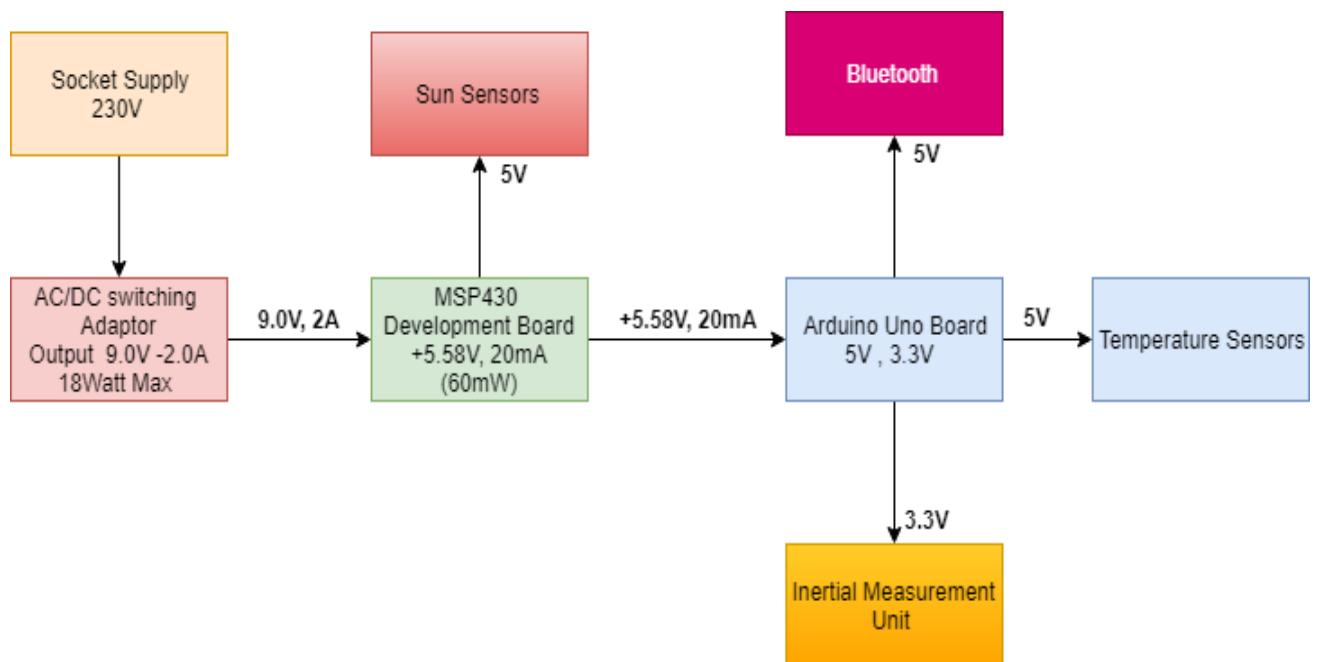


FIGURE 13: POWER MANAGEMENT DIAGRAM

3.2. Software Programming

To programme the development MSP430F2618 Texas instrument processor, the USB-Debug - interface MSP-FET420UIF was used to deploy the compiled programme from Code Composer studio Environment to the board. In order to setup the MSP430, the Grace software was utilised. Grace software offers a block programming interface which allows novices programmers to programme MSP430 and provide a better visualization of the entire programme, then the code composer studio compiler was used to complete the C to programme for the development board. The Arduino was programmed in C++ language at the Arduino Idle environment.

3.3. MSP430 Analogue to Digital Conversion

The MSP430f2618 has an ADC12 module that supports a rapid 12-bit (4095 samples steps) analogue to digital conversions with up to 200 ksps. The MSP430 ADC12 includes a 12-bit Successive-approximation-register (SAR) core, reference generator options, 16-word conversion, control buffer that permits the conversion and storage of 16 independent samples without CPU intervention. However, depending on the application, ADC12 offers the possibility of selecting different reference voltage levels. To illustrate the ADC sampling process, the ADC commenced by reading an analogue signal and sampled the signal, then returned a value representing the voltage at the certain sampling time. Figure 14 shows a diagram representation in Grace Software; the Grace helps to configure the registers of Texas instrument processors in an interactive approach using block programming. First, the ADC buffer is enabled by selecting the channels to read.

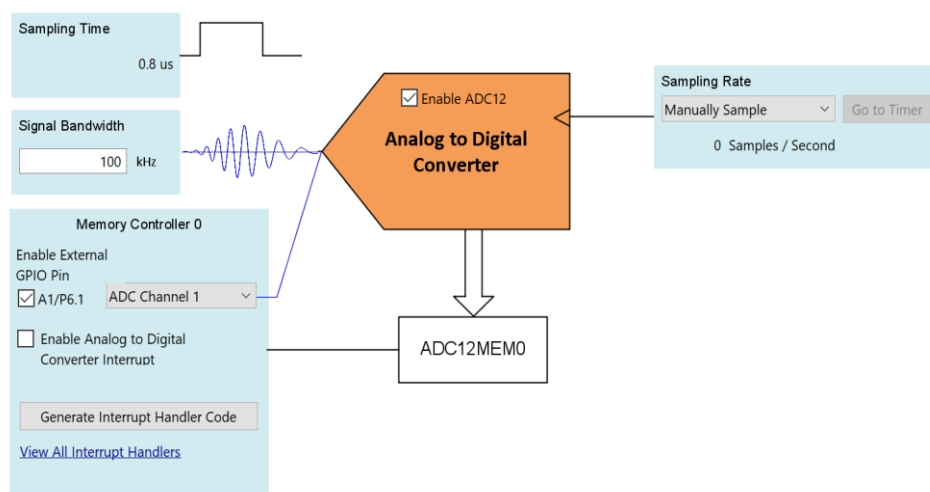


FIGURE 14: ADC BLOCK DIAGRAM SETUP IN GRACE

Three channels were enabled, channel 0, channel 1 and channel 2 as described in above section as pins P6.0, P6.1 and P6.2 from Port 6 in the MSP430F2618, these channels are directly connected to the analogue to digital converter memory register. This register stores the actual values read from each channel. The ADC Buffer is overwritten any time a new value arrives. Each channel signal was sampled at $0.8\mu\text{s}$ with bandwidth of 100 kHz. Figure 15 shows an amplified version of figure 14, as mentioned before; the ADC reference voltage offers a different voltage level such as 2.5V or 1.5V internal positive reference and external positive reference and system VCC.

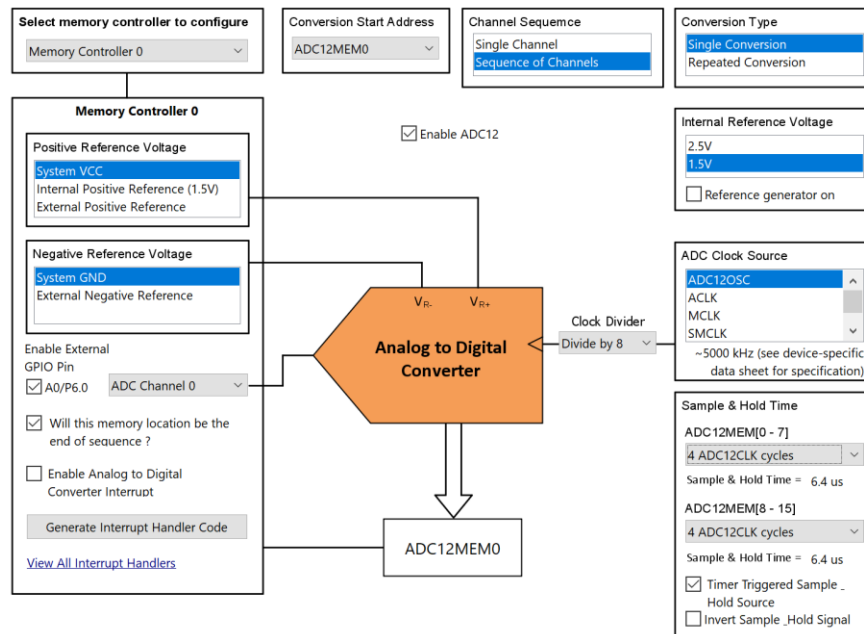


FIGURE 15: EXTENDED ADC BLOCK DIAGRAM IN GRACE

This project uses the System VCC reference voltage, which is 5V and connects the negative reference to the system ground that is common grounded together with other analogue grounds. The memory controller has to be selected and under the memory controller the reference voltage has to be set and the general-purpose input/ output (GPIO) pin has to be enabled in the code as input. The conversion started by addressing the ADC12MEM also known as the analogue to digital conversion memory register which is the main memory register for the ADC, it must be enabled, in order to read the analogue values. The system was programmed as a sequence of channels, since three channels were used, and a single conversion was achieved for each channel. The ADC12SC software triggers the ADC conversions and ADC interrupts which will raise the interrupt flag and begin the ADC conversions. After the conversion, the interrupt flag will be reset, and this permits the microprocessor to go into low power mode while executing other parts of the code.

3.4. Telemetry

In order for the satellite to communicate with the ground station (PC) a Bluetooth device was used to transfer data from the CubeSat to the ground station through UART communication. The Bluetooth device selected for this project was the HC-05 Bluetooth module, because it is an inexpensive device that is simple to interface with a PC and it was only used with a unique purpose of sending data to the PC. The device has six pins, transmit and receive pins described in figure 16 as RX and TX, the state pin indicates the pairing status of the Bluetooth device, the enable pin receives signal from the microprocessor to activate the Bluetooth.

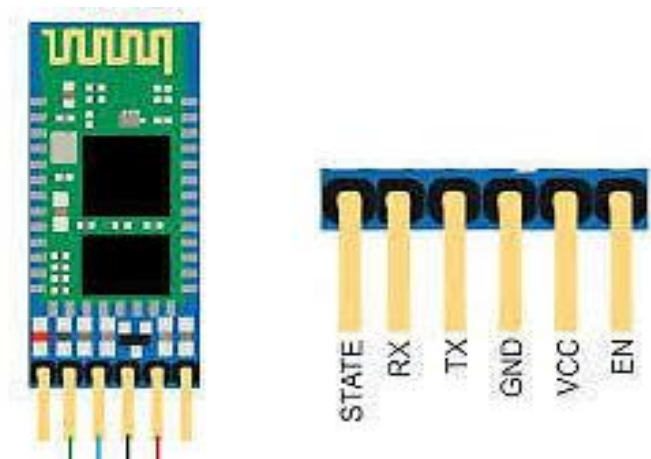


FIGURE 16: BLUETOOTH DEVICE

The device contains a switching mode between the master and slave mode this means the data can be wither transmitted or received. The figure 17 illustrates the connection of the Bluetooth with Arduino. Notably, TX of the Bluetooth needs to be connected to RX of Arduino and RX from Arduino to TX Bluetooth. The device is powered through the main source with 5V although, it can also operate with 3V. It is recommended to use a voltage divider between the receiver, transmitter and ground if the system will be running for long (Magdy, 2018).

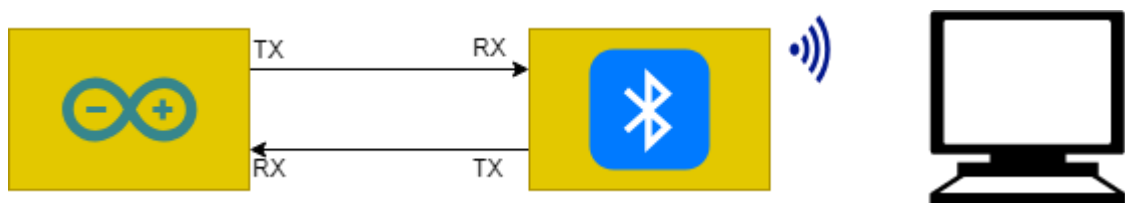


FIGURE 17: UART SETUP

3.5. Protocols of Communication

3.5.1. Inter-integrated circuit (I2C) Protocol

The development board was interfaced with Arduino through inter-integrated circuit (I2C) and to enhance the communication between the two boards strong pull-up resistors of $4.7\text{k}\Omega$ were connected to the bus(see figure 20) . The reason, I2C protocol was used for this project rather than other communication protocols was because I2C protocol has the advantage of connecting multiple slaves to a single master device or multiple master devices to a single slave. Additionally, it only uses two wires to transmit and receive the data which is an advantage for the CubeSat as it helps to constrain space into the CubeSat and develop an undemanding model with less wires.

The Serial data (SDA) is the wire on the bus responsible of receiving and transmitting data from the master to slave or slave to master. The Master, in this case the Arduino is responsible for providing the clock signal to the slave devices. The transmission is executed bit by bit along a single wire. The output of DB- MSP430 is synchronized to the sampling of bits by the clock signal from the Arduino board. Arduino offers three setting modes for the clock frequency, the standard mode is 100 kHz, the fast mode is 400kHz and some processor supports 10 kHz at low speed mode and 1000kHz or 3400kHz at high speed mode. The clock signal is generated by the master device. The DB-MSP430 sends the analogue to digital converted voltage to Arduino through I2C, the value transferred is divided into frames of data, each value sent has an address frame that contains the address of the slave device (DB-MSP430).



FIGURE 18: I2C INTERFACE BETWEEN BOARDS

When the starting bit is sent, the Serial data line (SDA) pulls the bus from high to low voltage level. Then the Arduino communicates with the DB-MSP430 and IMU sensor by sending the slave address for all the slave devices and the MSP430 check for the slave address sent from the Arduino. If the address corresponds to the slave DB-MSP430 or IMU sensor, the slave will send an acknowledgement bit to the Arduino board.

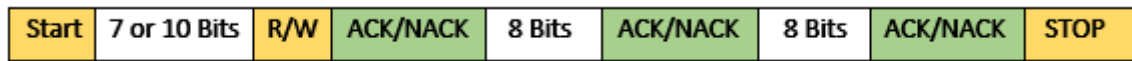


FIGURE 19: DATA FRAME

The address frame sends a single bit in return to notify that the DB-MSP430 requires to write or read from Arduino. The Arduino master requests data from the DB-MSP430. After the Arduino detects the acknowledge bit from the DB-MSP430, the DB-MSP430 sends the first data frame. The data frame is 8 bits long. However, the DB-MSP430 transmits 12 bits ADC value. The 12 bits are divided into two packages of data and sent each at a time to the Arduino board.

The IMU sensor sends each 16 bits of raw data from the accelerometer, gyroscope and magnetometer to Arduino. Both IMU sensor and DB-MSP430 data frame is instantly followed by an acknowledge bit or Non-acknowledge bit to confirm that the data frame has been collected successfully. After the DB-MSP430 and IMU Sensor finish sending all the data the Arduino sends the stop condition to the DB-MSP430 and IMU sensor to terminate the transmission. See figure 20, the two lines are pulled up with 4.7k Ω to the 5V system VCC.

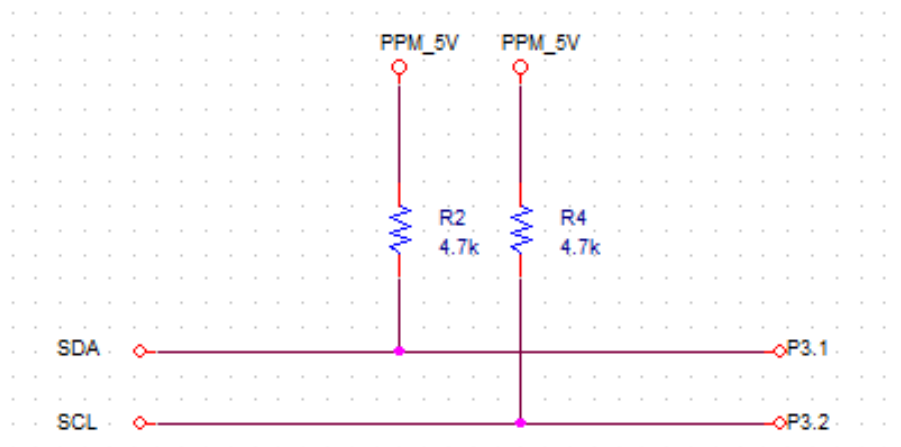


FIGURE 20: I2C BUS

3.5.2. One-Wire Protocol

The one-wire is a communication protocol that uses a single wire to transfer and receive data allowing the interaction between a microprocessor with many Maxim/Dallas semiconductor parts, those including temperature sensors, battery, thermal management devices, memory etc. The Dallas semiconductor one-wire communication protocol provides a solution for identification, memory, measurement and control. The one-wire is interfaced with single data with a ground reference, and it uses a single pull-up resistor to pull the bus at high state. The one-wire protocol utilises complementary metal – oxide semiconductor (CMOS) and transistor – transistor logic (TTL) logic and works with a supply voltage range of 2.8V to 6V (Maniyar, 2008). The data is only transferred at the single direction both masters and slaves' devices can read and write at the time slot. The master starts and control the flow of data and one-wire operations. The least significant bit is sent first and is transferred in time periods. One wire protocol does not require that the master device sends a clock cycle to the slave device since both transmitter and receiver devices are self-clocked and they will be synchronized by the falling edge of the master device.

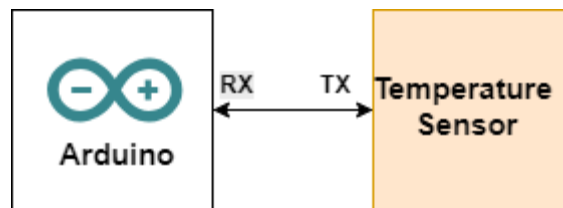


FIGURE 21: ONE-WIRE CONNECTION

One-wire communication has four setting operations, figure 22 shows the time diagram defined by the manufacturers: reset, write 0, write 1 and read bit. The frame byte can be derived using the four-bit operations. Figure 22, explain communication timing diagram, note that the data is sent by wide logic 0 and narrow logic 1 pulse widths at sample time. The timing diagram also describes the sample time to acquire a robust communication between master and slave device. The communication between the master and the slave starts with the first operation), the master sends a reset pulse to synchronize the bus. The slave device replies to the reset pulse by writing logic 0 pulse for 60 to 240 μ S then release the bus. The slave devices write to the bus when the master initialize by driving the one wire line to low then it holds the line wide pulse to send a logic 0 or release short pulse to permit the bus to return to the logic 1.

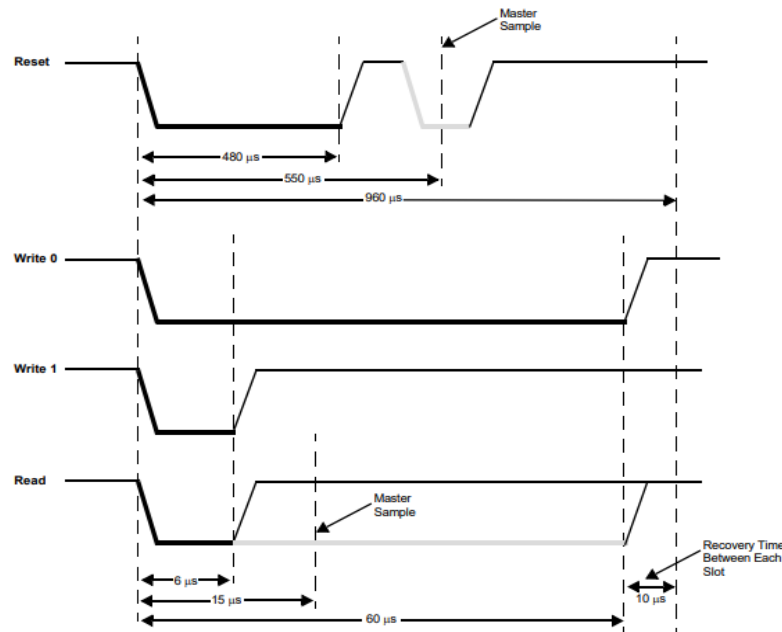


FIGURE 22: PRE-DEFINED SIGNAL

The master reads the frame data by driving the line wire with narrow low pulse. The slave device responds by returning logic 0 by holding the line low to enlarge the pulse period or by returning a logic 1. This protocol offers two sample rates operation mode. The standard speed of 15kbps and the higher speed of 111 kbps. As one wire is self-clocked, it accepts long bit delays, which guarantees a smoother operation. This protocol of communication was used to interface the temperature sensors with the Arduino board (see figure 21).

3.6. Sun Sensor Hardware

The sun sensors were designed using photodiodes with integrated transimpedance amplifiers. These photodiodes have high sensitivity to the light and can operate in various environment. The intent was to design a sun sensor with less financial assets while capable of producing functional results, that could be used later to control the reaction wheel and subsequently the orientation of the CubeSat. The photodiode employed for this project was OPT101 Monolithic photodiode with integrated single supply transimpedance amplifier from Texas instrument. This type of photodiode reduces the leakage current errors and noise. The light intensity is proportional to the output voltage. The integrated transimpedance amplifier can be supplied with single or dual power supply operation. The single supply takes an input from 2.7 to 36V. In this project, the photodiodes are supplied with the 5V from the MSP430 with low quiescent current of 120 μ A. The integrated chip uses 1M Ω internal feedback resistor. Using the internal feedback resistor, the bandwidth frequency is 14 kHz. This photodiode has high responsivity of 0.45A/W equivalent to 650nm. Figure 23 shows the schematic of the sun sensor designed in OrCAD software.

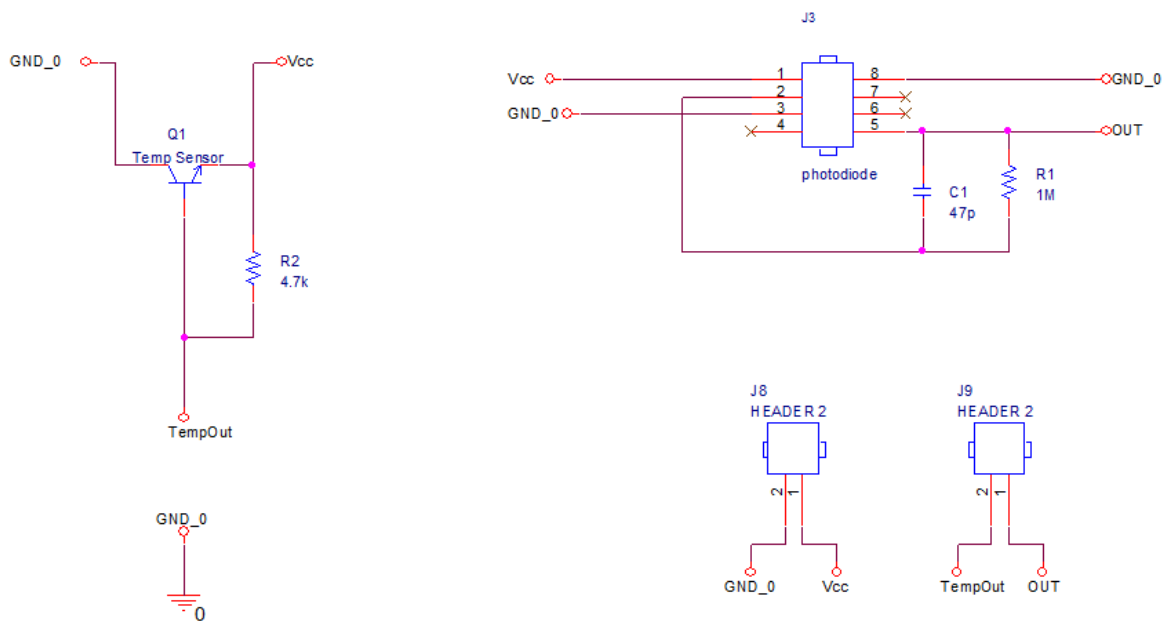


FIGURE 23: SUN SENSOR SCHEMATIC

The output voltage range from the transimpedance amplifier depends on the 1M Ω R1 resistor, the capacitor (C1) provides a smoother output voltage. Header J9 is the output sun sensor and temperature sensor. Header J8 receives the power and distribute 5 V to the sun sensor and 5V to the temperature sensor.

The transimpedance amplifier converts the input current into voltage. Figure 23 shows that the output voltage depends on the feedback resistor and the current produced by the photodiode. The photodiode with integrated transimpedance amplifier is competent to hold a constant voltage across the input source, whilst the input current varies due to the light falling in the sensitivity area of the photodiode. The area of sensitivity of the photodiode is 2.29mmx2.29mm. The external sensitive junction is protected with metallic enclosure. The area of light detection is relatively bigger than the shield around the photodiode. Note from Figure 24, if the sun simulator is directed to the small area, the light will fall on the sensitive area and It will produce an output proportional to the light falling in the photodiode.

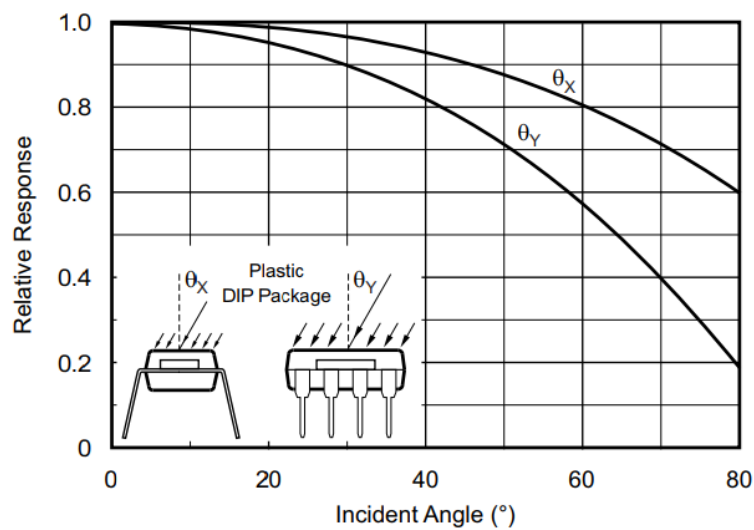


FIGURE 24: RELATIVE RESPONSE AGAINST INCIDENT ANGLE (TEXAS INSTRUMENTS, 2003)

Figure 24 shows that if the beam of light falling in the photodiode is focused purposely into the area of sensitivity, this will provide an accurate result compared to the results attained when the area of sensitivity is not illuminated. The light must fall on the sensitive area for the photodiode to perform accordingly. Figure 24, also shows, that at higher incident angle the light is deflected and dispersed. At lower incident angles, there is a loss in the sensitivity due to the smaller light effect on the photodiode which is proportional to the cosine of the angle.

3.6.1. Software diagram flow

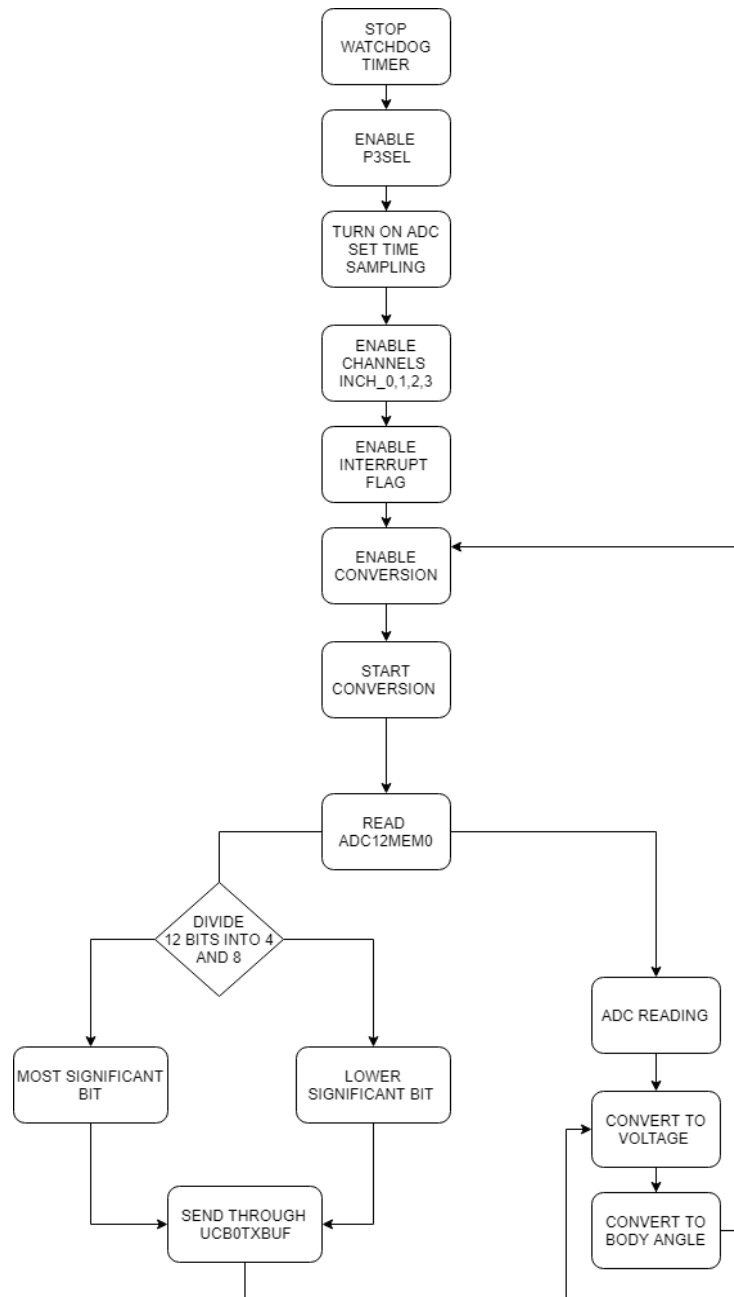


FIGURE 25: SOFTWARE DIAGRAM FOR THE SUN SENSOR

This diagram flow above represents the software programming developed for the sun sensor, note that as for any processors, the first task is to set the registers to be used. MSP430f2618 has a specific register for the ADC12 bit. The watchdog register is on by default, first the watchdog was disabled, this timer resets the microcontroller periodically unless the firmware triggers the timer to inform the processor that the programme is performing as expected. Second the ADC12 bit is configured using ADC12CTL0 and ADC12CTL1 (ADC Control register 0 and 1). The ADC12CTL0 controls the sample time (ADC12SHP) and multiple sample conversion (MSC).

The ADC12ON enables the ADC12bit which is off by default. Setting the ADC12ENC it enables the ADC conversion. To commence the conversion ADC12SC register must be set to high. After setting all the necessary registers, to read the ADC12, the channels must be open. The MSP430f2618 has 8 channels, four channels were opened to read the output from the sun sensor. Next, the ADC12 bit was divided into two packages, these values are sent to Arduino via I2C (UCB0TXBUF) at a time.

3.7. IMU Hardware and Software

As previously defined in the literatures review, inertial measurement unit is a device that incorporates three sensors. For this project, the IMU sensor was used to obtain the Euler angles that describe the orientation of the CubeSat by three angles. The IMU sensor utilised in this project was the Pmod NAV(see figure 26).

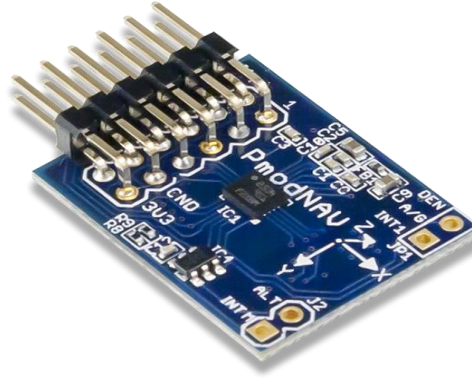


FIGURE 26: PMOD NAV IMU SENSOR

The Pmod NAV IMU uses the LSM9DS1 and LPS25HB chips. The LSM9DS1 includes three-axis gyroscope, three-axis accelerometer and three-axis magnetometer. The LPS25HB chip is a digital barometer which measures the atmospheric pressure, in total the Pmod Nav IMU provides 10axis. Pmod NAV IMU provides each 16-bit full scale registers for the magnetometer, accelerometer and gyroscope. This device also offers different ranges of sensitivities. For this project, it was used the default sensitivities range due to its application for CubeSat does not require high sensitivity.

table 1: sensitivity ranges

Linear Acceleration full scale (g)	Angular rate full scale (dps)	Magnetic full scale (Gauss)
±2	±245	±4
±4	±500	±8
±8	±2000	±12
±16		±16

The IMU sensor is connected to the Arduino board through I2C protocol. First, the raw data was read, subsequently, due to noise and bias error, a low pass filter was used and for further filtering a Kalman filter was also used to filter the pitch, roll and heading angles. The block diagram below describes how the orientation of the CubeSat was calculated using IMU. The 9 axis IMU sensor is fundamentally efficient for dynamic orientation calculation for either long or short period. The Euler angles can be designated as roll, pitch and heading, refer to Section 2.4. The absolute orientation of the CubeSat is obtained after Kalman filter.

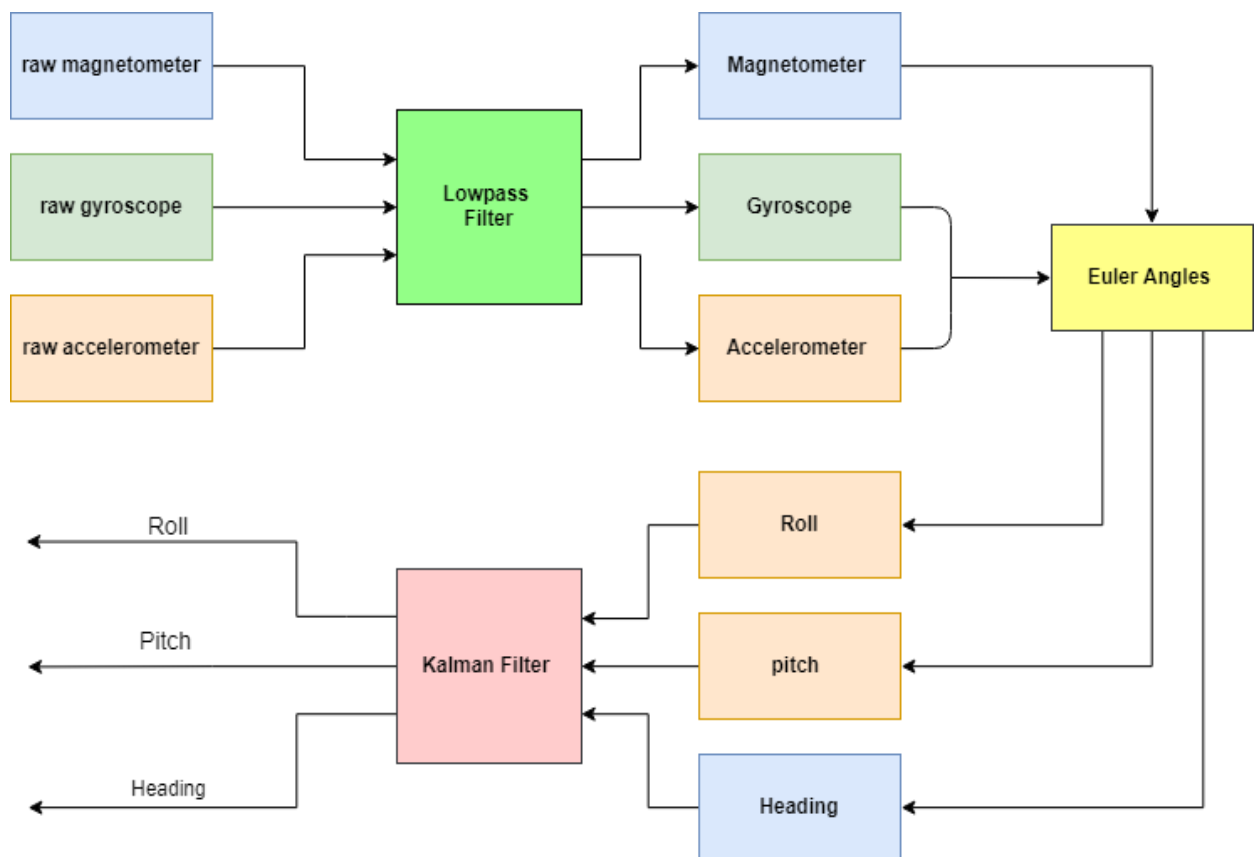


FIGURE 27: BLOCK DIAGRAM FOR THE IMU SUBSYSTEM

3.7.1. Software diagram flow

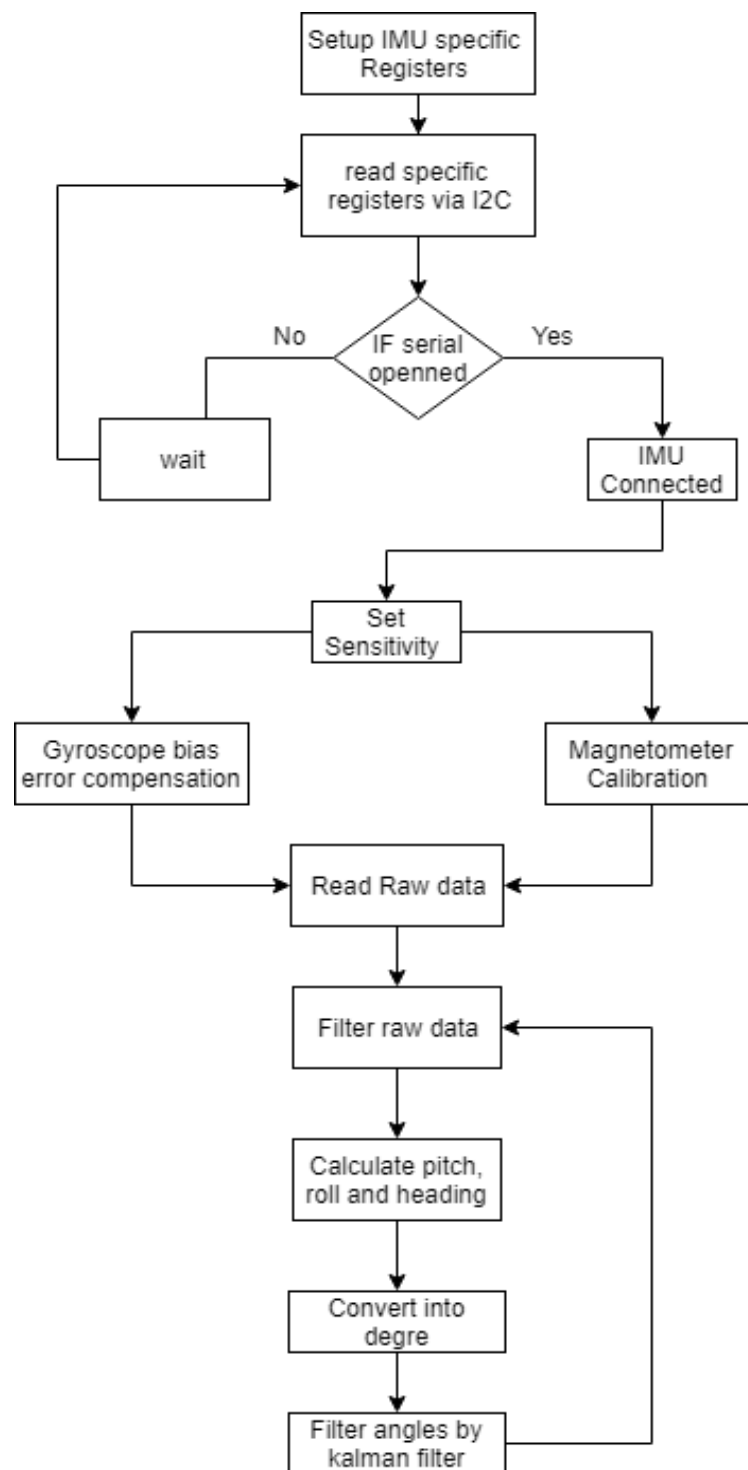


FIGURE 28: SOFTWARE DIAGRAM FLOW FOR IMU SENSOR PROGRAMMING

The first step was to set the specific registers in order to obtain the raw data from the IMU sensor. The IMU allows the communication with microcontroller via different protocol such as I2C, SPI and UART. However, for this project the IMU sensor was interfaced with Arduino board through I2C protocol (see Section 3.5).

The program was designed to check every time if the IMU is connected to the Arduino board. Next, the sensitivity range device was set. For this project, the sensitivity range is the default range: sensitivity range of the gyroscope was configured to $\pm 245\text{dps}$, accelerometer was configured to $\pm 2\text{G}$ and magnetometer sensitivity is $\pm 4\text{Gauss}$.

Two calibrations are performed for the magnetometer: soft iron and hard iron calibration. The hard iron calibration removes the frequent magnetic field which disturbs the stability of the sensor. Whereas, the soft iron calibration reduces the effect of electromagnetic field caused by some devices around the IMU sensor, this calibration depends on the environment of operation. In the space lab, there are different devices that can interfere with magnetic field produced by the magnetometer, which makes difficult to reduce the EM field effect. Section (2.2.2), the declination angle in Glasgow was also took into consideration the reduction of the angles from magnetometer by 2.6°N . A low-pass filter was used to filter the raw data. Subsequently, the Euler angles were calculated using Euler algorithm (Section 2.4) and to enhance the data, a Kalman filter was used which reduced the noise at each angle.

3.8. IIR Digital Control

The low-pass filter was applied to smooth out the high frequency signals in the raw readings from IMU. The digital low pass filter implemented was a first order recursive infinite impulse response (IIR) filter. Figure 29 shows that by changing gain a the output response of the filter changes. For this project a was 0.3.

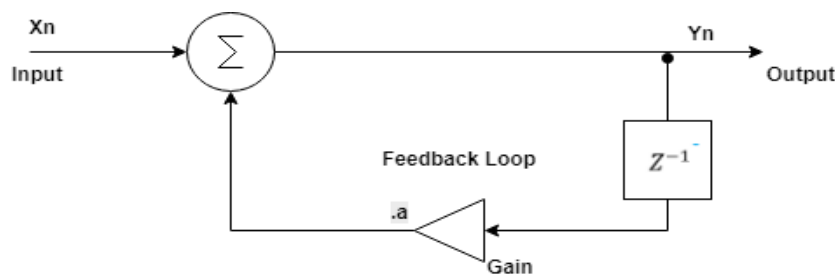


FIGURE 29: LOWPASS FILTER

The filtering algorithm used to implement this filter was designed based on the equations below. From Equation 48, a is the gain, once changing this gain the sensitivity of the filter changes correspondingly.

$$\text{Current output} = \text{Current input} + (1 - a) \times \text{Previous output}$$

(47)

3.9. Conclusion

This chapter intended to describe the hardware both designed and used to elaborate this project, there were certain challenges associated with the PCB fabricating machine, as it was not printing effectively. To overcome this issue, some of the trucks were re-soldered. Another challenge was associated with the PPM connection (*Appendix 1*), because each components footprint had to be designed for this specific project, thus some challenges were faced to connect the attitude determination module with the development board, directly. To resolve this problem, an extension connector was used to effortlessly connect the attitude determination module with the development Board. Overall, all the hardware was successfully designed, and operated as expected.

CHAPTER 4

Section 4: Methodology and Results

4. Experiment description

The figure 30 shows the experiment apparatus setup. The experiment was performed in James watt south building during the night, this is because in the afternoon the photodiodes is limited by glint and albedo affect. All the lights were turned off to avoid any other light to fall into the sun sensors, except the one from the sun simulator. Note as the sun simulator is extremely powerful for these sun sensors, to obtain a concise and effective results the sun simulator had to be placed at a certain distance of the CubeSat at least greater than 2m. The CubeSat uses three sun sensors. Each of them was placed with the same orientation as the IMU sensor at three orthogonal sides of the CubeSat. This is a new approach of looking into the orientation. The idea was to match the orientation of the sun sensors with the orientation of the IMU sensor. The CubeSat was placed at the same level as the light so that the light could be pointing directly to the CubeSat. Allowing that in the initial state, one of the sun sensors goes full blast which corresponds to unit vector as the body angle of this sun sensor will be 90° .



FIGURE 30: EXPERIMENT APPARATUS SETUP

By placing a single sensor at each side of the CubeSat, it permits an algorithm for the 3D orientation to be developed depending on the light source falling on the sensor (section 2.2). Next, the development board was powered through the 230V socket supplier (section 3.1), this allows the CubeSat to operate independently. The ground station (PC) was placed 1m away from the setup and the data was logged through a serial connection with Bluetooth. The Bluetooth has red LED for on/off state, LED blinks when the Bluetooth is connected to the board. The Bluetooth model requires a password to pair with the Bluetooth of the ground station, the default password is 1234. The data can be logged from any software capable of receiving serial data and plot the data in real time. This project uses the “serialPlot” software suggested by Dr. Matteo Ceriotti, to plot real time data in higher rate compared for instance PLX DAQ. Once, all apparatus is properly set, the data was logged into the PC showing different channels in Serialplot. As the project has not yet reached its final state to perform changes in orientation, the CubeSat was moved manually.

4.1. Sun sensor -Results and Discussion

The graphs below show the analogue to digital converted values. The 12 bits ADC converts the analogue value into digital readings. Each graph represents the 3-dimensional axis of the CubeSat frame. Figures 31, 32 and 33 show the ADC values when the sun simulator was illuminating the CubeSat. Note that from the interval of 0 to 100th sample, the sun sensor at the y-axis was responding to the ambient light, although the experiment was performed in a dark room, there was some ambient light in the room. See Figure 32, after 100 samples, the ADC reads 4000 resolution, which suggests that CubeSat was rotated and the sun sensor at the y-axis was pointing directly to the sun simulator. On the other hand, the sun sensor at the X-axis shows a resolution of 1000 from 0 to 200th sample, and the sun sensor at the z-axis shows 3000 resolution between 0 to 200th sample which means that there was light falling at the sensitive area of the sun sensor at the Y-axis. From 200 to roughly 300th sample, the ADC resolution at the sun sensor at X- axis and Z-axis estimate 4000 ADC value, while the sun sensor at the Y-axis shows a resolution of round 1000, this indicates that greater light was falling at the X-axis and Z-axis than at the Y-axis. Note from figure 31 and figure 32 that between 500 to 600th sample, the sun sensors at the X and Y-axis shows an ADC value less than 1000 resolution. This indicates that the photodiodes captured an ADC value below the one of ambient light (about 900 resolution). This suggests, that these sensors were instantaneously covered, therefore the light was blocked to fall at the surface of the photodiodes.

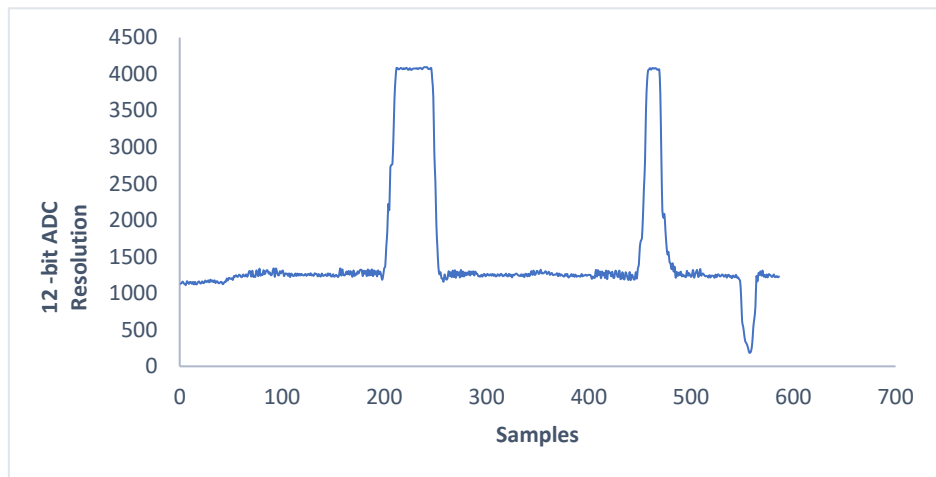


FIGURE 31: SUN SENSOR AT THE X-AXIS

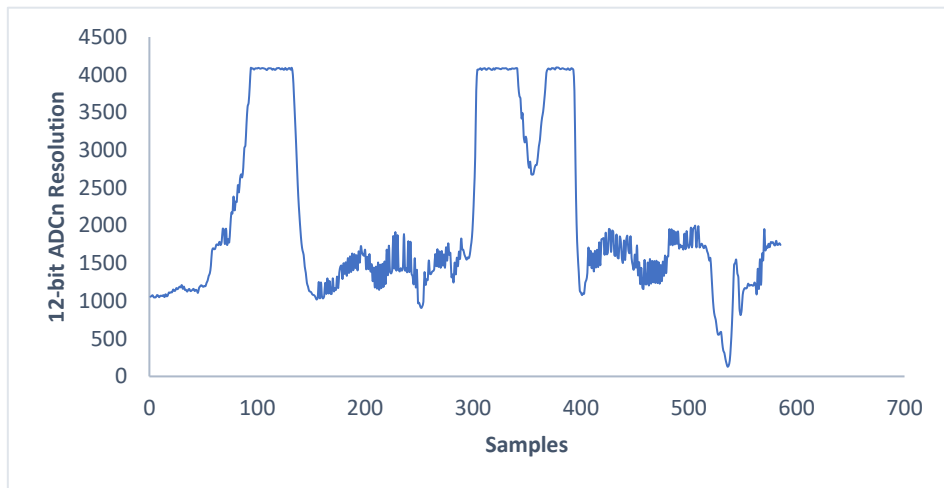


FIGURE 32: SUN SENSOR AT THE Y-AXIS

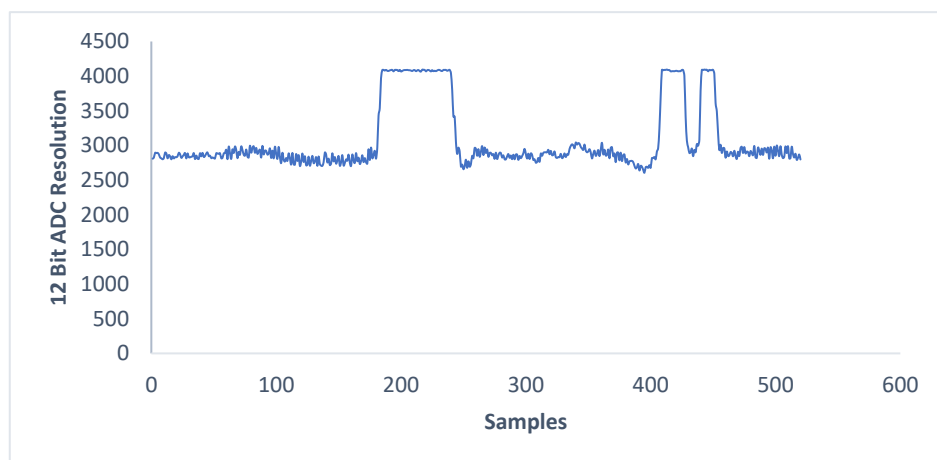


FIGURE 33: SUN SENSOR AT THE Z-AXIS

4.2. ADC 12 bit to Voltage

4.2.1. Methodology

Equation 48 is used to convert the ADC value into voltage, the output voltage ranges from 0 to 5V. The ADC value is a ratio-metric value, this means that the 5V is equivalent to 4095 resolution of ADC.

$$\text{Analogue voltage Measured} = \frac{\text{ADC reading} \times \text{System voltage}}{\text{Resolution of the ADC}}$$

(48)

Note that at 5V, the device reached the maximum resolution of 4095. Therefore, if the sun simulator directs towards the sun sensor, the device will output 5V. In the absence of the light, the device will output approximately 0V. The method applied to acquire the voltages was the same as the previous section, the CubeSat was moved while the sun simulator was maintained at a fixed location.

4.2.2. Results and Discussion

Figure 34 represents the voltage acquired from the sun sensor at the X-axis, Figure 35 is the voltage at the Y-axis and Figure 36 is the voltage at the z-axis. Figure 34 shows that from 0 to 400th sample, the sun sensor at the X-axis outputs about 3V, while Figure 35 shows that the sun sensors at the Y- axis is about 1V until 300 samples and reaches 5V at 400th sample. Note that in this sample interval, the sun sensor at the Z-axis shows roughly 1V for the first 100 samples and from about 150 to 300 samples the sun sensor outputs the maximum voltage 5V. This suggests that the sun sensor at z-axis was exposed to the light at this sample interval. From 400 to 500th sample, the sensor at X-axis outputs roughly 5V, which indicates that at this sample interval, the CubeSat was rotated. In consequence of this, the sensor at the X-axis was pointing towards the sun simulator. At this sample interval, the sensor at Y-axis is 1V and at z-axis is roughly 1.3V. From 500 to 600th sample, the sun sensor at X-axis shows approximately 0V. This suggests that both ambient light and the light from sun simulator were blocked.

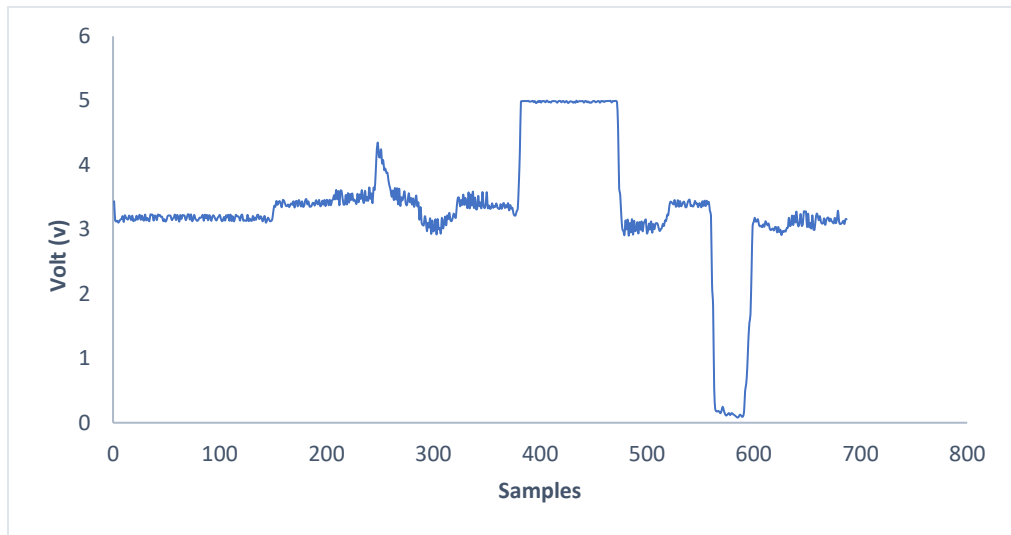


FIGURE 34: VOLTAGE OF THE SUN SENSOR AT THE X-AXIS

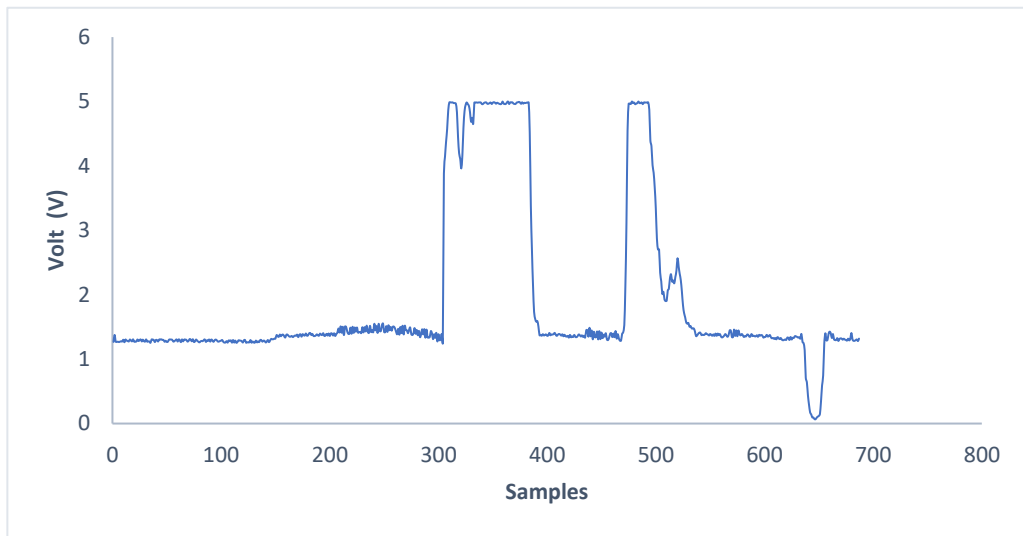


FIGURE 35: VOLTAGE OF THE SUN SENSOR AT THE Y-AXIS

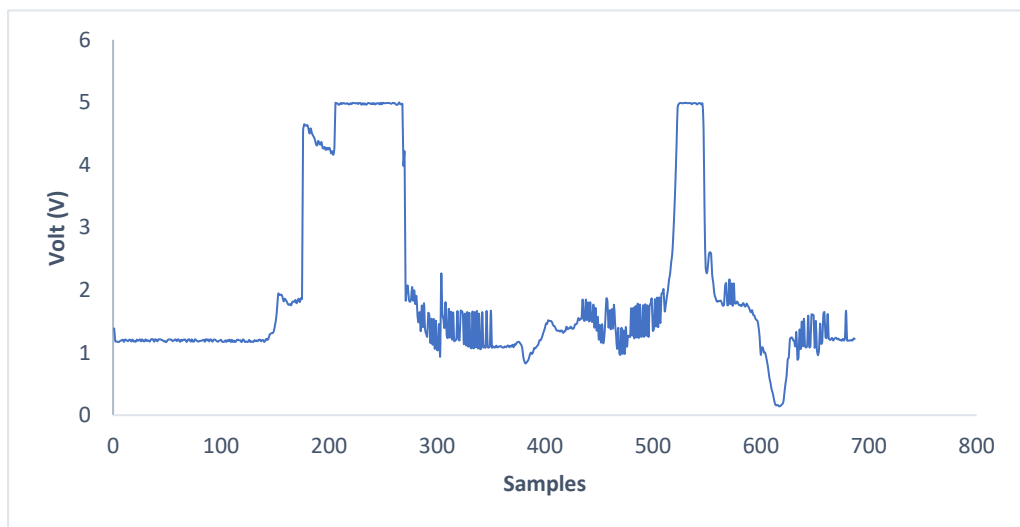


FIGURE 36: VOLTAGE OF THE SUN SENSOR AT THE Z-AXIS

4.3. Body Angles

4.3.1. Methodology

As aforementioned the sun sensors were positioned at the same direction as the IMU axis. Notably, the body angle is associated with the motion of the CubeSat (recall in Section 2.1.2 and 2.2.1). As the CubeSat moves around the sun simulator, different angles could be obtained. Moreover, if the sun sensor is pointing towards the sun simulator the body angle will be 90°. Thence, the body angle is limited by -90° to 90° with 180° field of view. The body angles were calculated using Equation (8) (section 2.2.1).

4.3.2. Result and Discussion

Figure 37 illustrates the body angles from the sun sensor at the x, y and z-axis direction. To note that the body angle varies with light falling at the sensitive area of the sun sensor (section 2.2.1). To analyse the body angles of the CubeSat with respect to a fixed frame (sun simulator), about 140 samples were taken. Figure 37 demonstrates that from 0 to 40th sample the body angle of the sensor located at the x-axis direction of the CubeSat is about 15° while the angle obtained from the sensor at the Y-axis varies between 37° to 48° for 140 samples. This suggests that as data was recorded, the Sun sensor at Y-axis was not entirely exposed to the light, and the CubeSat was rotated such that the body angles at Y- direction was constant with respect to the sun simulator. At this sample interval, the body angle at the Z-axis is also about 15°. From 40 to 60th sample, the sensor at z-axis increases gradually until it reaches roughly 84°, while the sensor at the X-axis is 15° until 60th sample and start varying from 60 to 100th sample. This suggest that the CubeSat was moved forward and back at the Y-axis direction. From 100 to 140th sample, the figure 37 illustrates that the body angles at the Z-axis vary as the light falls at the sensor at Z-axis while the body angle at x-axis is 15°. To make the body angles information useful, the sun vector must be calculated so that more information about the orientation of the CubeSat can be found. This can be obtained using Equation 18. Table 2 show the sun or solar vector of some body angles at certain direction. See the sun vectors at certain body angles below:

$$\vec{S} = (0.31i \quad 0.79j \quad 0.98k), \vec{S} = (0.12i \quad 0.78j \quad 0.90k), \vec{S} = (0.95i \quad 0.76j \quad 0.06k)$$

Table 2: Body angles and sun vectors

Body angles X-axis (degree)	Sun vectors At x-axis	Body angles y-axis (degree)	Sun vectors At y axis	Body angles z-axis (degree)	Sun vectors At z-axis
71.35	0.31	37.59	0.79	10.80	0.98
82.94	0.12	38.06	0.78	24.69	0.90
17.58	0.95	40.47	0.76	86.41	0.06

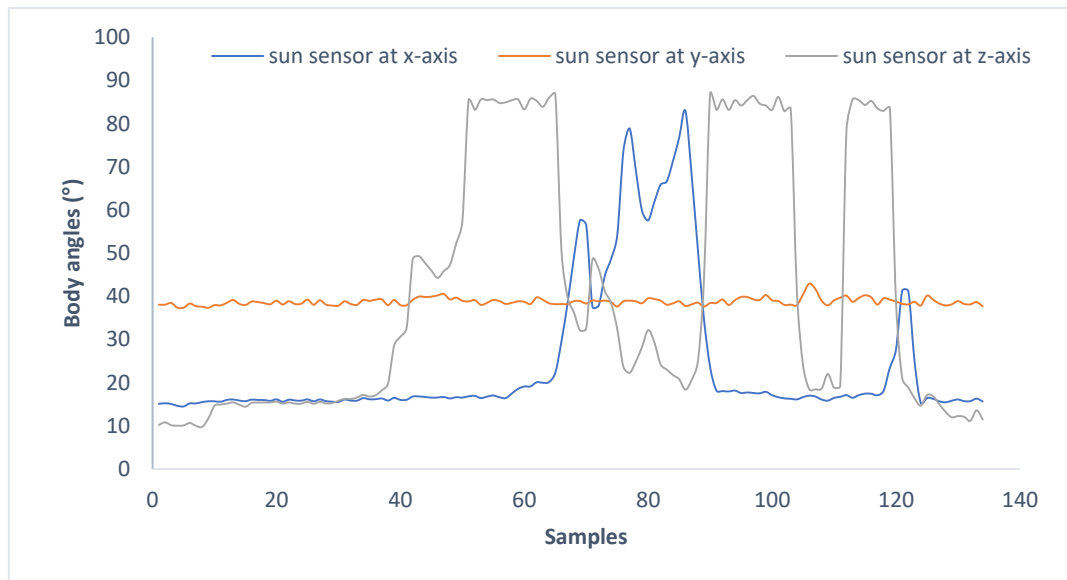


FIGURE 37: BODY ANGLES OF THE CUBESAT AS IT ROTATES AROUND THE SUN SIMULATOR

4.4. IMU sensor

4.4.1. Methodology

To test the IMU Subsystem, the CubeSat was manually moved whilst the data was recorded. To ensure that the data was correct an iOS mobile app “sensorKinetics” was used to compare the angles of the CubeSat. The mobile phone was rotated at the same direction as the CubeSat.

4.4.2. Results and Discussion

The graphs below illustrate the raw data and the data filtered by the low-pass filter⁴. The noise was generated as the CubeSat moved around. However, a low-pass filter was implemented to smooth out the high frequencies by reducing unwanted noise in the raw data. Additionally, by increasing the gain a , a smoother output can be acquired.

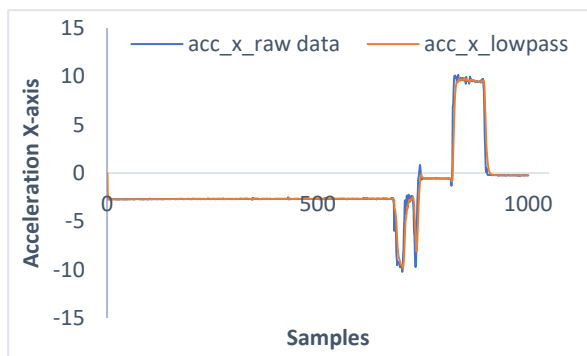


FIGURE 40: ACCELERATION AT X-AXIS

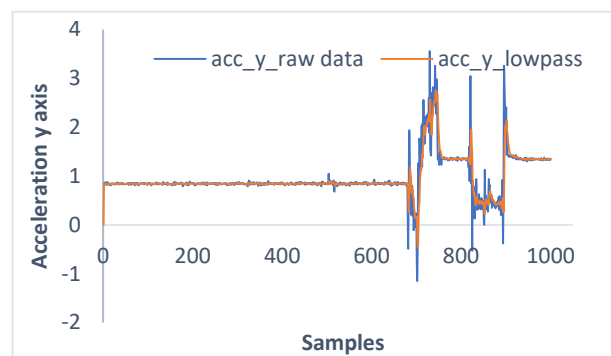


FIGURE 39 : ACCELERATION AT Y-AXIS

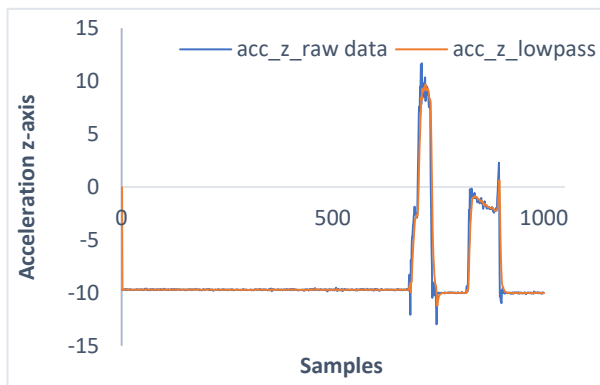


FIGURE 41: ACCELERATION AT Z-AXIS

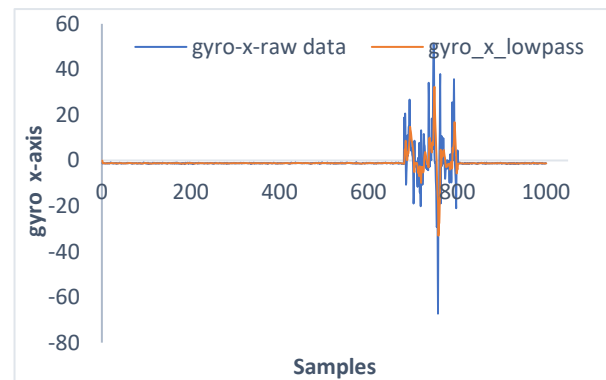


FIGURE 38:GYROSCOPE AT X-AXIS

⁴ By increasing the gain a to smooth the signal.

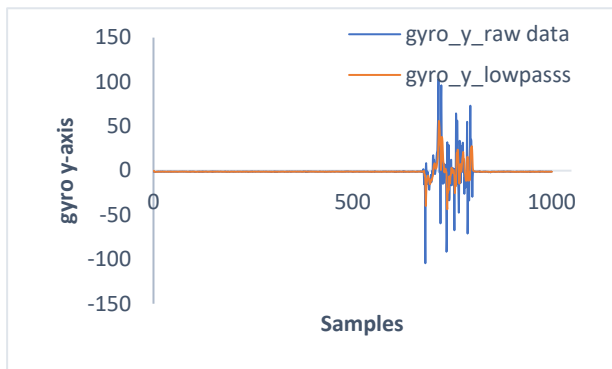


FIGURE 43:GYROSCOPE Y-AXIS

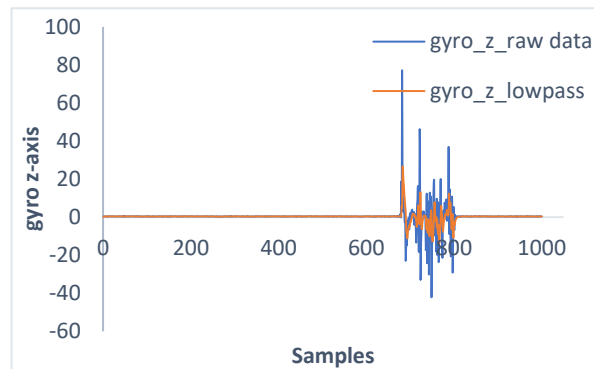


FIGURE 42: GYROSCOPE Z-AXIS

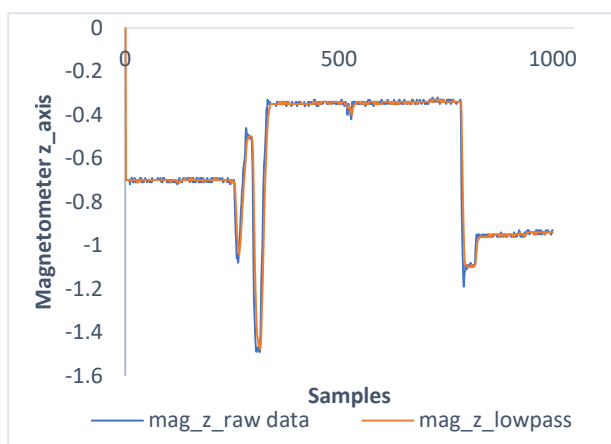


FIGURE 45: MAGNETOMETER Z-AXIS

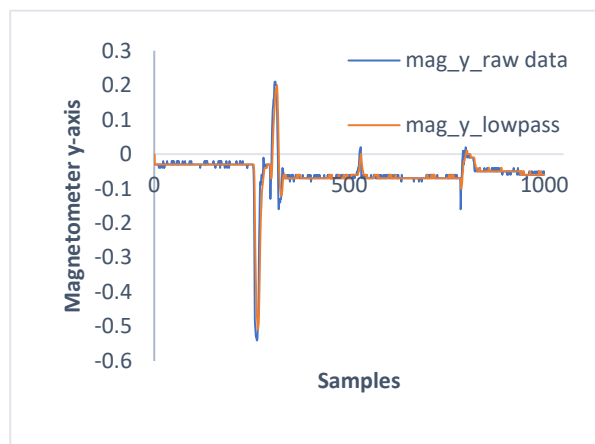


FIGURE 44: MAGNETOMETER Y-AXIS

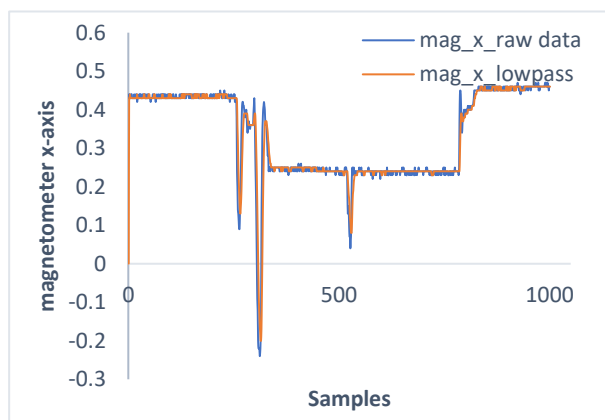


FIGURE 46: MAGNETOMETER X-AXIS

4.4.3. Result and Discussion - Euler Angles

After the data was obtained, the Euler angle was calculated and plotted (see Figure 47) . For the first 200 samples, the three angles are approximated to 0° , which indicates that the CubeSat remained unmoved at the reference frame, where no rotation motion was presented around the axis. From 200 to 400th sample, the roll angle is about 90° , this suggest that the CubeSat was gradually rolled towards 90° . At this sample interval, the pitch angle remains roughly 0° , the heading at this sample interval was estimated at -150° from the magnetic north pole, which was expected once compared with iOS mobile app sensorKinetics. From 400 to 600th sample, the three angles are 0° , which means that the CubeSat was rolled back at the reference fixed. From 600 to 800th sample, the roll angle is -90° while pitch remains 0° and the heading angle goes to 0° , which is an expected behaviour. This suggest that the CubeSat was rolled at inverse direction from initially.

From 1200 to 1400th sample, the roll angle is roughly 0° , while the pitch angle is estimated at -80° , which suggests that at this sample interval, the CubeSat was tilted -80° . The roll angle was expected to remain around 0° and this was confirmed from figure 47, the heading angle is roughly 50° from the magnetic north pole. From 1400 to 1600th sample, the three angles are estimated at 0° , this indicates that the CubeSat was pitched back to the reference frame at this sample interval. From 1600 to 1800th sample, the pitch angle is roughly 81° , at the opposite direction from previous rotation. It can be noticed that at this sample interval, the roll angle reading was affected by the pitch. This phenomenon is called the Gimbal lock affect. This happens due to the Euler angles algorithm Equation (30-32) (see section 2.4) When pitch angle approximates to 90° the gimbal lock will occur. For this project the gimbal lock will not be a problem as this is a lab-based project and the CubeSat will not be pitched higher than 90° .

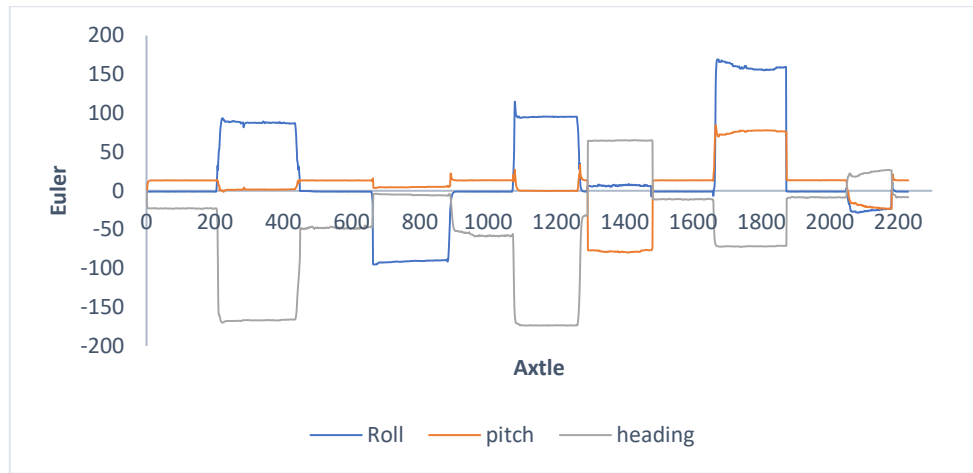


FIGURE 47: EULER ANGLES

4.4.4. Results and Discussion - Kalman filter

The Kalman filter used was a C library from Kristian Sloth Lauszus (Lauszus, 2012) which is available on GitHub. This library was utilised to reduce the complexity of the main program, whilst easing the difficulty of acquiring the filtered roll, pitch, heading angles. The default accelerometer variance is 0.001 rad and the variance of bias is 0.003 rad, the variance of measurement is set to be also 0.003 radian by default (refer to section 2.5). These gains can be tuned to improve the filter output. For this project, the default gain was used because they perform better. It was assumed that the bias based upon the accelerometer and gyroscope is 0 rad and also presumed that the initial state of the filter is 0 rad, so the error covariance matrix can be algorithmically described as:

```
P[0][0] = 0.0f;
P[0][1] = 0.0f;
P[1][0] = 0.0f;
P[1][1] = 0.0f;
```

The updated estimation error covariance matrix is:

```
P[0][0] += sample time * (sample time * P[1][1] - P[0][1] - P[1][0] + Q_angle);
P[0][1] -= Sample time * P[1][1];
P[1][0] -= sample time * P[1][1];
P[1][1] += Q_bias * Sample time;
```

The first step was to save the initial angles at the time when IMU starts operating for the first time. The Kalman filter will take these angles as the previous estimate, so the priori estimate will depend on the previous estimate. The Kalman filter will display the data of each angles in a range of -90° to 90° . This can be changed if wider range was required. Figures 48,49 and 50 show the plot of the pitch, roll and heading angles after the Kalman filter.

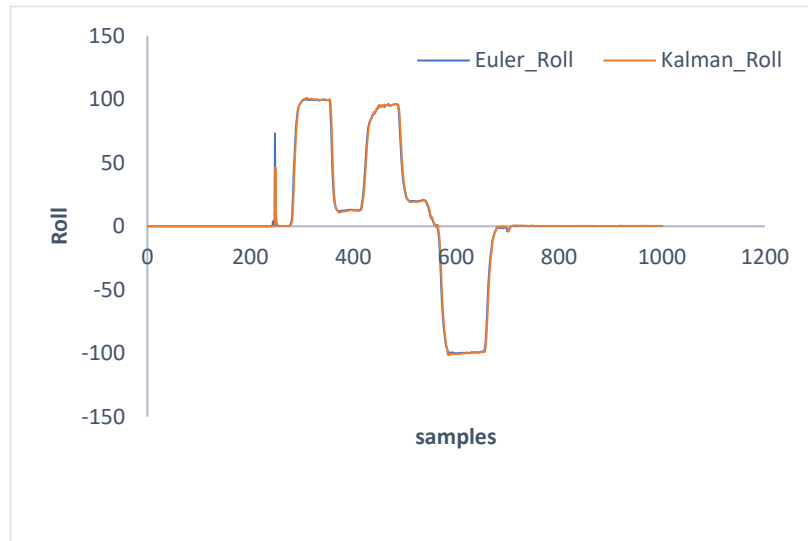


FIGURE 48: ROLL ANGLES

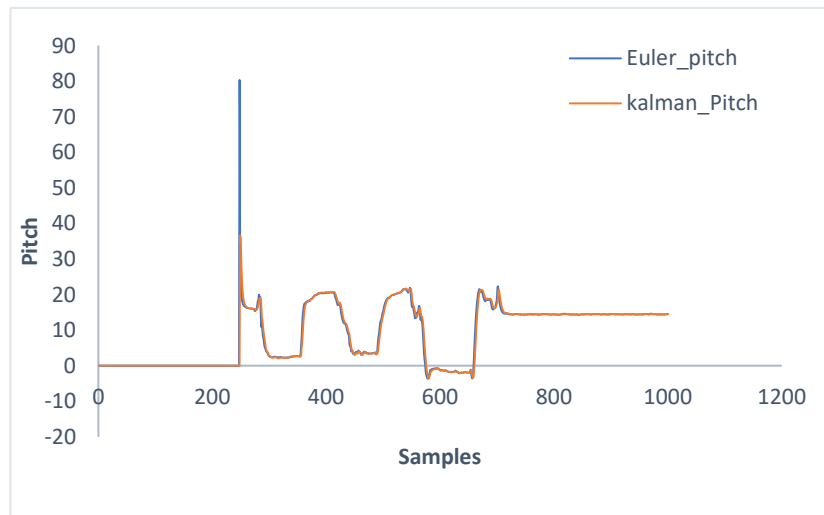


FIGURE 49: PITCH ANGLES

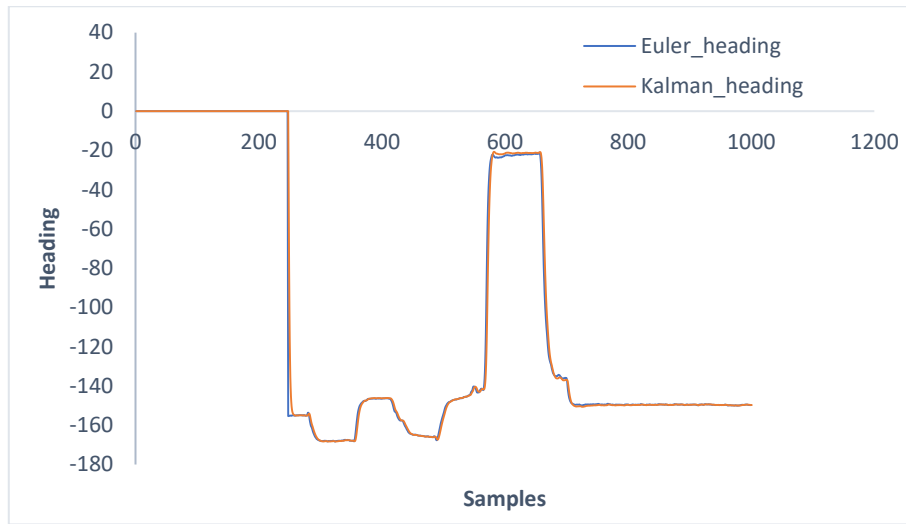


FIGURE 50: HEADING ANGLES

4.5. Conclusion

This chapter discussed the methodology used to test the attitude determination system design and developed at the space lab and follow its results. For the sun sensors, the hardware and software operate as expected with certain constraints in the light source and environment implemented, as the results are reliable and precise in a dark room. Overall, the sun sensor is highly sensitive to the light source and able of producing accurate body angles as it rotates around the sun simulator or the other way around. A sun simulator is extremely powerful for this experiment because the CubeSat needs to be at a certain distance apart from the sun simulator, otherwise, the photodiodes would reach full blast instantaneously. The inertial measurement unit sensor attached onto the CubeSat (*Appendix 1*). It was seen throughout the experiment that, the IMU performs as expected. Both low-pass and Kalman filter were, well implemented. The Kalman filter can be improved by changing the variance of the bias gain (Q_{angle}), accelerometer variance (Q_{bias}) gain. Certain small discrepancy at the Euler angles was detected, due to hand-shaking as the CubeSat was moved around manually.

CHAPTER 5

5. Overall Conclusion and Future work

The aim of this project was to develop a subsystem that determined the attitude of the CubeSat. The requirement was to succeed in the attitude determination system using the pumpkin kit CubeSat. Throughout the project, hardware was designed and developed to suit the requirement. The project utilized two sensors to perform the attitude determination: sun sensor and IMU sensor. The sun sensors were used to measure the sun's vectors from the CubeSat in relation to the reference fixed frame of the sun simulator. Three sun sensors were placed at three sides of the CubeSat representing x, y, and z axis. The body angles were calculated with respects to the light falling at the sun sensors surfaces. The results are shown in Chapter 4, overall the sun sensor operates without any issues and the C algorithm in the development board works computationally fast. The IMU was used to calculate the pitch, roll and heading angles. Section 4.5 describes the process of acquiring the data from the IMU sensor, in general the IMU sensor performs effectively. The orientations obtained from these sensors are the absolute orientation. Chapter 4 shows the plot of these angles, and to enhance the data a Kalman filter was used.

5.1. Future work

For further work, an experiment will be conducted by fusing the sun sensor with magnetometer or gyroscope. To perform this experiment, the CubeSat had to be rotated by hand. However, future investment has been made for this project and this allows to implement the Control part of this project. Therefore, Four DC Brushless motors have been bought, which will be used as CubeWheels to perform the rotation of the CubeSat, a proper test-bench will be required in order to perform a relevant experiment on the orientation of the CubeSat. The light source used to simulate the sun was overpowering for the sun sensor and this affects the sun sensor as the photodiode used was sensitive to the light. The photodiode sensitivity can be attenuated by reducing the field of view of the sensor. For future work, a CAD model to prevent all the light falling on the surface of the sun sensor must be designed. Other approach could be to purchase another linear photodiode (Hamamatsu S3932 IR Photodiode) which is expensive but with less sensitivity than the OPT101. Overall, the next step of this project is to use the absolute angles to control the CubeSat.

6. References

- Luka, M., Atayero, A., Kwaghdoo, O. & Alatishe, A., 2011. Satellite Communications: Impact on Developing Economies. *Journal of Computing*, 2(11), p. 579.
- Smitha, M. W. et al., 2010. Detecting transiting exoplanets using a low-cost CubeSat platform. *Proc. of SPIE*, 7731(773127), pp. 1-14.
- Acharya, R., 2014. *Understanding Satellite Navigation*. 1st ed. New York: ELSEVIER.
- Alemayehu , A. K. & Rezene, . E. . L., 2018. Finite Element Analysis of 3U CubeSat. *International Journal of Scientific & Engineering Research*, 9(6), pp. 1-7.
- Aliabadi, D. E. & Sharbafi, M. A., 2010. *MRL Team Description 2010 - Small Size Robot League*. Qazvin, ResearchGate.
- Baker , D. & Worden, S., 2008. The Large benefits of small satellite missions. *Eos Trans. AGU*, 89(33), pp. 1-6.
- Bender, E., 2011. *An Analysis of Stabilizing 3U CubeSats Using Gravity Gradient Techniques and a Low Power Reaction Wheel* , San Luis Obispo: The Faculty of the Aerospace Engineering Department .
- Bezold, M., 2013. *An Attitude Determination system with MEMS gyroscope drift compensation for small Satellites*, Kentucky: University of Kentucky Uknowledge.
- Bido, J. I., 2017. *System Study of an On-Orbit Additive Manufacturing CubeSat*, Florida: Florida Institute of Technology.
- Bona, B. & Canuto, E., 2002. *Dynamic Model of the Spacecraft Position and Attitude* , Torino: Dipartimento di Automatica e Informatica, Politecnico di Torino.
- Bugryniec, P., 2016. *CubeSat: The Need for More Power to Realise Telecommunications*, Sheffield: University of Sheffield.
- Carmona, Á. E., 2017. *Sun sensor for small satellite Attitude determination system*, Barcelona: Universitat Politècnica de Catalunya.
- Chaudhry, A., Shih, C., Skillin, A. & Witcpalek, D., 2018. *Inertial Measurement Units*, s.l.: s.n.
- Chin, J. et al., 2017. *Basic Concepts and Processes for First-Time CubeSat Developers*, San Luis Obispo : CubeSat 101.
- Cote, K. et al., 2011. *Mechanical, Power, and Propulsion Subsystem Design*, Worcester: Worcester Polytecnic Institute.
- Croce, N., 2005. *Newton and The Three Laws of Motion*. 1st ed. New York: The rosen publishing group, Inc.

Crook , M. & Sakoda , . D., 2009 . *NSP CubeSat Launcher Design, Process and requirement* , California: Naval Postgraduate School .

Crowell, C. W., 2011. *Development and Analysis of a Small Satellite Attitude Determination and Control System Testbed*, Massachusetts: Department of Aeronautics and Astronautics.

Cutler, J. W. & Springmann, J. C., 2012. *Photodiode Placement & Algorithms for CubeSat Attitude Determination*, Michigan: Michigan Exploration Lab.

Daffalla , M., Tagelsir , A. & Kajo , A., 2016. *Hardware selection for attitude determination and control subsystem of 1U cube satellite*. khartoum, ResearchGate.

Elbert, B. R., 2004. *The Satellite Communication*. 2nd ed. London: Artech House, Inc..

El-Fatatry, A., 2003. *Inertial Measurement Units – IMU*, Chelmsford: BAE SYSTEMS – Advanced Technology Centre.

Farhat , A., Ivase , J., Lu , . Y. & Snapp , A., 2013. *Attitude Determination and Control System for CubeSat*, Worcester: Faculty of Worcester Polytechnic Institute .

Farrel, J. & Barth, M., 1999. *The Global Positioning System and Inertial*. 1st ed. New York:: McGraw-Hill.

Foust, J., 2015. *Space News*. [Online]
Available at: <https://spacenews.com/government-agencies-differ-on-use-usefulness-of-cubesats/>
[Accessed 22 October 2018].

Garbee, E. & Maynard, A., 2016. *The Conversation*. [Online]
Available at: <http://theconversation.com/the-future-of-personal-satellite-technology-is-here-are-we-ready-for-it-58478>
[Accessed 10 October 2018].

Gunnarsdóttir, E. L., 2012. *The Earth's Magnetic Field*, Reykjavik: Univerity of Iceland.

Harrison, D. M., 2002. *Inertial Reference Frames*, Toronto: Department of Physics, University of Toronto.

Howell, E., 2017. *space.com*. [Online]
Available at: <https://www.space.com/24839-satellites.html>
[Accessed 21 September 2018].

Jin, S., 2015 . *Satellite positioning -Methods, models and application*. 1st ed. Croatia: InTech.

Kok, M., Hol, J. D. & Schon, T. B., 2017. Using Inertial Sensors for Position and Orientation Estimation. *Foundations and Trends in Signal Processing*, 11(1-2), pp. 1-153.

Kuipers, J. B., 1999. *Rotation in the Space*. 1st ed. Varna: Coral Press.

- Lauszus, K. S., 2012. *TKJ Electronics Development*. [Online]
Available at: <http://blog.tkjelectronics.dk/2012/09/a-practical-approach-to-kalman-filter-and-how-to-implement-it/>
[Accessed 27 December 2018].
- Liebe, C. C., 1995. Star trackers for attitude determination. *IEEE Aerospace and Electronic Systems Magazine*, 10(6), pp. 10-16.
- Lis, A., 2013. *High Power Media*. [Online]
Available at: <https://www.highpowermedia.com/blog/3646/monocoque-materials>
[Accessed 12 October 2018].
- Lyndon B. Johnson Space Center, 1984. *Systems Engineering and Integration Space Station Program Office*, Texas : National Aeronautics and Space Administration.
- Magdy, K., 2018. *DeepBlue*. [Online]
Available at: <https://deepbluembedded.com/bluetooth-module-hc05-interfacing-pic-microcontroller-tutorial/>
[Accessed 12 November 2018].
- Manickam, S. & Ranjan, R., 2013. *Performance analysis of sun sensors for satellite systems*. s.l., International Conference on Advanced Electronic Systems.
- Maniyar, S., 2008. *1-Wire Communication with PIC Microcontroller*, s.l.: Microchip Technology Inc..
- Markley, F. L. & Crassidis, J. L., 2014. *Space Technology Library Fundamentals of Spacecraft Attitude Determination and Control*, New York : Microcosm Press and Springer.
- May, S., 2017. *NASA*. [Online]
Available at: www.nasa.gov/audience/forstudents/5-8/features/nasa-knows/what-is-a-satellite-58.html
[Accessed 27 September 2018].
- NASA, 1970. *SPACECRAFT SUN SENSORS*, Virginia: NATIONAL AERONAUTICS AND SPACE ADMINISTRATION.
- Pelton, J. N., Madry, S. & Camacho-Lara, S., 2017. *Handbook of Satellite Applications*. 2nd ed. Switzerland : Springer.
- Pillet, K., 2017. *Attitude Determination of a Cube Satellite Using sun sensor*, Tallinn: Tallinn University of Technology.
- Post, M. A., Li, J. & Lee, R., 2013. A low-cost Photodiode Sun sensor for CubeSat and Planetary Microover. *International Journal of Aerospace Engineering*, Volume 2013, pp. 1-9.
- Pumpkin, 2003. *Pumpkin CubeSat Kit User Manual*, San Francisco: Inc and Desin Massif.

Pumpkin, Inc., 2012 . *Single Board Computer Motherboard for Harsh Environments* , San Francisco: Pumpkin, Inc..

Pumpkin, Inc, 2012. *Single Board Computer Development Platform* , San Francisco: Pumpkin, Inc.

Smith, C. G., 2017. *Development and implementation of star tracker based attitude determination*, MISSOURI: Scholars' Mine.

Springmann, J. C., 2013. *Satellite Attitude Determination with Low-Cost Sensors*, Michigan: University of Michigan.

Stengel, R., 2016. *Spacecraft Sensors and Actuators*, Princeton: Princeton University.

Svartveit, K., 2003. *Attitude determination of the NCUBE satellite*, Trondheim: Department of Engineering Cybernetics.

Sweeting, M., 2018. Modern Small Satellites-Changing the Economics of Space. *Proceedings of the IEEE*, 106(3), pp. 343-361.

Tamir, A., 2013. Conservation Law of Mass. *J Chem Eng Process Technol*, 4(8), p. 114.

Texas Instruments, 2003. *Monolithic Photodiode and single-supply Transimpedance amplifier*, s.l.: Texas Instruments Incorporated.

Theil, S., Appel, P. & Schleicher, A., 2017. *Low Cost, Good Accuracy - Attitude Determination Using Magnetometer and Simple Sun Sensor*. Bremen, Center of Applied Space Technology and Microgravity (ZARM), University of Bremen.

Toorian, A., Blundell, E., Suari, J. P. & Twiggs, R., 2005. *CubeSats as reponsive satellite*, Los Angeles: AIAA 3rd Responsive Space Conference.

Tran, L., 2017. *Data Fusion with 9 Degrees of Freedom Inertial Measurement Unit To Determine Object's Orientation*, San Luis Obispo: California Polytechnic State University.

Virk, R., 2012. *Modular sun Sensor System for Orbital Attitude Determination*, Pennsylvania : The Pennsylvania State University .

Welch, G. & Bishop, G., 2006. *An Introduction to the Kalman Filter*, Chapel Hill: University of North Carolina .

Wertz, J. R., 1990. *Spacecraft Attitude determination and control*. 1978 ed. Dordrecht: Kluwer Academic Publishers.

Wessels, J. H., 2018. *Infrared Horizon Sensor for CubeSat Implementation*, Stellenbosch : Stellenbosch University.

7. Appendix 1:



FIGURE 52:MODULAR CHASSIS -STRUCTURE

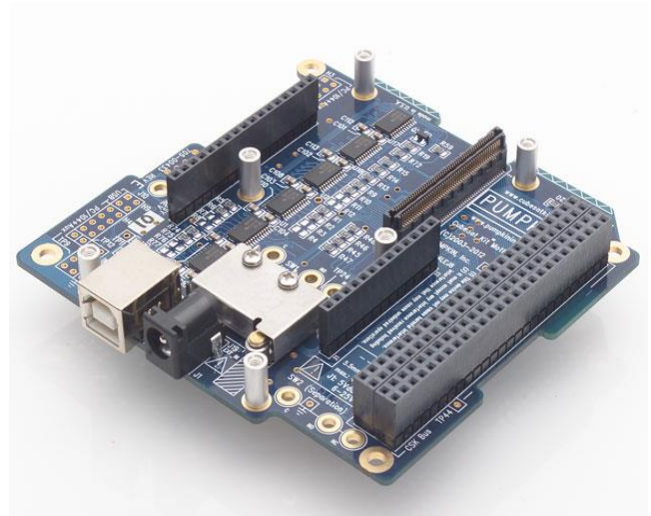


FIGURE 51:MOTHERBOARD



FIGURE 53:DEVELOPMENT BOARD

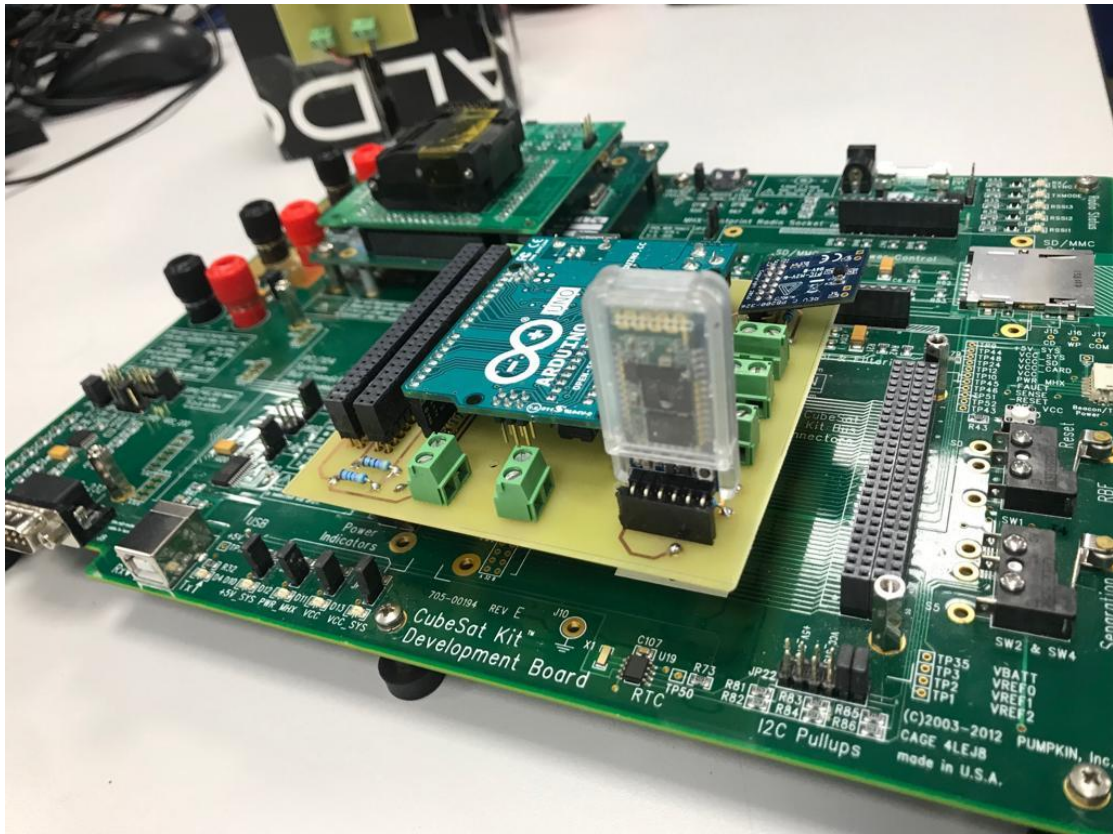


FIGURE 54: HARDWARE SYSTEM

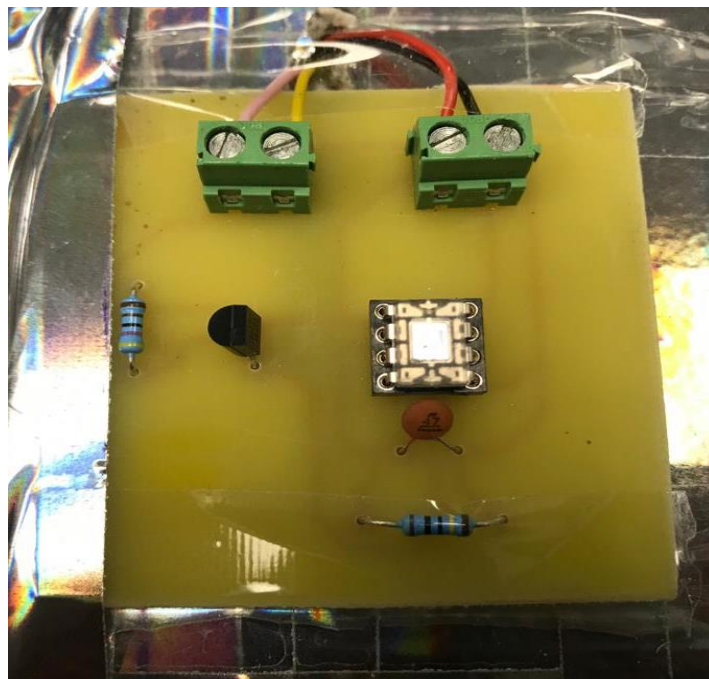


FIGURE 55: SUN SENSOR WITH TEMPERATURE SENSOR

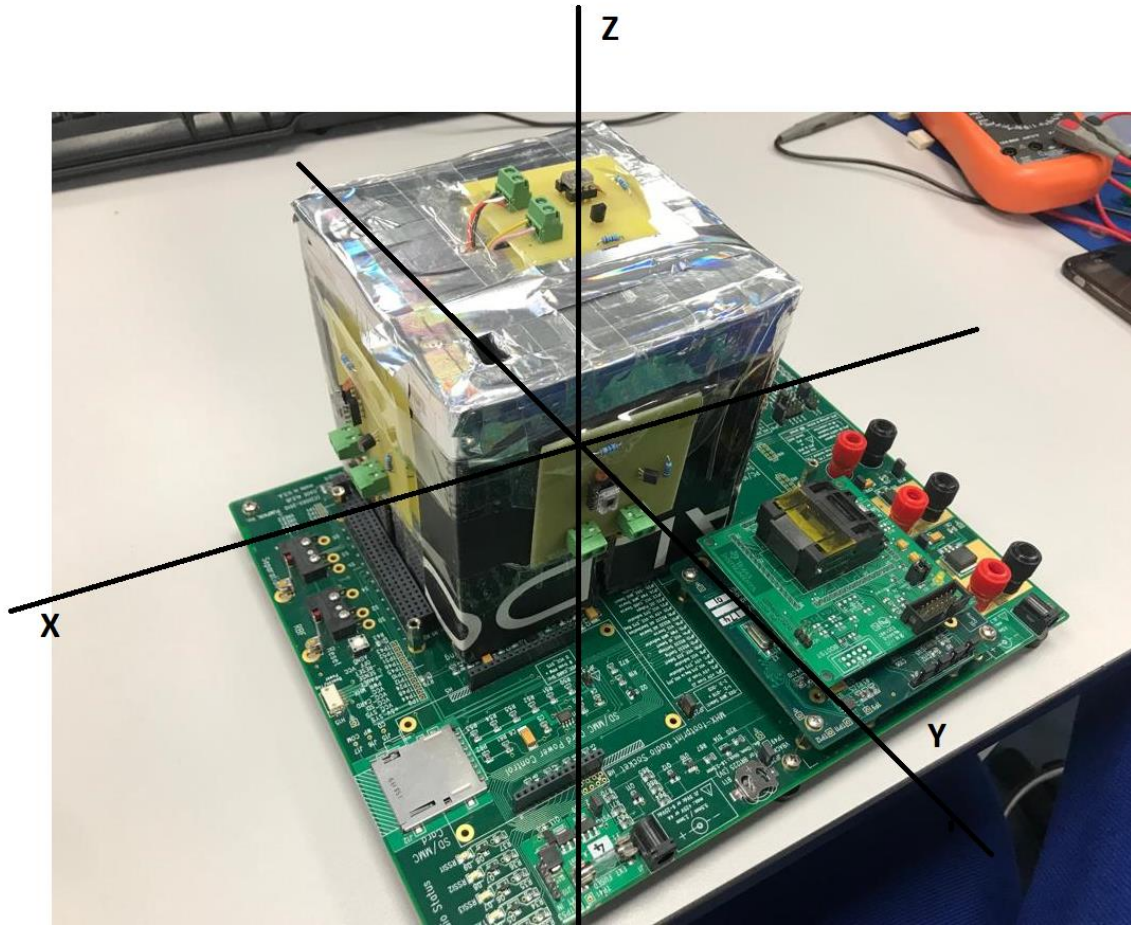


FIGURE 56: SUN SENSOR FIXED COORDINATES

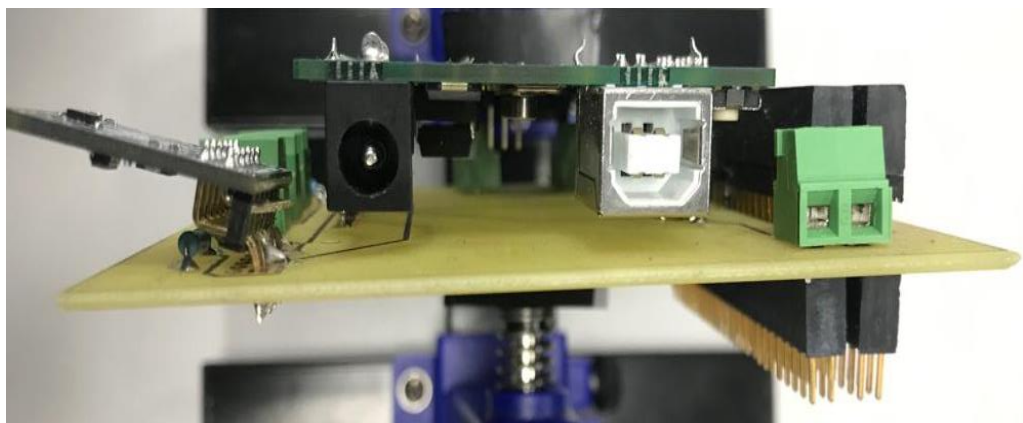


FIGURE 57: BOARD WITH ARDUINO, IMU, AND PPM CONNECTOR AT LEFT

8. Appendix 2:

Code:

Due to the complexity of the Code and length, The Code will not be shown at the report, but it can be accessed or download clicking one links below or by sending an email to the emails below.

<https://drive.google.com/drive/folders/1WD0EqSdIMixi-xzCzLluF7ATVyh47HCJ?usp=sharing>

Or:

<https://www.dropbox.com/sh/xfxnjhugmf82j0s/AADPHLn2dHcBixQc6-Y4uY2Da?dl=0>

Alternatively: send an email to:

josefneto11@hotmail.com

2169252n@student.gla.ac