CUBESPACE

# CubeADCS Hardware Configuration and Mission Overview

## Possible CubeADCS Hardware configurations

| | |
|---|---|
| DOCUMENT NUMBER | CS-DEV.GD.TPL-02 |
| VERSION | 1.03 |
| DATE | 10/07/2023 |
| PREPARED BY | C. LEIBBRANDT |
| REVIEWED BY | FL, LV, JG, WH |
| APPROVED BY | W. Morgan |
| DISTRIBUTION LIST | External |

# Revision History

| VERSION | AUTHORS | DATE | DESCRIPTION |
|---|---|---|---|
| 1.0 | W. Morgan | 08/05/2023 | First published version |
| 1.01 | W. Morgan | 11/05/2023 | Remove template reference |
| 1.02 | J. Gerber | 22/06/2023 | Minor updates |
| 1.03 | W. Morgan | 10/07/2023 | Minor updates |

# Reference Documents

The following documents are referenced in this document.

[1] CS-DEV.PD.CA-01 CubeADCS Product Description Ver.1.00 or later

[2] CS-DEV.ICD.CA-01 CubeADCS Standard ICD Ver.1.01 or later

# List of Acronyms/Abbreviations

| | |
|---|---|
| ACP | ADCS Control Program |
| ADCS | Attitude Determination and Control System |
| CAN | Controller Area Network |
| CoM | Center of Mass |
| COTS | Commercial Off The Shelf |
| CSP | Cubesat Space Protocol |
| CSS | Coarse Sun Sensor |
| CVCM | Collected Volatile Condensable Materials |
| DUT | Device Under Test |
| EDAC | Error Detection and Correction |
| EHS | Earth Horizon Sensor |
| EM | Engineering Model |
| EMC | Electromagnetic Compatibility |
| EMI | Electromagnetic Interference |
| ESD | Electrostatic Discharge |
| FDIR | Fault Detection, Isolation, and Recovery |
| FM | Flight Model |
| FSS | Fine Sun Sensor |
| GID | Global Identification |
| GNSS | Global Navigation Satellite System |
| GPS | Global Positioning System |
| GYR | Gyroscope |
| I2C | Inter-Integrated Circuit |
| ID | Identification |
| LTDN | Local Time of Descending Node |
| LEO | Low Earth Orbit |
| MCU | Microcontroller Unit |
| MEMS | Microelectromechanical System |
| MoI | Moment of Inertia |
| MTM | Magnetometer |
| MTQ | Magnetorquer |
| NDA | Non-Disclosure Agreement |

| | |
|---|---|
| OBC | On-board Computer |
| PCB | Printed Circuit Board |
| RTC | Real-Time Clock |
| RWA | Reaction Wheel Assembly |
| RW | Reaction Wheel |
| SBC | Satellite Body Coordinate |
| SOFIA | Software Framework for Integrated ADCS |
| SPI | Serial Peripheral Interface |
| SRAM | Static Random-Access Memory |
| SSP | Sub-Satellite Point |
| STR | Star Tracker |
| TC | Telecommand |
| TCTLM | Telecommand and Telemetry (protocol) |
| TID | Total Ionizing Dose |
| TLM | Telemetry |
| TML | Total Mass Loss |
| UART | Universal Asynchronous Receiver/Transmitter |

# Table of Contents

# Table of Tables

**No table of figures entries found.**

# Table of Figures

# 1 Introduction

CubeSpace aims to simplify the complicated task of integrating an ADCS into a client's satellite design.

This document serves as a collection of client-provided information related to a particular mission ADCS. This information includes the initial mission overview and parameters, which allows CubeSpace to suggest an optimal ADCS configuration. The document further captures the ADCS sensor and actuator component selection, and applicable configurable options for all components, which facilitates ordering of the CubeADCS.

The purpose of this document is thus to assist and guide a prospective client to allow the optimal configuration of a CubeSpace offered CubeADCS solution including allowing for customization choices. If however, additional or specialised customisation is required not suitably catered for in this document, please contact CubeSpace directly at info@cubespace.co.za.

This document is written with four main parts:

- **Part One** is presented in Chapter 3 "**Client Mission Overview**"
    - In this chapter, the client is initially prompted to provide as much information relating to the client mission as is available at the time of initial discussions.
    - This first part is initially all that is needed by CubeSpace to understand the client's satellite / mission needs, and as such, the first revision of this document can be treated as the Client mission overview baseline.
    - The client mission overview information provided allows CubeSpace to conduct an analysis (and simulation of expected performance if applicable) which will be supplied to the prospective client either as a Simulation Report, or an ADCS Review and Recommendation Report.
        - A) An ADCS Review and Recommendation Report is provided where ADCS requirements are not certain yet, or where the ADCS requirements are obviously fulfilled by standard solutions. It reports on significant aspects of the client mission ADCS and implications to consider,
        - B) A Simulation Report will indicate to what extent the client mission requirements can be satisfied by the CubeADCS offering.
    - In both reports, an optimal CubeADCS configuration (including sensor and actuator selection) that will meet the client needs is defined.
- **Part Two** is presented in Chapter 4 "CubeADCS Configurations"
    - Based on the above analysis, CubeSpace will subsequently transfer / indicate the proposed CubeADCS configuration in the tables provided in this document
    - Once completed by CubeSpace, a second version of this document can be baselined / considered the CubeADCS optimal configuration from a CubeSpace point of view, based on the information available at the time.
    - The intent is then for the client to confirm the CubeSpace proposed configuration details, and / or to iterate the previous information and /or counter suggest alternative configurations.

  - Such iteration (versions) of this document can then be managed until a mutually agreed configuration is reached, which can then serve as the basis for a formal CubeSpace proposal.
- **Part Three** is presented in Chapter 5 “Configurability options”
  - This part of the document can be completed concurrently to Part One and / or Part Two as and when the client feels comfortable to indicate further preferences and to indicate which options are preferred.
  - Part Three should ideally be completed together with Part Two in order to facilitate subsequent accurate proposal generation, system architecture definition and (hopefully) successful order placement.
- **Part Four** is presented in Chapter 6 “ADCS flight configuration”
  - This section is completed prior to launch of the satellite when final mass properties (moment of inertia matrix) and mounting locations of the ADCS sensors and actuators are known.
  - Part four is typically only completed after the CubeADCS has been integrated into the client satellite but may be filled in prior to this if the information is available.

# 2 Iteration History

Since this document serves as a collection of all client-provided information applicable to a mission, and the fact that this information will mature and expand over time, this section aims to capture the version information and changes that are made at each iteration of the document, to give CubeSpace clarity on the exact changes, and to indicate baselines.

| Iteration | Date | Document change details | Performed/requested by |
|---|---|---|---|
| A | | CubeSpace empty document | |
| | | | |
| | | | |
| | | | |
| | | | |

# 3 Client Mission Overview

## 3.1 Project information

| Information required | Information provided |
|---|---|
| Organization Name | |
| Country | |
| Name of main point of contact | |
| E-Mail of main point of contact | |

## 3.2 Satellite information

Note: When specifying information dependent on spacecraft axes, CubeSpace requires that the following convention should be used:

- +X is fixed to the satellite body, and points towards the flight direction with nominal attitude.
- +Y is fixed to the satellite body, and points towards the orbit anti-normal direction with nominal attitude.
- +Z is fixed to the satellite body, and points towards nadir direction with nominal attitude.

It is important to recognize that the ADCS defined spacecraft body axes definition above is used throughout the CubeADCS design, configuration, and operations. CubeADCS control modes are also axes dependent. The CubeADCS does not make use of, or transform between, any other body-fixed coordinate frames that the client may have defined. The examples below illustrate different axes definitions and their rationale.

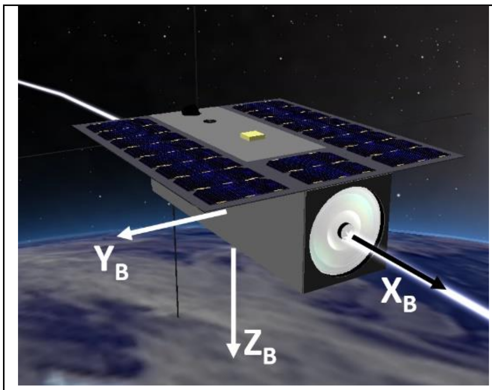


**Figure 1: Axes definition A - Imager payload points towards satellite +XB axis but nominal flight orientation is chosen to minimize drag. Chosen axes definition makes sense for an orbit with varying sun beta angle (i.e. ISS orbit)**

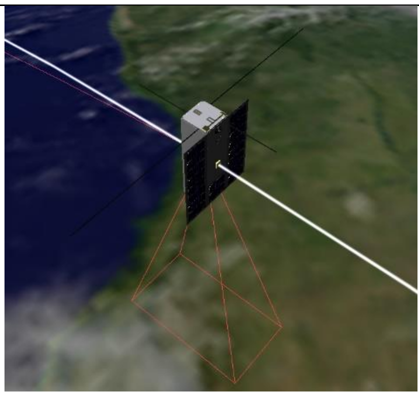

**Figure 2: Axes definition A - The satellite in Figure 1 must perform a -90° pitch manoeuvre to point the imager to a ground target.**

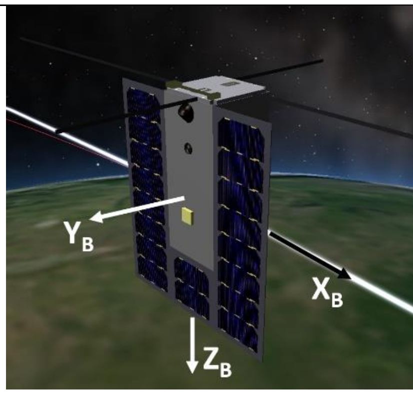


**Figure 3: Axes definition B - Nominal orientation is chosen so that imager points nadir thus no manoeuvre is necessary. But nominal orientation results in higher drag and only makes sense for a fixed sun angle (i.e. SSO orbit)**

| Information required | Information provided |
|---|---|
| Project/Satellite name | |
| Satellite Qty | |
| Satellite Size (3U, 6U, etc. For non-Cubesat sizes, please specify rough dimensions) | |
| Payload Type and preferred pointing direction (nadir, sun, ground station, etc.)<br><br>If optic, type of imager (Push-broom/TDI/Snapshot, etc.) | |
| Intended orbit(s)[1] with expected altitude and inclination (e.g. SSO, ISS, polar, etc.) | |
| Expected launch date | |
| Estimated MoI (please respect the indicated units)[2] | Lxx =<br><br>Lxy = |

[1] The orbit, especially the altitude, will have a big influence on the performance of the system. This is a necessary input to our simulation.

[2] This information is required to size the actuators needed for the mission. With the orbit, it is one of the most important inputs to our simulations.

| Information required | Information provided |
|---|---|
| (Moments of inertia: (kilograms * square meters) Taken at the center of mass and aligned with the output coordinate system (Using positive tensor notation.)<br><br>Products of inertia should be indicated too, if known<br><br>The MoI must be specified around the CoM and for the axes defined earlier<br><br>If you are using SolidWorks, please send us the "L" values as shown below<br><br>Mass = 3297.23 grams<br><br>Volume = 1130783.24 cubic millimeters<br><br>Surface area = 1257903.58 square millimeters<br><br>Center of mass: ( millimeters )<br>X = -1.51<br>Y = -3.78<br>Z = -5.46<br><br>Principal axes of inertia and principal moments of inertia: ( grams * square millimeters )<br>Taken at the center of mass.<br>Ix = ( 0.00, -0.01, 1.00) Px = 6567763.46<br>Iy = (-0.02, 1.00, 0.01) Py = 31922191.96<br>Iz = (-1.00, -0.02, 0.00) Pz = 32283548.38<br><br>Moments of inertia: ( grams * square millimeters )<br>Taken at the center of mass and aligned with the output coordinate system.<br>Lxx = 32282895.65 Lxy = -5620.02 Lxz = -117050.45<br>Lyx = -5620.02 Lyy = 31920552.52 Lyz = -211224.75<br>Lzx = -117050.45 Lzy = -211224.75 Lzz = 6570055.62<br><br>Moments of inertia: ( grams * square millimeters )<br>Taken at the output coordinate system.<br>Ixx = 32428182.36 Ixy = 13163.33 Ixz = -89950.05<br>Iyx = 13163.33 Iyy = 32026176.75 Iyz = -143203.16<br>Izx = -89950.05 Izy = -143203.16 Izz = 6624684.99 | Lxz =<br><br>Lyx =<br><br>Lyy =<br><br>Lyz =<br><br>Lzx =<br><br>Lzy =<br><br>Lzz =<br><br>Center of mass: (meters)<br><br>X =<br><br>Y =<br><br>Z =<br><br>Center of pressure/Geometric origin: (meters)<br><br>X =<br><br>Y =<br><br>Z =<br><br>Mass (kg) = |
| List of deployables and estimated dimensions | |
| Rough Sketch/CAD showing: *<br>- Flight direction[3]<br>- (Please include figures as deemed relevant either here or just after this table)<br>- Deployables and antennas[4]<br>- Available locations for sensor placement[5] | |

[3] This information is needed to understand how the satellite will be oriented and what the nominal attitude is.

[4] This information is needed to model the aerodynamic disturbances.

[5] This information is needed to suggest a preliminary sensor configuration to optimize the performance.

| Information required | Information provided |
|---|---|
| - Placement of solar panels (deployable, if any, and body-mounted) and number of solar cells[6] | |

**[Figure Placeholder for client]**

**Figure 4: 3D illustration of Spacecraft geometry. Direction of travel is in the +X direction. +Z direction is nominally Earth-facing.**

[6] This information is needed to model the magnetic disturbances from the solar panels

**[Figure Placeholder for client]**

**Figure 5: Illustration of key dimensions and locations of spacecraft and solar panels.**

**[Figure Placeholder for client]**

**Figure 6: Illustration of acceptable field of view of side and/or Earth facing sensors.**

**[Figure Placeholder for client]**

**Figure 7: [Title for additional figure]**

## 3.3 ADCS requirements

Over and above the minimum set of performance requirements requested in the following table, and if available, the client is requested to provide a separate "CubeADCS Requirements" document to CubeSpace. Using these together with the information provided by the client in the earlier sections of this chapter, CubeSpace will aim to propose an optimal CubeADCS configuration in Part Two tables in the next iteration (revision) of this document. In a fully complaint situation, CubeSpace will propose one of our standard units and present the expected performance.

If, however, CubeSpace is not able to comply with all "CubeADCS Requirements", CubeSpace will indicate compliance / non-compliance / make counter suggestions in the separate CubeADCS Requirements document with the aim to quickly converge to a mutually agreed requirement list. Once agreement is reached, CubeSpace will be able to propose an optimal CubeADCS configuration which may not be a stock standard CubeADCS but rather a customized version instead.

> Note: In the case where the requested "CubeADCS Requirements" document is not readily available from the client, CubeSpace can offer such a document as a template for the client, which will allow the client to quickly adapt the proposed CubeSpace ADCS requirements to accurately reflect the client requirements.

Lastly, finalizing an agreed set of requirements prior to the proposal is not imperative and is most often further matured post order placement as satellite details become clearer to the client (and thus to CubeSpace). Nevertheless, a final agreed baseline set of requirements is imperative for non-standard CubeADCS configurations to be fully acceptance tested.

| Information required | Information provided[7][8] |
|---|---|
| **Detumbling** | |
| Maximum tip-off rate | |
| Maximum detumbling duration | |
| **Nadir pointing and reference attitude tracking** | |
| Is the ADCS required to follow a given roll, pitch, and yaw reference? | |
| If yes, what is the maximum values for roll, pitch and yaw angles? | |
| If yes, what is the required slew rate when changing attitude reference angles? | |
| If yes, what is the required attitude knowledge error? | |
| If yes, what is the required attitude pointing error? | |
| **Ground target tracking** | |
| Is the ADCS required to track a ground target? | |
| If yes, which satellite axis should point to the target? | |

[7] Specify 'TBD' for values not yet known

[8] For attitude pointing and knowledge errors, specify the error margins (1-sigma or 3-sigma)

| Information required | Information provided[78] |
|---|---|
| If yes, should target tracking be possible in sunlit, eclipse or both? | |
| If yes, what is the required attitude knowledge error? | |
| If yes, what is the required attitude pointing error? | |
| **Sun tracking** | |
| Is the ADCS required to track the sun? | |
| If yes, which satellite axis should point to the sun? | |
| If yes, what is the maximum allowed error angle? | |
| **Inertial pointing** | |
| Is the ADCS required to track an inertial pointing vector (i.e. point to a star?) | |
| If yes, which satellite axis should point to the inertial target? | |
| If yes, what it the required attitude knowledge error? | |
| If yes, what is the required attitude pointing error? | |
| **Satellite pointing** | |
| Is the ADCS required to track another satellite? | |
| If yes, which satellite axis should point to the target satellite? | |
| If yes, what it the required attitude knowledge error? | |
| If yes, what is the required attitude pointing error? | |
| **Moon pointing** | |
| Is the ADCS required to track the moon? | |
| If yes, which satellite axis should point to the moon? | |
| If yes, what is the required attitude knowledge error? | |
| If yes, what is the required attitude pointing error? | |
| **Spin control modes** | |
| Is the ADCS required to provide a safe-mode spin control? | |
| If yes, around which axis? | |
| If yes, should the spin vector point to the sun? | |
| **Imaging and ground scanning** | |
| Is the ADCS required to perform any ground scanning operations not covered by previous modes? | |
| If yes, which scanning operations are required? | |
| **Stability** | |
| Does the satellite or payload have any stability requirements? | |
| If yes, in which of the above modes? | |

| Information required | Information provided[78] |
|---|---|
| If yes, what is the required stability? | |
| **Other** | |
| Are there any other ADCS requirements or modes not covered by the above? | |
| Use this section to provide any additional information | |
| 1) | |
| | |
| | |
| | |
| | |

# 4 CubeADCS Configurations

As discussed in the **Introduction** chapter, CubeSpace will suggest optimal choices based on the information provided in **Part One** and the simulation done there-after. If the client is comfortable to complete the following tables when filling out this document for the first time (before CubeSpace has reviewed the information provided in **Part One**), the client may do so. If not, then the previous point stands.

This section must be completed before CubeSpace can issue a quote.

Please reference to [1] and [2] for details regarding the products and the following standard options.

## 4.1 CubeADCS configuration

Please indicate your choice of the CubeADCS configuration.

| # | CubeADCS configuration | Selection |
|---|---|---|
| 1. | Standard CubeADCS Core Stack for a 3U satellite | ○ |
| 2. | Standard CubeADCS Core Stack for a 6U satellite | ○ |
| 3. | Standard CubeADCS Core (external sensors and actuators) for larger than 6U satellite | ○ |
| 4. | Non-standard CubeADCS | ○ |

## 4.2 Non-standard CubeADCS options

If a non-standard CubeADCS solution is needed, please provide details in separate document.

| # | CubeADCS configuration | Selection |
|---|---|---|
| 1. | Non-standard CubeDoor<br>(Please provide relevant electrical interfacing details between the OBC and the CubeADCS (ICD documentation, diagrams, backplane design etc will be greatly appreciated) | ☐ |
| 2. | Non-standard Mounting<br>(The four corner-rail mounting system commonly used in CubeSats can also be accommodated with e.g. threaded holes allowing for the enclosure of the core CubeADCS stack to be bolted into position. Please provide relevant mechanical interfacing details applicable to the mounting of the CubeADCS. CAD, STP file, diagrams, backplane design etc will be appreciated) | ☐ |
| 3. | Non-standard CubeConnect<br>Please provide relevant motivation and details if the standard CubeConnect is not considered suitable | ☐ |

## 4.3 Non-standard CubeADCS motivations

If a non-standard CubeADCS solution is needed, please provide the reasons for needing a custom solution. (As an example of reason: the standard CubeDoor PC104 interface will not be compatible with the client satellite OBC interface – a Custom CubeDoor Design is needed)

| # | Non-standard CubeADCS motivations | CubeSpace response |
|---|---|---|
| | | |
| | | |
| | | |
| | | |

## 4.4 CubeSpace Sensors

Based on your option of CubeADCS configuration above, indicate the appropriate CubeSpace sensors, noting that a CubeMag Deployable is always included in the CubeADCS:

| # | CubeSpace CubeADCS Sensors | Number of sensors |
|---|---|---|
| 1. | CubeMag Deployable – minimum 1 always included | |
| 2. | CubeMag Compact | |
| 3. | CubeSense Sun (up to 4) | |
| 4. | CubeSense Earth (up to 2) | |
| 5. | CubeStar (up to 2) | |
| 6. | CubeStar Baffle (add on to CubeStar) | |
| 7. | Coarse Sun Sensors (up to 10)<br>(CubeSpace can provide the Coarse Sun Sensor photodiodes in loose form, or the client can opt to supply and implement them during satellite assembly themselves, with CubeSpace only providing the harnessing) | |

## 4.5 Third party sensors

Please indicate the appropriate CubeSpace supported 3rd party sensors:

| # | CubeSpace supported 3rd party CubeADCS Sensors | Model | Number of sensors |
|---|---|---|---|
| 1. | TYSpace Pico Star Tracker (Up to 2) | PST3S-H5 | |
| 2. | µFORS Fibre-optic gyros (3-axis, one sensor per axis) | µFORS-3UC | |
| 3. | | | |
| 4. | | | |

## 4.6 CubeSpace Actuators

Please indicate the appropriate CubeSpace actuators, noting that 3 CubeTorquers are always included in the CubeADCS. A “Standard CubeADCS Core Stack for a 3U satellite” includes 1x CR0002 and 2x CR0003. A “Standard CubeADCS Core Stack for a 6U satellite” includes 3x CR0004. Refer to 4.1.

For the “Pyramid” wheel options, a single Pyramid contains 4 wheels with the pyramid mounting structure.

| # | CubeSpace CubeADCS Actuators | Sizing | Number of actuators |
|---|---|---|---|
| 1. | CubeTorquer CR0002 | 0.2 $Am^2$ | |
| 2. | CubeTorquer CR0003 | 0. 3 $Am^2$ | |
| 3. | CubeTorquer CR0004 | 0.4 $Am^2$ | |
| 4. | CubeTorquer CR0006 | 0. 6 $Am^2$ | |
| 5. | CubeTorquer CR0008 | 0.8 $Am^2$ | |
| 6. | CubeTorquer CR0010 | 1.0 $Am^2$ | |

| | | | |
|---|---|---|---|
| 7. | CubeTorquer CR0012 | 1.2 $Am^2$ | |
| 8. | CubeTorquer CR0020 | 2.0 $Am^2$ | |
| 9. | CubeWheel CW0017 (up to 4) | 0.23 mNm | |
| 10 | CubeWheel CW0057 (up to 4) | 4 mNm | |
| 11 | CubeWheel CW0162 (up to 4) | 12 mNm | |
| 12 | CubeWheel CW0057 Pyramid (max 1) | 4 mNm | |
| 13 | CubeWheel CW0162 Pyramid (max 1) | 12 mNm | |

## 4.7 Third party actuators

Based on your option of CubeADCS configuration above, indicate the appropriate CubeSpace supported third party actuators:

| # | CubeSpace supported 3rd party CubeADCS Actuators | Number of actuators |
|---|---|---|
| 1. | NewSpace Systems (NSS) T065 reaction wheels | |
| 2. | NewSpace Systems (NSS) T2 reaction wheels | |
| 3. | NewSpace Systems (NSS) 5 $Am^2$ Torquer rods | |
| 4. | NewSpace Systems (NSS) 10 $Am^2$ Torquer rods | |
| 5. | NewSpace Systems (NSS) 20 $Am^2$ Torquer rods | |
| 6. | | |
| 7. | | |

## 4.8 Third party GNSS

Based on your option of CubeADCS configuration above, indicate the appropriate CubeSpace supported GNSS:

| # | CubeSpace supported GNSS | Selection |
|---|---|---|
| 1. | No GNSS receiver required, and no PPS required (decision pending outcome of simulation report) | ○ |
| 2. | SkyTraQ GNSS receiver (Supported model number S1216) (Max 1) | ○ |
| 3. | SkyTraQ GNSS receiver (Supported model number Orion B16) (Max 1) | ○ |
| 4. | u-Blox GNSS receiver (Supported model number ZED-f9P) (Max 1) | ○ |
| 5. | Novatel GNSS receiver (Supported model number OEM719) (Max 1) | ○ |
| 6. | Other (Please provide interfacing details to allow CubeSpace to determine potential compatibility or if R&D will be required) | ○ |

## 4.9 Simulation software

Based on your option of CubeADCS configuration above, indicate whether the Simulation software (D2S2 – previously called EOS) – CubeSpace plugin should be included:

| # | CubeSpace Simulation Software | |
|---|---|---|
| 1. | D2S2: CubeSpace plugin | ☐ |

# 5 Configurability options

Please complete the options in this section and submit at time of Purchase Order. Any delay in receiving this completed section may delay production planning.

With reference to [1] and [2]:

## 5.1 Communication Interfaces

| # | CubeADCS to OBC communication | Selection |
|---|---|---|
| 1. | Communication between the CubeADCS and the OBC is via CAN (standard) | ☐ |
| 2. | CAN communication (if used) makes use of CSP | ☐ |
| 3. | Communication between the CubeADCS and the OBC is via UART (preferred secondary) | ☐ |
| 4. | Communication between the CubeADCS and the OBC is via I2C | ☐ |
| 5. | Communication between the CubeADCS and the OBC is via RS485 (cannot be selected together with RS422 below) | ○ |
| 6. | Communication between the CubeADCS and the OBC is via RS422 (cannot be selected together with RS485 above) | ○ |

## 5.2 Power supply and regulation

| # | CubeADCS Power supply and regulation | Selection |
|---|---|---|
| 1 | Satellite EPS supplies Battery Voltage and regulated 3V3, 5V | ○ |
| 2 | Satellite EPS supplies Battery Voltage and regulated 3V3 only | ○ |
| 7. | Customer supplies separate 3.3V – 18.0V supply for CubeADCS backup registers<br>Note: If this option is not selected, CubeSpace will supply the backup registers from the 5V supply internally. | ☐ |

* From Q1 2024 option for satellite EPS to supply only Battery voltage will be available, pending on environmental qualification.

## 5.3 Battery voltage

| # | CubeADCS Battery voltage | Client Nominal Voltage |
|---|---|---|
| 1. | Battery voltage, if supplied to CubeADCS, has a minimum of 7.5 V and maximum level of 18.0 V (standard) | |

## 5.4 Communication hardware specific options

| # | CubeADCS communication hardware specific options | Selection |
|---|---|---|
| 1. | 3k ohm pull-up resistors implemented on I2C bus | ☐ |
| 2. | I2C voltage level is other than the standard 3V3 (please contact CubeSpace) | ☐ |
| 3. | 1k ohm termination implemented on differentially signalled 1PPS interface | ☐ |
| 4. | 120 ohm termination implemented on CAN differential bus (preferred, if CAN communication is used) | ☐ |
| 5. | 120 ohm termination implemented on RS485/RS422 A/B differential pair (CubeADCS input/output, if RS485, or CubeADCS input, if RS422 communication is used) | ☐ |
| 6. | 120 ohm termination implemented on RS485 Y/Z differential pair (CubeADCS output, if RS422 communication is used) | ☐ |

## 5.5 Enable control line

| # | CubeADCS Enable control line | Selection |
|---|---|---|
| 1. | Satellite OBC uses CubeADCS Enable control line (preferred) | ○ |
| 2. | CubeADCS Enable control line is not connected to the OBC. CubeADCS is powered on when the 3V3 rail is powered (alternative) | ○ |

## 5.6 Boot control line

| # | CubeADCS Boot control line | Selection |
|---|---|---|
| 1. | Satellite OBC connects to the CubeADCS Boot control line<br>Note: To make use of this functionality, it is required that the CubeADCS UART pins are also connected from the CubeADCS to the OBC. | ○ |
| 2. | CubeADCS Boot control line is not connected to the OBC | ○ |

## 5.7 GNSS and PPS interface

| # | CubeADCS GNSS and PPS interface | Selection |
|---|---|---|
| 1. | UART interface to client supplied GNSS<br>(The GNSS will be set up by the OBC or from a pre-configured configuration. CubeADCS will only listen to the received data from the GNSS when it is active) | ☐ |
| 2. | RS485 interface to client supplied GNSS<br>(Alternative to above, however CubeADCS will still only listen to received data from the GNSS.) | ☐ |
| 3. | Differentially signalled 1PPS signal interface<br>(RS485 or LVDS voltage levels, please indicate if otherwise) | ☐ |
| 4. | Single-ended (CMOS) 1 PPS signal interface (alternative to above) | ☐ |

## 5.8 Grounding

| # | Grounding | Selection |
|---|---|---|
| 1. | CubeADCS and all -sensors and -actuators' signal GND is connected to their respective enclosures via at least one EMI filter.<br>As per 2, several of these filters are implemented on the CubeADCS, and at least one on each sensor or actuator. Each filter consists of a 1.5 Mega Ohm resistor in parallel with a 1 nF, 50V, C0G capacitor that connects the signal GND to the enclosure. | O |
| 2. | This is not acceptable (please contact CubeSpace) | O |

## 5.9 PC104 OBC / EPS interface pinout

With reference to [2]:

The client is requested to confirm the pin number assignment of the standard PC104 interface CubeDoor in the following table. Please indicate all assigned pins in the case that more than one pin is assigned to the same function.

Alternatively, if a custom option for the OBC / EPS interface is required, please provide the details in the next table (a template example is provided for a Samtec connector solution in Section 5.10).

Note: The CubeADCS internal CAN bus is normally not exposed on the external interface.

Note: The pin for supplying a different I2C supply voltage in the case that a different level than the standard 3V3 or 5V is required, is not yet routed on the PC104 CubeDoor, but will be connected in a future hardware revision.

| # | CubeDoor pinout | PC104 pin number options | | Default pin numbers(s) | Selected option<br>(Please select only from indicated options)<br>(Pin number / NC) |
|---|---|---|---|---|---|
| | | H1 header pins options | H2 header pin options | | |
| 1. | VBat | | 45, 46<br>(always connected) | H2: 45, 46 | H2: 45, 46 |
| 2. | GND | | 29, 30, 32<br>(always connected) | H2: 29, 30, 32 | H2: 29, 30, 32 |
| 3. | 5V | 47, 49, 51 | 25, 26 | H2: 25, 26 | |
| 4. | 3V3 | 48, 50, 52 | 27, 28 | H2: 27, 28 | |
| 5. | Vbackup | | 42 | NC | |
| 6. | CubeADCS Boot control (See 5.6) | | 19, 20 | H2: 20 | |
| 7. | CubeADCS Enable control | | 17, 18 | H2: 18 | |

| | | | | | |
|---|---|---|---|---|---|
| 8. | UART_RX from OBC (CubeADCS input) | 17, 18, 19 20 | 21, 22 | H1: 20 | |
| 9. | UART_TX to OBC (CubeADCS output) | 17, 18, 19 20 | 21, 22 | H1: 19 | |
| 10. | CAN_L to OBC | 1 | | H1: 1 | |
| 11. | CAN_H to OBC | 3 | | H1: 3 | |
| 12. | CubeADCS internal CAN_L | 2 | | NC | |
| 13. | CubeADCS internal CAN_H | 4 | | NC | |
| 14. | I2C_SDA to OBC | 41 | | H1: 41 | |
| 15. | I2C_SCL to OBC | 43 | | H1: 43 | |
| 16. | UART_TX to GNSS (CubeADCS output) | 39, 40 | | NC | |
| 17. | UART_RX from GNSS (CubeADCS input) | 39, 40 | | H1: 39 | |
| 18. | RS485 TXRX_P (CubeADCS input/output) / RS422 RX_P (CubeADCS input) | 9, 11, 13, 15 | | H1: 9 | |
| 19. | RS485 TXRX_N (CubeADCS input/output) / RS422 RX_N (CubeADCS input) | 9, 11, 13, 15 | | H1: 11 | |
| 20. | RS422 TX_P (CubeADCS output) | 9, 11, 13, 15 | | H1: 13 | |
| 21. | RS422 TX_N (CubeADCS output) | 9, 11, 13, 15 | | H1: 15 | |
| 22. | PPS_P (differential input) | 8, 10, 14, 16 | | NC | |
| 23. | PPS_N (differential input) | 8, 10, 14, 16 | | NC | |
| 24. | PPS (CMOS single-ended input) | 8, 10, 14, 16 | | H1: 14 | |

## 5.10 Alternative OBC / EPS interface pinout

The following table is a template example for a **Samtec TFM-150-01_02_03-X-D-WT** connector solution, which can be edited to detail a custom CubeDoor pinout.

| # | CubeDoor pinout | Samtec pin number options | Default pin number(s) | Selected option<br>(Please select only from indicated options)<br>(Pin number / NC) |
|---|---|---|---|---|
| 1. | VBat | 9, 10, 11, 12, 53, 54 | 53, 54 | |
| 2. | GND (all pins always connected) | 5-8, 13, 14, 19, 21-23, 25, 27, 31, 33, 35, 39, 43, 47, 48, 51, 52, 55, 56, 65, 75, 76 | 5-8, 13, 14, 19, 21-23, 25, 27, 31, 33, 35, 39, 43, 47, 48, 51, 52, 55, 56, 65, 75, 76 | 5-8, 13, 14, 19, 21-23, 25, 27, 31, 33, 35, 39, 43, 47, 48, 51, 52, 55, 56, 65, 75, 76 |
| 3. | 5V | 40, 41, 44, 45 | 40 | |
| 4. | 3V3 | 24, 26, 28, 32, 34, 36 | 24 | |
| 5. | Vbackup | 66 | NC | |
| 6. | I2C custom external supply voltage | 64 | NC | |
| 7. | CubeADCS Boot control | 97, 98 | 98 | |
| 8. | CubeADCS Enable control | 95, 96 | 96 | |
| 9. | UART_RX from OBC (CubeADCS input) | 61, 62, 81, 82 | 62 | |
| 10. | UART_TX to OBC (CubeADCS output) | 61, 62, 81, 82 | 82 | |
| 11. | CAN_H to OBC | 78 | 78 | |
| 12. | CAN_L to OBC | 77 | 77 | |
| 13. | CubeADCS internal CAN_H | 18 | NC | |
| 14. | CubeADCS internal CAN_L | 20 | NC | |
| 15. | I2C_SDA to OBC (always connected) | 74 | 74 | 74 |
| 16. | I2C_SCL to OBC (always connected) | 73 | 73 | 73 |
| 17. | UART_TX to GNSS (CubeADCS output) | 70 | 70 | |
| 18. | UART_RX from GNSS (CubeADCS input) | 69 | 69 | |
| 19. | RS485 TXRX_P (CubeADCS input/output) / RS422 RX_P (CubeADCS input) | 86, 88, 90, 92 | 86 | |
| 20. | RS485 TXRX_N (CubeADCS input/output) / RS422 RX_N (CubeADCS input) | 86, 88, 90, 92 | 88 | |
| 21. | RS422 TX_P (CubeADCS output) | 86, 88, 90, 92 | 90 | |
| 22. | RS422 TX_N (CubeADCS output) | 86, 88, 90, 92 | 92 | |
| 23. | PPS_P (differential input) | 49, 50, 57, 58 | 49 | |

| # | CubeDoor pinout | Samtec pin number options | Default pin number(s) | Selected option (Please select only from indicated options) (Pin number / NC) |
|---|---|---|---|---|
| 24. | PPS_N (differential input) | 49, 50, 57, 58 | 50 | |
| 25. | PPS (CMOS single-ended input) | 49, 50, 57, 58 | NC | |

# 6 Harness options

Please complete the options in this section at least 4 weeks prior to shipment. If these details are not available to CubeSpace 4 weeks prior to shipment, the shipment can be delayed.

## 6.1 Harness type

| # | CubeADCS Harness options | Selection |
|---|---|---|
| 1. | CubeSpace is to supply test harnesses for all interconnect within the CubeADCS<br><br>See the next item below for differences between test and flight harnesses. These test harnesses are black, pre-made harnesses that allows CubeSpace to streamline testing procedures. Test harnesses are not suitable for spaceflight due to their outgassing nature, and the individual wires are not twisted. Their preferred length is 300mm (please contact CubeSpace if this length is not suitable). | O |
| 2. | CubeSpace is to supply flight harnesses for all interconnect within CubeADCS<br><br>These are multicoloured, non-outgassing harnesses. Several of the individual wires form twisted pairs, for example CAN_H and CAN_L. See the next item below regarding flight harness lengths. | O |

## 6.2 Harness lengths

Please refer to client specific system diagram provided by CubeSpace after CubeADCS Configuration options have been concluded.

| # | CubeADCS Harness lengths | | | | | |
|---|---|---|---|---|---|---|
| | Subsystem A | Header | Wire Type | Subsystem B | Subsystem Name from System Diagram | Length [mm] |
| 1. | CubeConnect | Actuator | PTFE 26AWG | Reaction Wheel 1 | | |
| 2. | | Actuator | PTFE 26AWG | Reaction Wheel 2 | | |
| 3. | | Actuator | PTFE 26AWG | Reaction Wheel 3 | | |
| 4. | | Actuator | PTFE 26AWG | Reaction Wheel 4 | | |
| 5. | | Sensor | PTFE 26AWG | Sensor 1 | | |
| 6. | | Sensor | PTFE 26AWG | Sensor 2 | | |
| 7. | | Sensor | PTFE 26AWG | Sensor 3 | | |
| 8. | | Sensor | PTFE 26AWG | Sensor 4 | | |
| 9. | | Sensor | PTFE 26AWG | Sensor 5 | | |
| 10. | | Sensor | PTFE 26AWG | Sensor 6 | | |
| 11. | | Sensor | PTFE 26AWG | Sensor 7 | | |
| 12. | | Sensor | PTFE 26AWG | Sensor 8 | | |
| 13. | | PPS | PTFE 26AWG | Sensor [ ] | | |
| 14. | | | PTFE 26AWG | Sensor [ ] | | |
| 15. | | | PTFE 26AWG | Sensor [ ] | | |
| 16. | | Torquers | PTFE 28AWG | Magnetorquer 1 | | |
| 17. | | | PTFE 28AWG | Magnetorquer 2 | | |
| 18. | | | PTFE 28AWG | Magnetorquer 3 | | |
| 19. | | CSS | PTFE 28AWG | CSS 1 to 5 | | |
| 20. | | CSS | PTFE 28AWG | CSS 6 to 10 | | |

Please note the following:

1. Harnesses are measured from the top of the connectors (where the terminals are plugged in), see Figure 8.
2. CS assumes a tolerance of -2% (short) and +10% (long) for the harnesses, with the nominal length provided by the client.
3. Ground Support Equipment and harnesses are not discussed in this document, which includes the interfacing harness to the CubeADCS for ground tests.
4. Sensor harnesses can be connected to any sensor header, and similarly actuator harnesses can be connected to any actuator header.
5. The deployable magnetometer will be an in-line harness connecting from CubeConnect to a soldered-in wire loom on CubeMag Deployable of 50mm. This 50mm must not be included in the harness length above.
6. The CSS has an extra 50mm harness attached to which is not included in the length specified in this document
7. The harness to the CSS can be provided open ended for the client to cut to the desired length. The client is then required to crimp the connector. Refer to the "Course Sun Sensor" section in [2].
8. For further details on CubeConnect, refer to the "CubeConnect" section in [2].

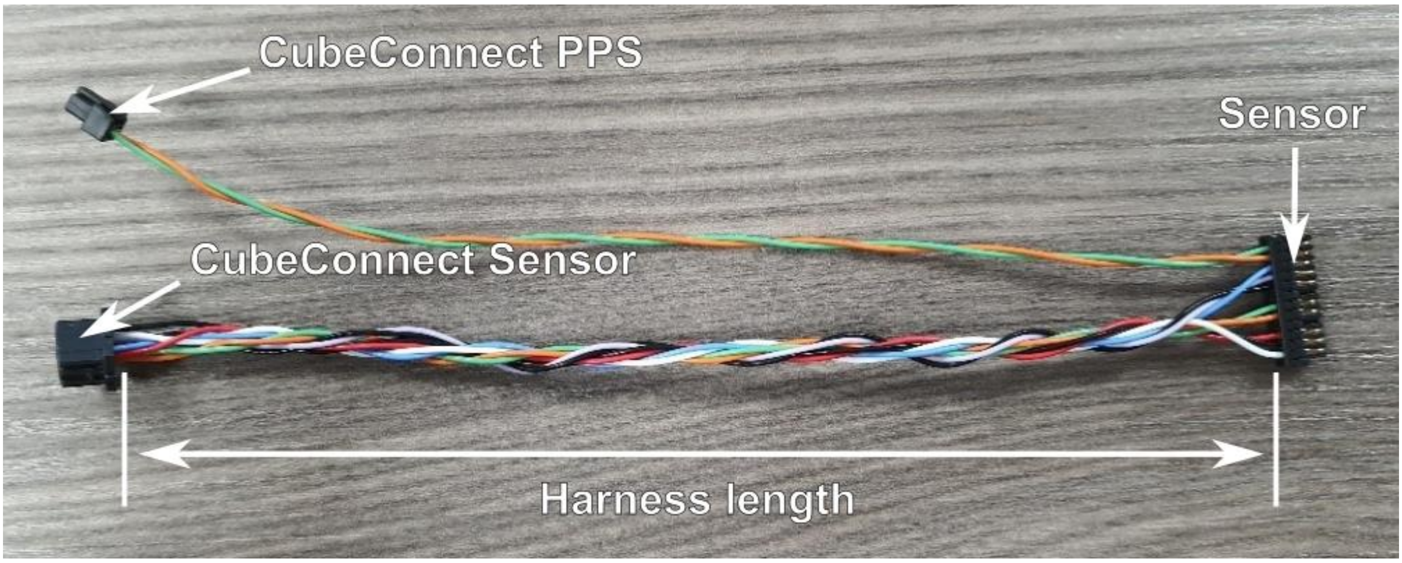


**Figure 8: Example CubeNode with PPS harness**

# 7 ADCS Flight Configuration

The information supplied in this chapter will be used to define the CubeADCS firmware configuration for flight. It is important that the supplied information is correct otherwise performance and successful operation of the CubeADCS cannot be guaranteed.

## 7.1 Satellite Parameters

Please provide the satellite's inertia moments and products, taken at the centre of mass with respect to the SBC frame defined in section 0. If the satellite has more than one configuration (i.e. stowed vs. deployed solar panels), please specify values for all configurations.

| | Configuration 1 | Configuration 2 | Configuration 3 |
|---|---|---|---|
| **Label** | Stowed | Deployed | |
| **Moments of Inertia** | | | |
| $I_{xx}$ **(kg.m²)** | | | |
| $I_{yy}$ **(kg.m²)** | | | |
| $I_{zz}$ **(kg.m²)** | | | |
| **Products of Inertia** | | | |
| $I_{xy}$ **(kg.m²)** | | | |
| $I_{xz}$ **(kg.m²)** | | | |
| $I_{yz}$ **(kg.m²)** | | | |

When using sun-tracking, ground target tracking, satellite tracking (if applicable), inertial pointing (if applicable) and moon tracking (if applicable) control modes, the ADCS will control the satellite to point the payload(s) to the target. The payloads can be mounted in any orientation relative to the SBC. It is thus necessary for the ADCS to know how payloads are mounted, in order for it to point to the target for each of the mentioned control modes.

This information is included in the ADCS configuration by specifying the SBC relative unit vector direction per control mode. The CubeADCS will control the satellite attitude so that the configured SBC vector direction is pointed towards the target.

For example, if a patch antenna is mounted to have its a bore-sight towards the $+X_B$ direction, the target tracking body vector will be specified as $[+1, 0, 0]^T$. When selecting ground target tracking control mode, the satellite attitude will be controlled to point the antenna boresight (the SBC $+X_B$ axis) towards the ground station.

Please provide the sun-tracking, ground target-tracking and satellite-tracking (if applicable) and moon tracking (if applicable) SBC unit vectors that should be pointed towards the target.

If multiple target tracking payloads or antennae are present, indicate the primary selection.

| | Sun tracking | Target tracking | Satellite tracking | Moon tracking |
|---|---|---|---|---|
| X | | | | |
| Y | | | | |
| Z | | | | |

## 7.2 Initial Orbit Parameters

It is unlikely that exact orbit parameters will be available prior to launch. Orbit parameters can be changed by sending a telecommand to the ADCS. But if there is a need to have specific initial orbit parameters programmed, specify them in the table below.

| Orbit parameter | Value |
|---|---|
| **Epoch (SGP4 Epoch format YYDDD.ddd)** | |
| **Inclination (deg)** | |
| **RAAN (deg)** | |
| **B-Star** | |

| Orbit parameter | Value |
|---|---|
| **Eccentricity** | |
| **Arg, of Perigee (deg)** | |
| **Mean anomaly (deg)** | |
| **Mean Motion (rev/day)** | |

## 7.3 Mounting configuration

This section details the mounting of ADCS components inside the satellite. The mounting configuration is crucial to the correct and accurate operation of the CubeADCS. Specifying the correct mounting angles is sometimes error-prone without the necessary understanding of coordinate systems and coordinate transforms. It is therefore recommended that images (CAD renderings or photos) are supplied so that CubeSpace may verify the mounting information.

**[Figure Placeholder for client]**

**Figure 9: Photo or CAD rendering of satellite showing ADCS sensor and actuator placement.**

**[Figure Placeholder for client]**

**Figure 10: Photo or CAD rendering of satellite showing ADCS sensor and actuator placement.**

**[Figure Placeholder for client]**

**Figure 11: Photo or CAD rendering of satellite showing ADCS sensor and actuator placement.**

**[Figure Placeholder for client]**

**Figure 12: [Title for figure]**

### 7.3.1 Mounting transform specified by Euler angle sequence

The mounting of CubeSense Sun and Earth sensors, Star trackers, and magnetometers are specified using a Euler 3-2-1 angle sequence. The resulting transform will rotate vectors in the local sensor coordinate frame to SBC. The three angles are labelled alpha (α), beta (β), and gamma (γ), and they are applied in the same order: α is applied first, around the sensor local Z-axis. β is applied next, around the newly transformed Y-axis, and γ is applied last, around the newly transformed X-axis.

The CubeADCS Standard ICD [2] describes the sensor coordinate frames.

An example is given here for the CubeSense Earth (HSS). The sensor is mounted so that the boresight (the sensor +Z axis) points towards $+Y_B$, but angled by 22 degrees towards nadir ($+Z_B$) in the nominal flight orientation.

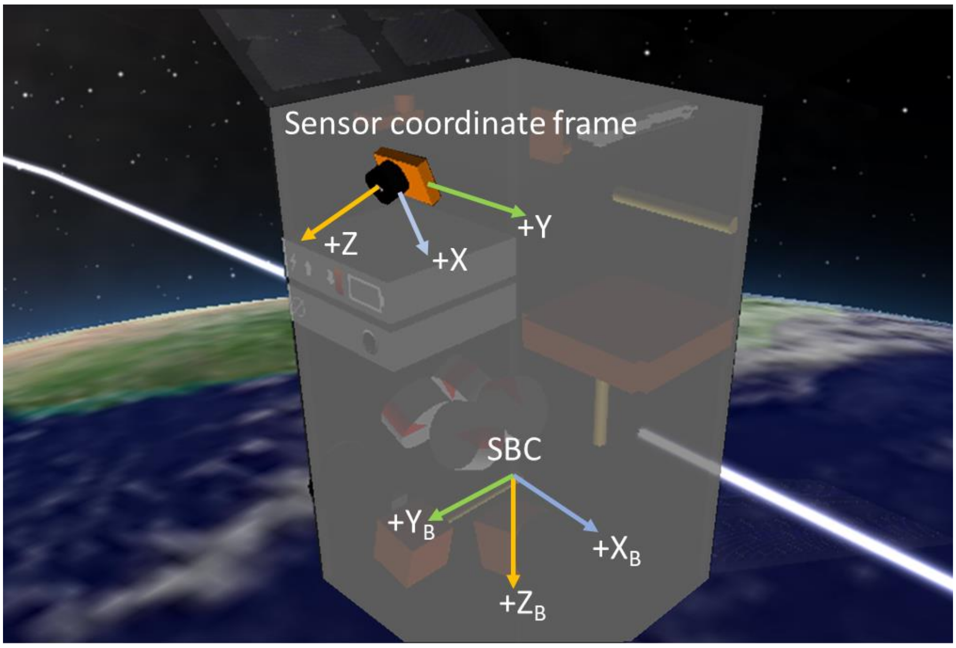


*Figure 13 Example mounting transform for the CubeSense Earth*

The transform angles for this scenario are α = 90°, β = 0°, γ = 68°.

Note that there may be more than one combination of angles that result in the same transform.

Note that the abstract nodes will be identified by serial number in the delivered unit configuration (i.e. which CubeSense Sun is FSS0, if more than one FSS is included).

Please complete the table below for the ADCS components in this unit.

| | Alpha (deg) | Beta (deg) | Gamma (deg) |
|---|---|---|---|
| FSS0 | | | |
| FSS1 | | | |
| FSS2 | | | |
| FSS3 | | | |
| HSS0 | | | |
| HSS1 | | | |

| | | | |
|---|---|---|---|
| MAG0 | | | |
| MAG1 | | | |
| STR0 | | | |
| STR1 | | | |
| EXT0 | | | |
| EXT1 | | | |

### 7.3.2 Mounting transform specified by axis direction

The CSS photodiodes, Gyro (IMU) and torquer rods have only a single axis that matters to ADCS operation. The CubeADCS allows for CSS photodiodes, gyros and torquer rods to be aligned along a principle SBC axis only.

For the IMU located on-board the CubeComputer, use the CubeComputer local coordinate frame from [2] to specify the mounting angles for GYR0.

| ADCS component | Axis selection (+/-X, +/-Y, +/-Z) |
|---|---|
| Torquer rods | |
| MTQ0 | |
| MTQ1 | |
| MTQ2 | |
| Gyroscopes | |
| GYR0 X | |
| GYR0 Y | |
| GYR0 Z | |
| GYR1 X | |
| GYR1 Y | |
| GYR1 Z | |

| ADCS component | Axis selection (+/-X, +/-Y, +/-Z) |
|---|---|
| CSS photodiodes | |
| CSS0 | |
| CSS1 | |
| CSS2 | |
| CSS3 | |
| CSS4 | |
| CSS5 | |
| CSS6 | |
| CSS7 | |
| CSS8 | |
| CSS9 | |

### 7.3.3 Reaction wheel mounting

The CubeADCS supports 3-axis aligned reaction wheels and a pyramid configuration.

#### 7.3.3.1 3-axis aligned reaction wheels

In case of 3-axis aligned reaction wheels, please specify the SBC axis to which each wheel is aligned.

Note that the abstract nodes (RWL0, RWL1, RWL2 and RWL3) will be identified by serial number in the delivered unit configuration.

| Reaction wheel | Axis selection<br>(+/-X, +/-Y, +/-Z) |
|---|---|
| RWL0 | |
| RWL1 | |
| RWL2 | |

#### *7.3.3.2 Pyramid wheel configuration*

In the case of a pyramid wheel configuration, the abstract wheel nodes (RWL0, RWL1, RWL2 and RWL3) must be matched to a pyramid axis as specified in the ICD [2]2].

Following this, the pyramid local coordinate frame must be transformed to SBC, using a Euler 3-2-1 sequence in the same manner as in 7.3.1.

| Reaction wheel | Pyramid axis selection<br>(A, B, C, D) |
|---|---|
| RWL0 | |
| RWL1 | |
| RWL2 | |
| RWL3 | |

| Pyramid mounting transform angle | Angle (deg) |
|---|---|
| Alpha | |
| Beta | |
| Gamma | |

# 8 Appendix: Configuration comments and notes

| | |
|---|---|
| Note 1: | This is an example note. Feel free to add notes on options where further discussion is required with |
| | reference to the applicable section. |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

| | |
|---|---|
| | |
| | |
| | |

# 9 Declaration

I, ____________________________________________________________, hereby declare that I am a legal representative of _______________________________________________________________.

| Signature | Date |
|---|---|
| | |