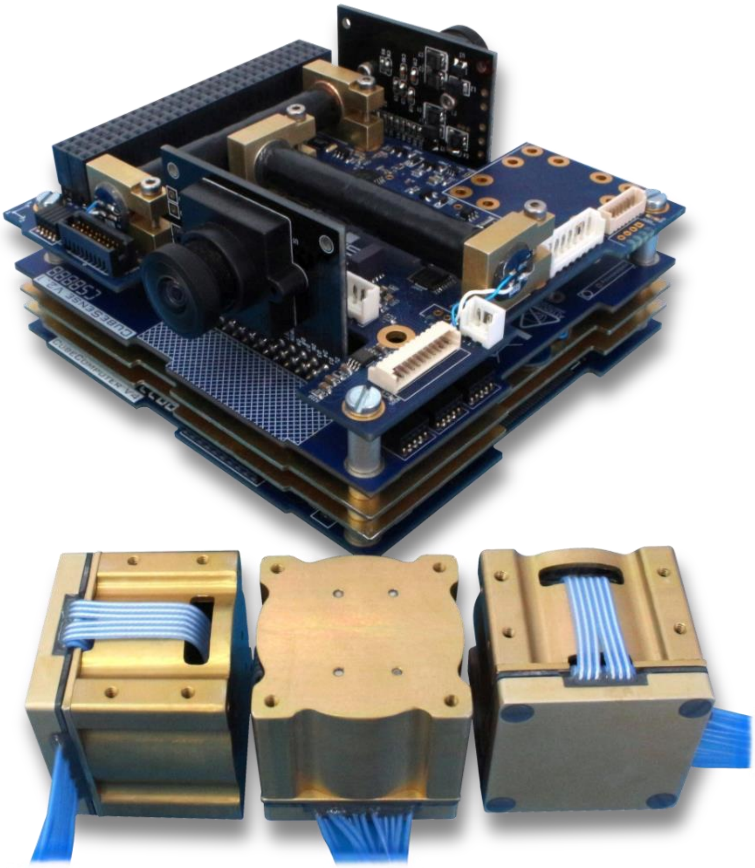
CubeADCS 3-Axis

The complete ADCS solution



CubeSupport Manual

Contact Us

Phone	:	+27 21 808 9499
E-mail	:	info@cubespace.co.za
Web	:	www.cubespace.co.za
Facebook	:	/CubeSpaceADCS
Twitter	:	@CubeSpace_ADCS

Physical Address

CubeSpace
The LaunchLab
Hammanshand Road
Stellenbosch, 7600
South Africa

Document	CubeADCS 3-Axis CubeSupport Manual
Version	3.02
Domain	Public
Date modified	17 October 2017
Approved by	Name: Mike-Alec Kearney
	Signature:

Document Version History

Version	Responsible person(s)	Pages	Date	Description of change
3.0	MK	ALL	20/03/2017	V3 First draft
3.01	MK	ALL	27/07/2017	Accept/Reject team changes
3.02	HWJ	19-21	17/10/2017	Added I2C cable and scripting functions

List of Acronyms/Abbreviations

ACP	ADCS Control Program
ADCS	Attitude Determination and Control System
CSS	Coarse Sun Sensor
ESD	Electrostatic Discharge
I^{2}C	Inter-Integrated Circuit
GSE	Ground Support Equipment
MCU	Microcontroller Unit
MEMS	Microelectromechanical System
OBC	Onboard Computer
PCB	Printed Circuit Board
RTC	Real-Time Clock
SBC	Satellite Body Coordinate
SPI	Serial Peripheral Interface
TC	Telecommand
TLM	Telemetry
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
UI	User Interface

Relevant reference documents

This document is to be used in combination with the following documents

Ref 1	CubeADCS – ICD	V3.0 or higher
Ref 2	CubeADCS – Option sheet	Completed by user at order of unit
Ref 3	CubeADCS – User Manual	V3.0 or higher
Ref 4	CubeADCS – Reference Manual	V3.0 or higher

Table of Contents

1. Introduction

This manual describes the CubeSupport utility and how it is used to interface with the CubeSpace ADCS module. The CubeSupport application runs on a PC with Windows and interfaces to the ADCS hardware module through a USB <-> UART cable. The utility is used to exercise various ADCS functions, view telemetry and status, download files and upload new firmware.

Demonstrations are available on the CubeSpace YouTube Channel.

2. Requirements

In order to run the CubeSupport utility your system must meet these minimum requirements

- Windows XP, Vista, 7, 8, 10
- .NET Framework v4.0 or higher
- 100MB Disk Space
- 512 MB RAM
- 1x USB 2.0 port

3. Installation

The CubeSupport utility is distributed as a compressed ZIP file. To install, extract the contents to a convenient location and run the CubeSupport.exe executable file. There is no need to run an installation program.

4. User Interface

The main parts of the CubeSupport user interface are shown below. After establishing a connection with the ADCS hardware through the *Connection Settings* part of the UI, the *User Controls* section will update to show the ADCS specific UI.

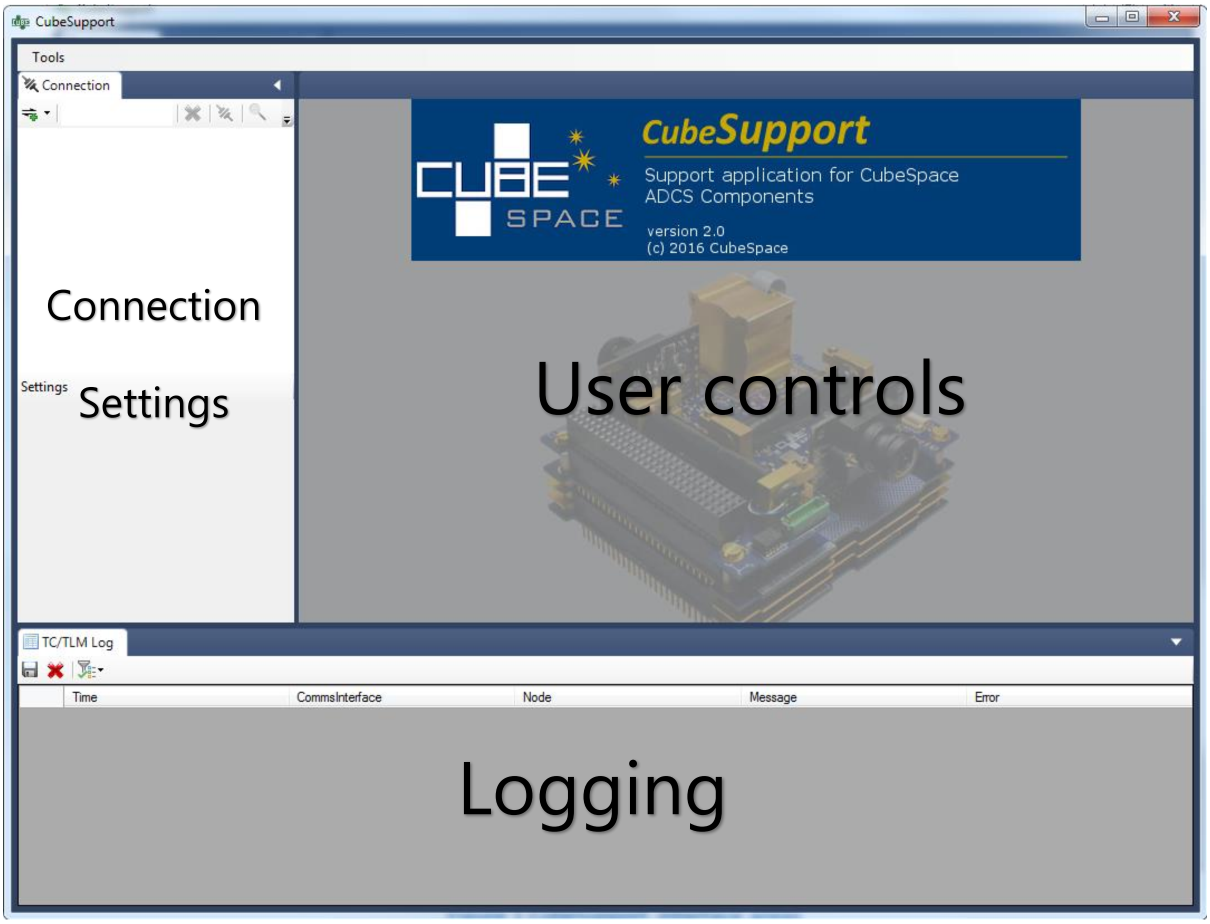


Figure 1 – CubeSupport interface areas

5. Setup

5.1 GSE connection

Before applying power to the ADCS unit, connect the GSE cable to the CubeComputer Debug UART. The connection should be made as per Figure 2. Observe the order in which the pins on the cable are connected. (From left to right, the cable colours should be yellow, orange, black).

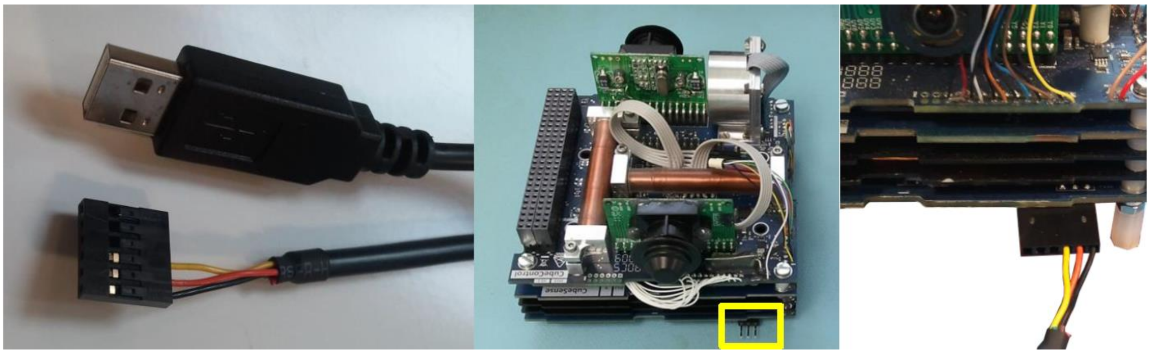


Figure 2 – GSE connection to ADCS module

The other end of the GSE cable should be connected to the USB port of the PC that is running the CubeSupport program.

5.2 Power

The ADCS unit should be powered through the PC104 connector as per the ICD [Ref 1] and Option Sheet [Ref 2].

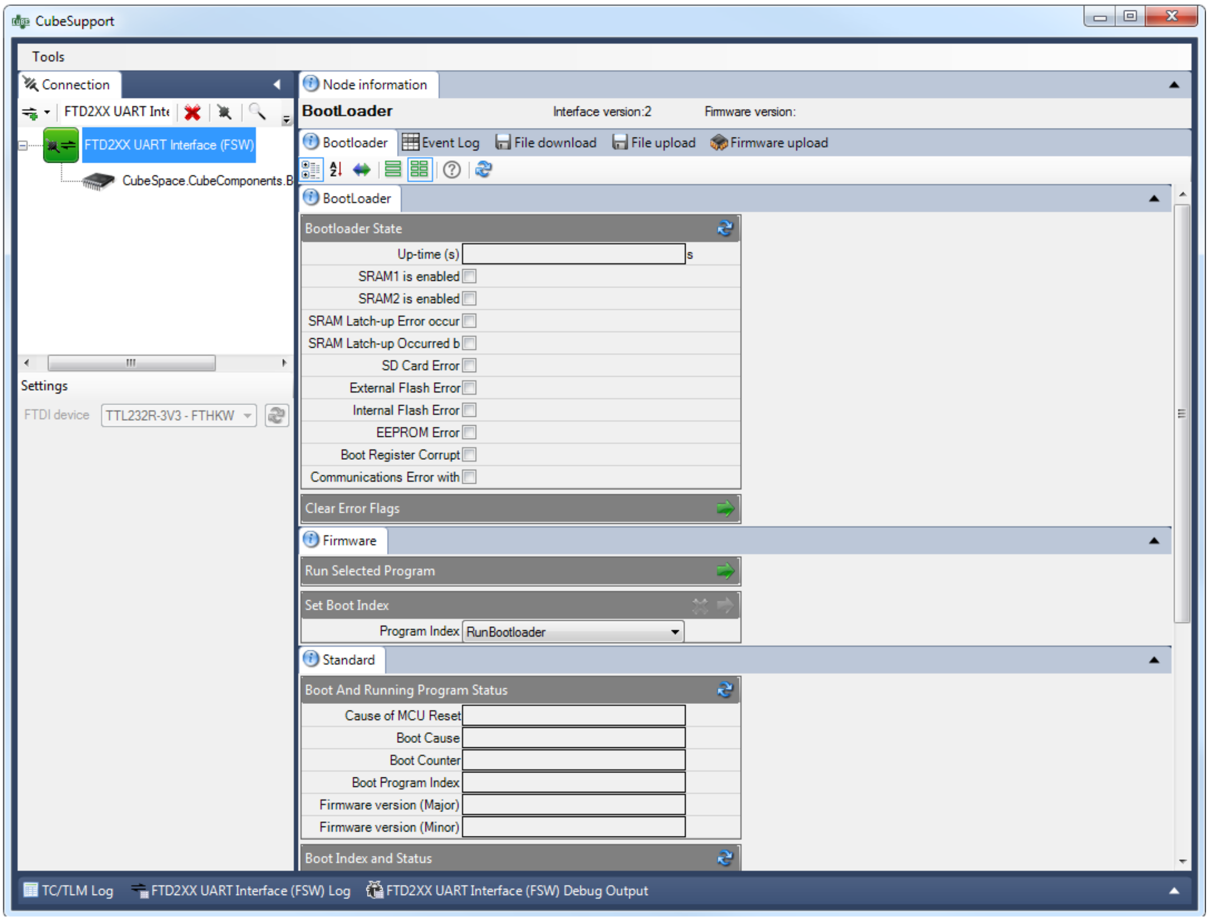
5.3 Connecting to the ADCS through CubeSupport

After applying power to the ADCS unit, select the appropriate option from the *FTDI Interface* drop-down list on the *Connection Settings* part of the CubeSupport UI. It may be necessary to press the refresh button to update the selection. If you have only one GSE cable connected to the PC there will only be one available selection.

Press the *Connect* button to attempt to connect to the ADCS hardware.

5.4 Bootloader vs. ADCS Application

The ADCS processor makes use of a bootloader. The bootloader will be active in the first 5s after applying power to the unit. If the connection through CubeSupport is attempted within 5s of powering on the system, the bootloader will remain active and the CubeSupport UI will show the bootloader interface.



BootLoader Interface version:2 Firmware version:

FTDI device TTL232R-3V3 - FTHKW

Program Index RunBootloader

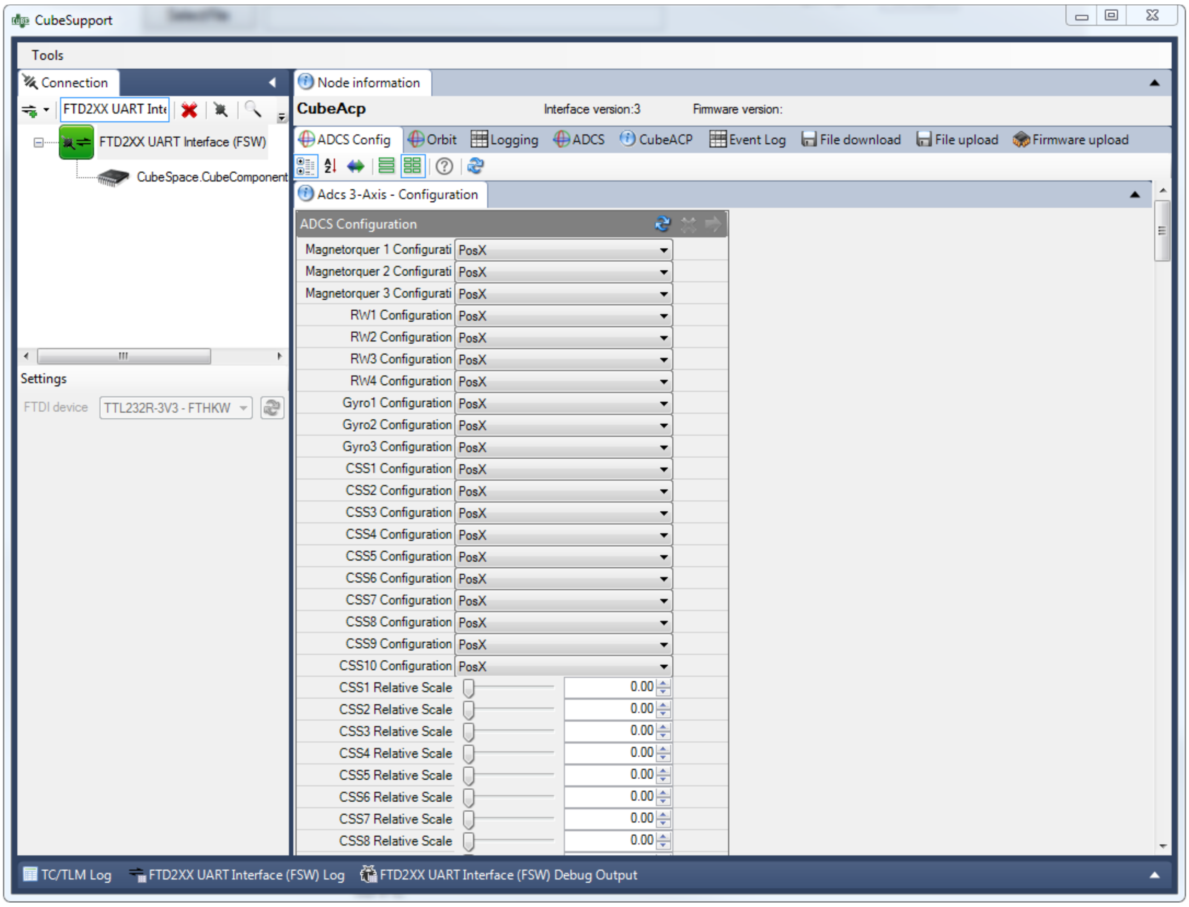
Figure 3 – Bootloader UI

The UI for the bootloader application has a number of tabs running along the top of the *User controls* area.

Bootloader	Houses buttons to retrieve and set the bootloader settings
Event Log	Obtain the log for major activities that has taken place
File Download	Allows the user to initiate file downloads from the SD card
File Upload	Allows the user to initiate file uploads to the SD card
Firmware Upload	Uploading new firmware

If the main ADCS program is allowed to start (after 15s or longer after reset or power-on) before attempting to connect using CubeSupport, the user interface will initialise to the main ADCS interface.

The UI for the main ADCS application has a number of tabs running along the top of the *User controls* area.



FTDI device TTL232R-3V3 - FTHKW

Magnetorquer 1 Configurati PosX
Magnetorquer 2 Configurati PosX
Magnetorquer 3 Configurati PosX
RW1 Configuration PosX
RW2 Configuration PosX
RW3 Configuration PosX
RW4 Configuration PosX
Gyro1 Configuration PosX
Gyro2 Configuration PosX
Gyro3 Configuration PosX
CSS1 Configuration PosX
CSS2 Configuration PosX
CSS3 Configuration PosX
CSS4 Configuration PosX
CSS5 Configuration PosX
CSS6 Configuration PosX
CSS7 Configuration PosX
CSS8 Configuration PosX
CSS9 Configuration PosX
CSS10 Configuration PosX
CSS1 Relative Scale 0.00
CSS2 Relative Scale 0.00
CSS3 Relative Scale 0.00
CSS4 Relative Scale 0.00
CSS5 Relative Scale 0.00
CSS6 Relative Scale 0.00
CSS7 Relative Scale 0.00
CSS8 Relative Scale 0.00

Figure 4 – ADCS application specific UI

The UI for the main ADCS application has a number of tabs running along the top of the *User controls* area.

ADCS Configuration Houses buttons to retrieve and set the ADCS configurations

Orbit Houses controls to change the orbit parameters within the ADCS

Logging Allows the user to setup logging by the ADCS application on the on-board SD card or through UART channel

ADCS Contains all the controls to retrieve and send telecommands to the ADCS

CubeACP Allows the user to obtain and set house-keeping and timing information.

Event Log Obtain the log for major activities that has taken place

File Download Allows the user to initiate file downloads from the SD card

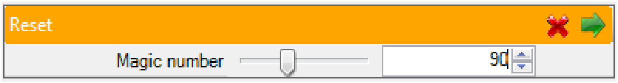
File Upload Allows the user to initiate file uploads to the SD card

Firmware Upload Uploading new firmware

6. Uploading New Firmware

Using the CubeSupport interface it is possible to program new application code into various code areas. To place a new program into flash memory, perform the following steps:

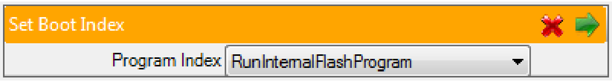
1. Power cycle the CubeComputer or perform a reset from the ADCS application by setting the *Magic number*, found in the *CubeACP* tab, to 90. Click the transmit button (green arrow). You will then receive a timeout error message, indicating the ADCS has reset.



Reset ✖ ➡

Magic number 90

2. Wait for 3-4s
3. Connect to the CubeComputer through the UART. The CubeSupport user interface should display the bootloader interface as shown in Figure 3.
4. Go to the *Firmware Upload* tab.
5. Press the *Refresh* button (🔄). This retrieves the current applications available within flash memory.
6. Select the External Flash position where the new firmware needs to be placed, and press the erase file segment button (✖) to remove the current content.
7. In the *Local File System* section browse to the location of the new binary. Select the binary along with the External Flash position. An Upload button (<< Upload) will appear.
8. Press the Upload button. This will initiate a file upload procedure. Wait until the *File Synchronization Status* indicates "Local and remote files match".
9. Select the new file within the External Flash position and select the *Copy into Internal Flash* button (⇥).
10. After this operation is finished, return to the *Bootloader* tab. In the *Set Boot Index* box select the *RunInternalFlashProgram* as the program index and press the transmit button.



Set Boot Index ✖ ➡

Program Index RunInternalFlashProgram

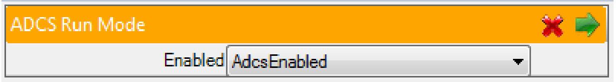
11. Power cycle the CubeComputer or perform a reset from the Bootloader application by setting the *Magic number* to 90 and transmitting.
12. Wait at least 5s before connecting to the CubeComputer UART using CubeSupport. Verify that the CubeSupport user interface is set up for the ADCS application.

7. ADCS Functionality

The ADCS specific functionality of the module is exercised through the *ADCS* tab of the main *User controls* area. The ADCS Reference Manual [Ref 4] should be consulted for detailed information about the ADCS functions.

7.1 Mode selection

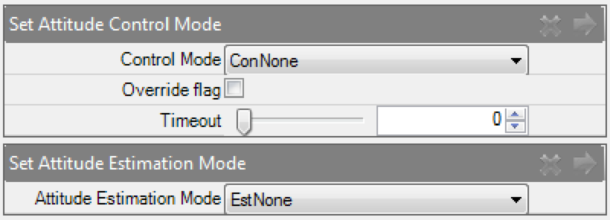
The *ADCS Mode* group of controls allows the user to place the ADCS module in a specific estimation and control state. It also allows the user the activate or de-activate the ADCS processing loop using the *ADCS Run Mode* setting. For any ADCS actions to be performed or telemetry to be returned, the module has to be in the *Enabled* run mode.



ADCS Run Mode

Enabled AdcsEnabled

Figure 5 – Setting of ADCS run mode



Set Attitude Control Mode

Control Mode ConNone

Override flag □

Timeout 0

Set Attitude Estimation Mode

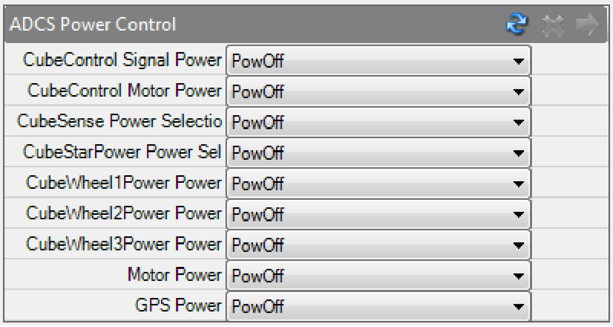
Attitude Estimation Mode EstNone

Figure 6 – Setting of attitude control and estimation mode

Certain transitions of estimation and control modes will not be allowed. See the User Manual [Ref 3] for more detail.

7.2 Sub-system power selection

The power to the sub-systems can be controlled by the *ADCS Power Control* controls.



ADCS Power Control

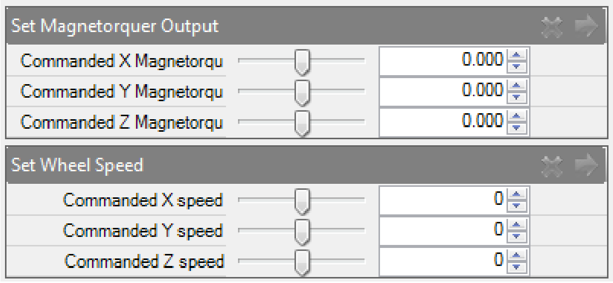
CubeControl Signal Power	PowOff
CubeControl Motor Power	PowOff
CubeSense Power Selectio	PowOff
CubeStarPower Power Sel	PowOff
CubeWheel1Power Power	PowOff
CubeWheel2Power Power	PowOff
CubeWheel3Power Power	PowOff
Motor Power	PowOff
GPS Power	PowOff

Figure 7 – Power selection of sub-systems

7.3 ADCS Commands

7.3.1 Actuator commands

It is possible to send actuator commands to the ADCS through the CubeSupport UI. This is done from the Actuator Commands control group.



Set Magnetorquer Output

Commanded X Magnetorqu 0.000

Commanded Y Magnetorqu 0.000

Commanded Z Magnetorqu 0.000

Set Wheel Speed

Commanded X speed 0

Commanded Y speed 0

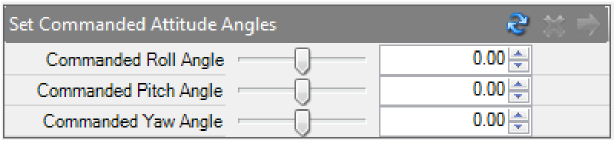
Commanded Z speed 0

Figure 8 – Actuator commands

For the ADCS unit to act on these commands, the run mode has to be set to *Enabled*, and the control mode must be *None*. Power to the CubeControl MCUs and CubeWheels must be set to *On*.

7.3.2 Reference Attitude

The ADCS unit allows for the pitch, roll and yaw angles to be controlled to a reference value if the module is in the correct control mode. This reference value can be specified through the CubeSupport UI.



Set Commanded Attitude Angles

Commanded Roll Angle 0.00

Commanded Pitch Angle 0.00

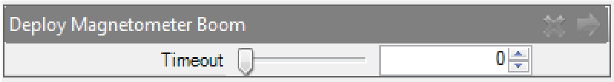
Commanded Yaw Angle 0.00

Figure 9 – Setting of the reference attitude

7.3.3 Magnetometer deployment

This function is provided to test the deployment interface. Activating the magnetometer boom deployment while the magnetometer is connected to the ADCS module will cause the magnetometer boom to deploy!

The magnetometer boom deployment can be activated through the CubeSupport UI by pressing the *Deploy* button after selecting an appropriate time-out.



Deploy Magnetometer Boom

Timeout 0

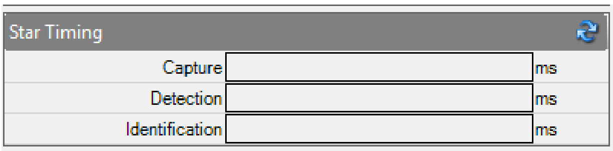
Figure 10 – Deployment of magnetometer command

7.3.4 Other commands

Many other commands are also available and are mostly available at the top of the ADCS tab. Refer to the Reference manual [Ref 4] for other ADCS related telecommands.

7.4 Telemetry

Refer to the Reference Manual [Ref 4] for all the ADCS related telemetries. To obtain a certain telemetry, go to the telemetry block and press the refresh button. This will initiate a telemetry request and retrieve the information from the ADCS application.



Star Timing

Capture		ms
Detection		ms
Identification		ms

Figure 11 – Telemetry request block with refresh button

In order to return valid telemetry, the sub-system from which the telemetry is ultimately measured should be powered on. Consult the Reference Manual [Ref 4] for more information on which sub-system supplies which telemetry.

7.5 UART Telemetry Logging

When interfacing to the ACP through the CubeComputer UART, frequent telemetry sampling such as described above can be avoided by setting up the logging of the output to the UART. This is the preferred way of obtaining telemetry when using the UART as communications interface. Navigate to the Logging tab for the logging settings.

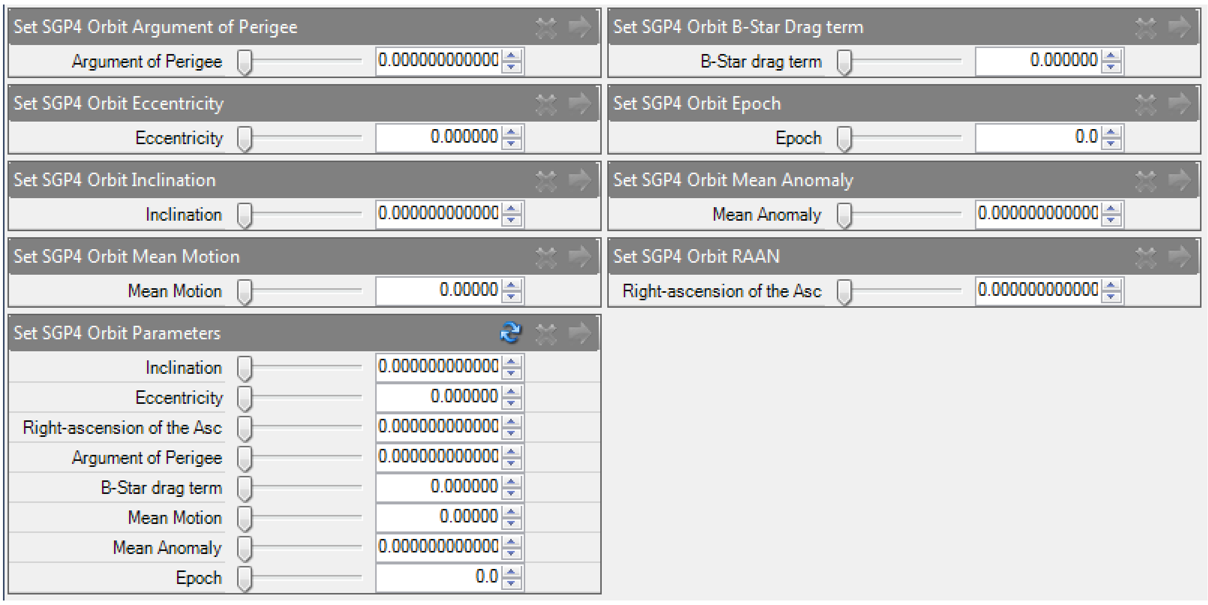
Logging to UART is enabled the same way as the telemetry logging to SD card. Once set up, log telemetry will arrive without the need for telemetry requests.

While unsolicited UART telemetry is being received by CubeSupport, the telemetry windows on the ADCS tab of CubeSupport will automatically update. CubeSupport will also save the received telemetry to a file, the same as when telemetry is explicitly requested.

8. Configuration

The ADCS configuration and orbit parameters can be adjusted through the CubeSupport UI under the *ADCS Config* and *Orbit* tabs of the main *User controls* area.

8.1 Orbit elements



Set SGP4 Orbit Argument of Perigee
Argument of Perigee: 0.000000000000

Set SGP4 Orbit B-Star Drag term
B-Star drag term: 0.000000

Set SGP4 Orbit Eccentricity
Eccentricity: 0.000000

Set SGP4 Orbit Epoch
Epoch: 0.0

Set SGP4 Orbit Inclination
Inclination: 0.000000000000

Set SGP4 Orbit Mean Anomaly
Mean Anomaly: 0.000000000000

Set SGP4 Orbit Mean Motion
Mean Motion: 0.00000

Set SGP4 Orbit RAAN
Right-ascension of the Asc: 0.000000000000

Set SGP4 Orbit Parameters
Inclination: 0.000000000000
Eccentricity: 0.000000
Right-ascension of the Asc: 0.000000000000
Argument of Perigee: 0.000000000000
B-Star drag term: 0.000000
Mean Motion: 0.00000
Mean Anomaly: 0.000000000000
Epoch: 0.0

Figure 12 – Orbit configuration

The current orbit parameters can be requested. Each orbit parameter can be set individually or as a complete packet. After setting the parameters, these values can be saved into flash memory by sending the telecommand in the *CubeACP* tab. Details about the configuration settings can be found in the Reference Manual [Ref 4].

Please refer to CubeSpace when these are set.

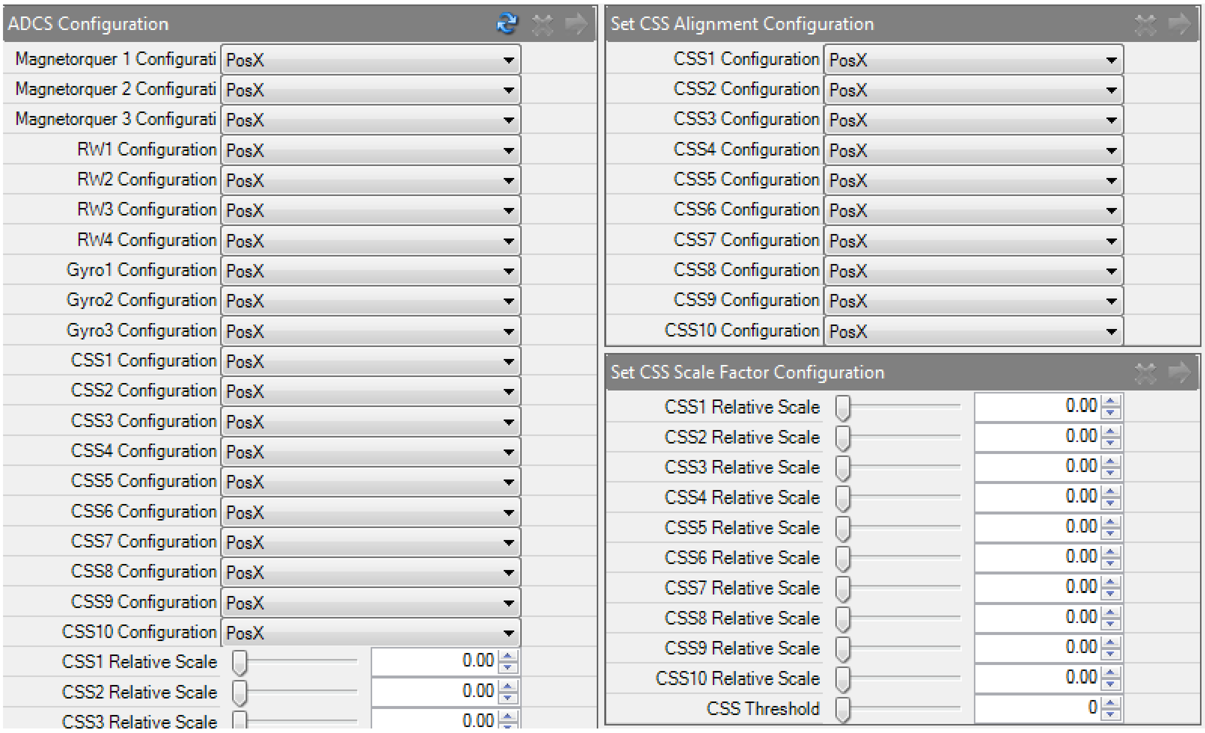
Save Orbit Parameters

Figure 13 – Saving orbit parameters in flash

8.2 ADCS Configurations

The same approach applies to the ADCS configuration. The current configuration can be requested. Each configuration section can be set individually or as a complete packet. After setting the parameters, these values can be saved into flash memory by sending the telecommand in the *CubeACP* tab. Details about the configuration settings can be found in the Reference Manual [Ref 4].

Please refer to CubeSpace when these are set.



ADCS Configuration

Magnetorquer 1 Configurati	PosX
Magnetorquer 2 Configurati	PosX
Magnetorquer 3 Configurati	PosX
RW1 Configuration	PosX
RW2 Configuration	PosX
RW3 Configuration	PosX
RW4 Configuration	PosX
Gyro1 Configuration	PosX
Gyro2 Configuration	PosX
Gyro3 Configuration	PosX
CSS1 Configuration	PosX
CSS2 Configuration	PosX
CSS3 Configuration	PosX
CSS4 Configuration	PosX
CSS5 Configuration	PosX
CSS6 Configuration	PosX
CSS7 Configuration	PosX
CSS8 Configuration	PosX
CSS9 Configuration	PosX
CSS10 Configuration	PosX
CSS1 Relative Scale	0.00
CSS2 Relative Scale	0.00
CSS3 Relative Scale	0.00

Set CSS Alignment Configuration

CSS1 Configuration	PosX
CSS2 Configuration	PosX
CSS3 Configuration	PosX
CSS4 Configuration	PosX
CSS5 Configuration	PosX
CSS6 Configuration	PosX
CSS7 Configuration	PosX
CSS8 Configuration	PosX
CSS9 Configuration	PosX
CSS10 Configuration	PosX

Set CSS Scale Factor Configuration

CSS1 Relative Scale	0.00
CSS2 Relative Scale	0.00
CSS3 Relative Scale	0.00
CSS4 Relative Scale	0.00
CSS5 Relative Scale	0.00
CSS6 Relative Scale	0.00
CSS7 Relative Scale	0.00
CSS8 Relative Scale	0.00
CSS9 Relative Scale	0.00
CSS10 Relative Scale	0.00
CSS Threshold	0

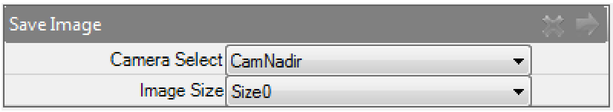
Figure 14 – ADCS configuration

9. Images and Logs

The CubeSupport UI allows the user to capture images using the CubeSense and CubeStar cameras and log telemetry internally and save it to the SD card on the CubeComputer. These files (TLM log files and image files) can then be downloaded to the PC.

9.1 Images

To initiate a new image capture ensure that either the CubeStar or CubeSense is switched on. Under the *ADCS* tab a new capture can be initiated by sending a *Save Image* command. Select the camera and image size. The image size is a numbered list where Size0 is the largest resolution image and Size4 the smallest.



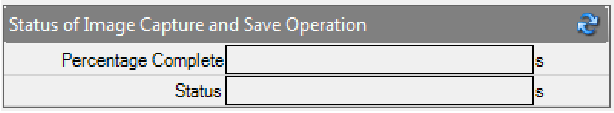
Save Image

Camera Select: CamNadir

Image Size: Size0

Figure 15 – Image Capture and Save command

The save image process is broken down into a capture step and then a download step. The capture step will take a number of seconds before the downloading step starts. The progress of the downloading step can be obtained by requesting the status.



Status of Image Capture and Save Operation

Percentage Complete: s

Status: s

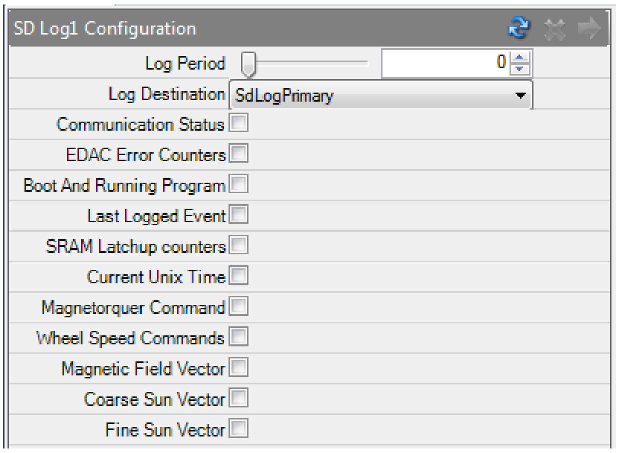
Figure 16 – Save status

After the percentage complete reaches 100 the image is download is complete and the image is saved on the SD card of the CubeComputer.

9.2 TLM log files

The ADCS can log telemetry to the SD card for future download, and to the UART for telemetry sampling by an external OBC. Logging can be initiated through the CubeSupport UI under the *Logging* tab. The selection is performed by a series of checkboxes. The logging period is adjusted through the up-down selection. After the settings have been transmitted to the ADCS application the logging will commence. To stop the logging process, simply transmit a logging option with a 0 second logging period.

If the SD card logging is selected, a log file is automatically created on the on-board SD card. In the case of UART logging the log file is also automatically created and available in the directory of the CubeSupport application.



SD Log1 Configuration

Log Period 0

Log Destination SdLogPrimary

Communication Status □

EDAC Error Counters □

Boot And Running Program □

Last Logged Event □

SRAM Latchup counters □

Current Unix Time □

Magnetorquer Command □

Wheel Speed Commands □

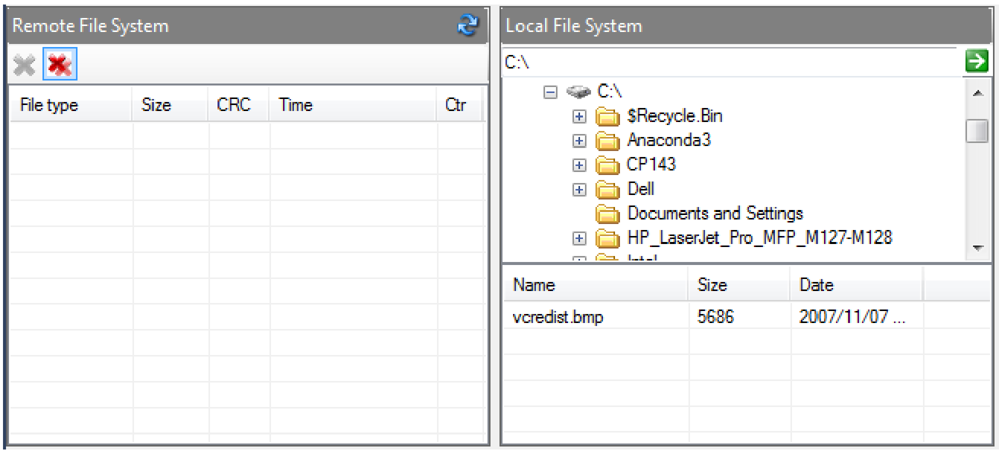
Magnetic Field Vector □

Coarse Sun Vector □

Fine Sun Vector □

Figure 17 – Initiate logging command

9.3 SD card file management



Remote File System

File type	Size	CRC	Time	Ctr

Local File System

C:\

- C:\
  - $Recycle.Bin
  - Anaconda3
  - CP143
  - Dell
  - Documents and Settings
  - HP_LaserJet_Pro_MFP_M127-M128

Name	Size	Date
vcredist.bmp	5686	2007/11/07 ...

Figure 18 – File download from SD card

Files on the SD card are managed under the *File download* tab. To get the current list of files on the SD card, press the 🔄 button. The list will update to show all the available files. To perform an operation on a file, first select the row with the relevant file. To download the selected file to PC, select the root in the *Local File System* box. After pressing the download button the final path to where the file should be saved and the filename can be selected. To delete the file from the SD card, press the ✖ button.

The download process is complete when the message is visible that the local and remote file are identical.

10. I2C Interface Option

The CubeSupport has the functionality to interface with a slave through an I2C interface cable. This function enables all the testing that is done through UART to be repeated through the I2C bus. The I2C interface is a bus topology and this function assumes that the CubeSupport with the required hardware is the only master on the bus and do not support multi-master functions.

10.1 Hardware Connection

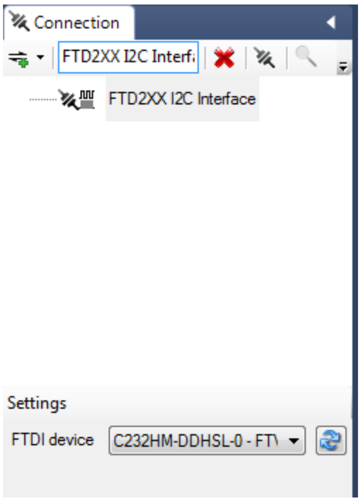
The I2C interface is created with the use of the FDTI C232HM cable (available at any large electronic distributor). This cable contains 10 colour coded wires with PCB sockets. The cabling and connection options for I2C can be seen in die document available at this site.

Note that Table 3.4 within this document describe the connection required for I2C use and that there are two pins required for the data line. The connections required for the ADCS bundle are that the Orange cable should be connected to H1-43 (the CLK line of the system bus), the Yellow and Green cable should be connected to H1-41 (the SDA line of the system bus) and the Black line connected to H2-29, 30, 32 (the Ground of the CubeComputer).

This cable must be plugged into a PC with a CubeSupport available. After turning on the device with the cable inserted and plugged into a PC just ensure all the I2C lines are at 3V3. If this is not the case, add the necessary pull-up resistors and voltage source.

10.2 CubeSupport Settings

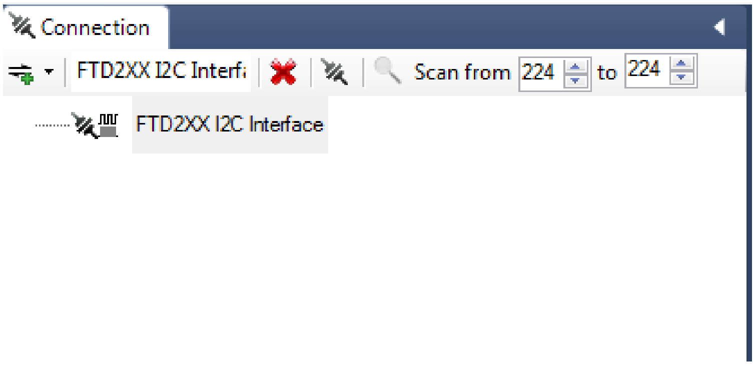
Open CubeSupport, and instead selecting the FTD2XX UART Interface (FSW) connection option, rather select the FTD2XX I2C Interface. In the Settings section below ensure that you select the device ID which contains the C232 code.



FTDI device C232HM-DDHSL-0 - FT\

Figure 19 – Selecting the I2C interface in CubeSupport

Increase the width of the Connection window to reveal the scan addresses option.

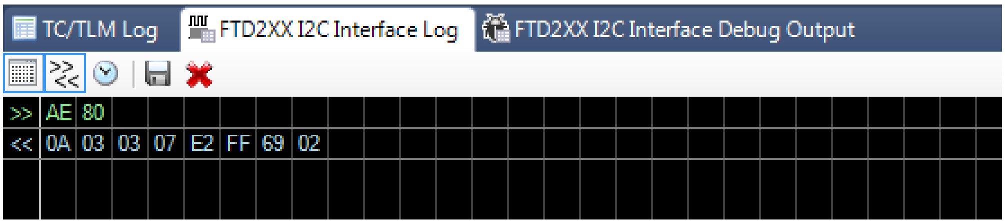


Scan from 224 to 224

Figure 20 – Scan option within CubeSupport

Type in the I2C addresses that are present on the I2C bus. Note that this is the 7-bit I2C address of the device and not the 8-bit write address as is normally the case in CubeSpace documentation. This means the CubeACP 0xAE 8-bit address is actually 87 decimal number in the connection settings.

After setting the scanning addresses press the connect button. The FTD2XX I2C Interface Log at the bottom of the screen reveals the transmission and receive information.



```
>> AE 80
<< 0A 03 03 07 E2 FF 69 02
```

Figure 21 – I2C interface log information

The rest of CubeSupport functions the same, except that all telemetry and telecommands are sent through the I2C interface instead of the UART interface.

11. Scripting

Some CubeSpace components will enable the scripting function when connected to the CubeSupport. The scripting function enables the abilities to perform a predefined set of telemetry requests and telecommands. Certain assertions can be defined which test whether certain expected outputs/behaviour have been obtained.

11.1 Running a Script

After successfully connecting to a CubeSpace component through the CubeSupport select the Script Testing tab.

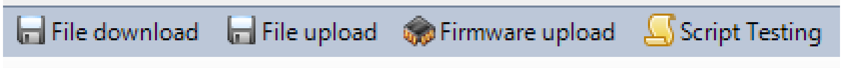


Figure 22 – Script testing tab

This will open the script window as seen below.

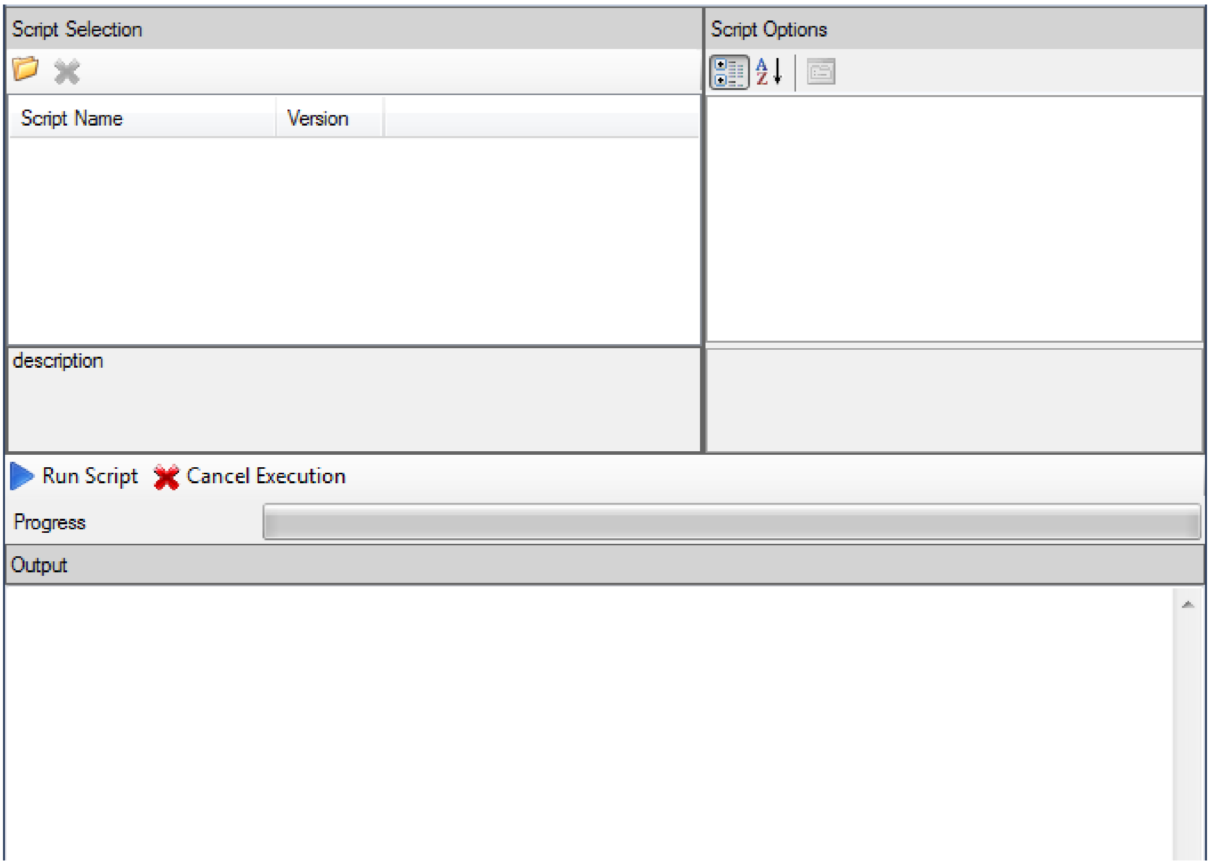


Figure 23 – Scripting window

Press the open button in the top left of this window. This will enable a file browser by which a script can be selected. If the script is in a correct state, the selected script should be visible in the left-side list. A number of scripts can be uploaded at this time.

Select the requested script and then press the Run Script button. This will activate the script by which any output generated by the script will be visible in the Output section of the window. Some scripts may have settings which can be changed before pressing the Run Script button. These option, if any, are visible after selecting the script in the Script Option panel.

11.2 Creating Scripts

This function is currently not available to the user and certain scripts can be requested from CubeSpace.

12. Offline Functionality

The following offline utilities are available from the Tools menu of CubeSupport. These utilities do not require a connection to the ADCS.

12.1 Telemetry log decoder

Downloaded TLM log files are binary files with format as described in the Reference Manual [Ref 4]. CubeSupport includes a convenient utility to convert the downloaded TLM binary files into calibrated CSV files. The function is initiated by selecting the *Decode Telemetry Log* menu item from the Tools menu. A dialog window will appear.

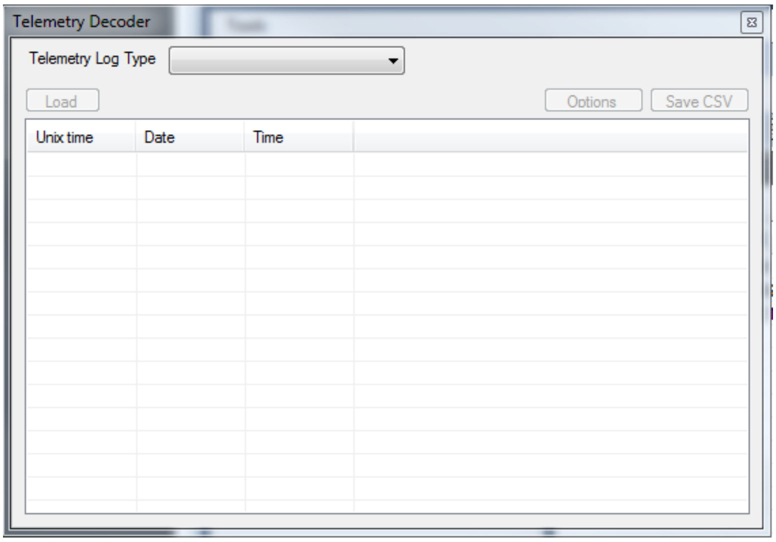


Figure 24 – Telemetry decoder window

Select *CubeACP* in the Telemetry Log Type. Press the Load button and a file dialog will appear asking for the user to choose an input TLM file. The decoded TLM will be displayed in the table after successful loading. Press the Save CSV button to save the decoded telemetry to a target CSV file. CSV output file options can be set when pressing the Options button.