

UKPIK-1

Communication Subsystem

2020-10-02

Ukpik-1

Communication Subsystem

REVISION NUMBER: *0.1*

DATE: *2020-10-02*

COMPILED BY: *Nicholas Mitchell*

CONTRIBUTIONS FROM: *Nicholas Mitchell*

Document Change Record

Issue	Date	Changes Made	Name
0.1	2020-10-02	First Draft	Mitchell, Nicholas

Reference Documents

Insert applicable reference document titles, such as requirements documents

Terms, Definitions, Abbreviations

CSA	Canadian Space Agency

Contents

Requirements.....	1
Architecture and Interface Diagrams.....	4
Functional Operations.....	5
Ground Station and Operations.....	13
Design and Status.....	13
Payload Data Plan	14
Operation Organization	17
RF Licensing Status.....	17
Link Budgeting and Protocols	17
Antenna design	18
Assembly and Integration Plan	18
Test and Verification Plan	18
Schedule and Work Plan for Phase C2 and D.....	21
Datasheets for COTS Components.....	21

Requirements

COMMs Requirements		
Requirement #	Requirement	Parent Req ID
COM-01	The CubeSat shall be able to communicate with ground stations	SY-GR07
COM-02	The CubeSat shall downlink [5] minimum pictures to ground	SY-PA04
COM-03	The CubeSat shall receive commands	SY-GR08, SY-SS08
COM-04	The CubeSat shall downlink telemetry	SY-GR09, SY-SS09
COM-05	The CubeSat shall be able to receive data and talk to ground station	SY-GR08
COM-06	The CubeSat must adhere to licensing restrictions	
COM-07	The CubeSat must not deploy the antenna within the first 30 minutes of deployment	SY-SS31
COM-08	The CubeSat shall communicate via the AX.25 protocol	
COM-09	The CubeSat shall turn off transmitter upon request from Groundstation	
COM-10	The mission design must comply with CCSDS telecommunication standards as outlined in [AD##] (TBD)	SYS-COM-010
COM-11	All necessary RF communications licenses must be obtained prior to launch	SYS-COM-011
COM-12	The CubeSat must be fitted with devices to ensure immediate cessation of its radio emissions by command whenever such a cessation is required, in accordance with the provisions of the ITU Radio Regulations [AD7]	SYS-COM-012
COM-13	A communications link with the satellite must be established when it is at least ## degrees (TBD) above the horizon from the ground segment	SYS-COM-020

COM-14	The CubeSat must be able to receive and execute real-time commands from the ground at all times when the satellite is within view of an operating ground station and at least 5 degrees above the horizon	SYS-COM-021
COM-15	Command data must be uplinked to the spacecraft in S-band Earth Exploration Satellite Service	SYS-COM-030
COM-16	The CubeSat satellite must be protected against incomplete or incorrect transmission of commands	SYS-COM-031
COM-17	An ineffective, non-operating command must exist to test the communication link	SYS-COM-032
COM-18	The command function must support time-tagged commands sufficient to cover a minimum of three days of autonomous operation	SYS-COM-033
COM-19	Science and telemetry data must be downlinked by the spacecraft in S-band Earth Exploration Satellite Service	SYS-COM-040
COM-20	The maximum bit error rate during data downlink must be better than 10^{-5}	SYS-COM-050
COM-21	The telecommunication system must be capable of simultaneously handling telemetry and science data downlinking and uplinked commands i.e. full duplex (TBC)	SYS-COM-060
COM-22	The telecommunication equipment must support the linkage needs of the demo payload as specified in [AD##]. (TBD)	SYS-COM-070
COM-23	The CubeSat must have sufficient communications systems to transfer data from the demo payload back to Earth	SYS-COM-090
COM-24	The communications subsystem must draw less than 6 W during transmission to the ground (TBC)	SYS-COM-110
COM-25	The communications subsystem must have mass no more than 500 g (TBC) including antenna	SYS-COM-120
COM-26	The communications subsystem must have at least one RF inhibit (TBC)	SYS-COM-130

COM-27	The communications subsystem must produce RF power output of no greater than 31.8 dBm (TBC) at the transmitting antenna	SYS-COM-140
COM-28	The communications subsystem must provide a reliable link to the ground station of at least 3 dB margin under nominal contact geometry of a ## degree (TBD) elevation angle.	SYS-COM-150
COM-29	The communications subsystem downlink frequency range must be 2200 – 2290 MHz (TBC) according to the Canadian Frequency Allocation Table	SYS-COM-170
COM-30	The communications subsystem uplink frequency range must be 2025 – 2110 MHz (TBC) according to the Canadian Frequency Allocation Table	SYS-COM-180
COM-31	The communications subsystem must use one of the recommended modulation schemes as per CCSDS 401.0-B-29 [AD8]	SYS-COM-190
COM-32	The communications subsystem downlink transmission rate must be at least 250 kbps (TBC)	SYS-COM-200
COM-33	The communications subsystem uplink transmission rate must be at least 120 kbps (TBC)	SYS-COM-210
COM-34	The XYZ communications subsystem must use Two Line Element values for orbit propagation in order to define RF link periods	SYS-COM-220
COM-35	The XYZ satellite must provide command channel (uplink) authentication to ensure that only authorized mission control centers have access to the space segment	SYS-COM-230
COM-36	The communications system must utilize ## (TBD) –hand polarization for uplink and ## (TBD) –hand polarization for downlink	SYS-COM-240
COM-37	All intentional receivers must be designed to minimize their susceptibility to out-of-band RF signals found in the surrounding environment, including emissions from the on-board transmitters	SYS-COM-250
COM-38	All intentional receivers must be designed to withstand 160 dBμV/m at any frequency below 40 GHz at the receiving antenna (TBC), without permanent degradation to performance, reliability or life	SYS-COM-251

COM-39	All intentional receivers must be designed to operate without degradation in the presence of ground, sea, and space-based radars as defined in [RD3]	SYS-COM-252

Architecture and Interface Diagrams

Note that all impedances are 50 ohms unless otherwise specified.

Functional Block Diagram Endurosat Type II Transceiver

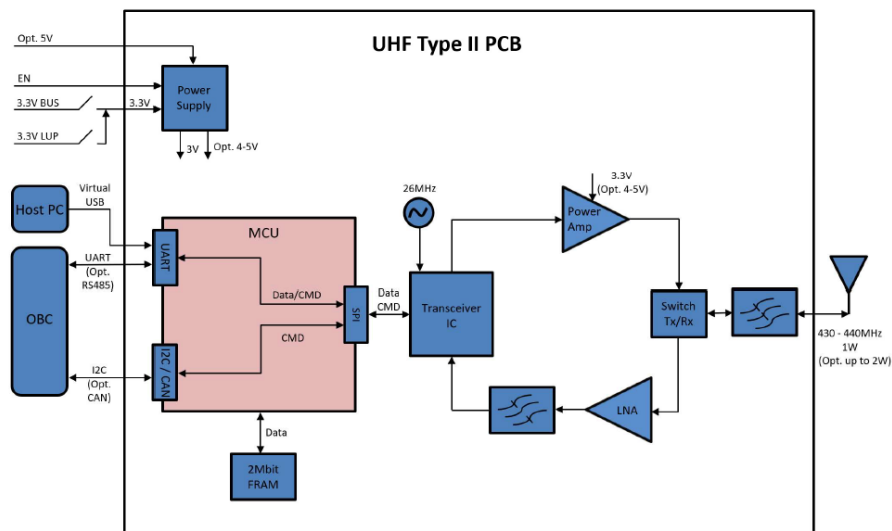
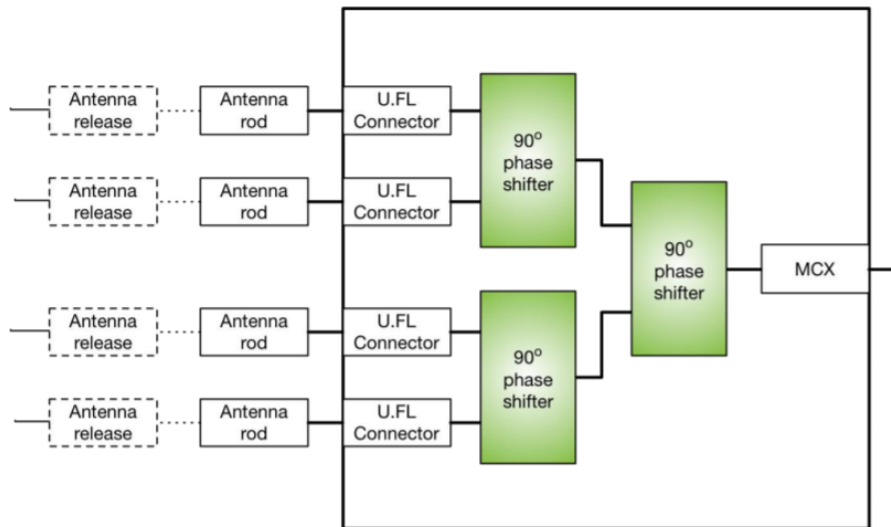


Figure 3: Functional Block Diagram of the UHF Transceiver Type II

Connector: MMCX to antenna

Antenna Block Diagram:

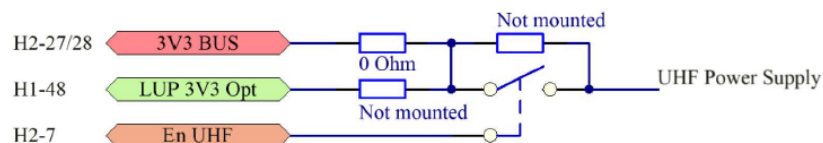


Note that antenna release mechanism is not included inside the antenna purchase.



Functional Operations

Power Supply Diagram:



Enable UHF:

ON 1V – 5.5V

OFF 0V – 0.3V

Figure 2: Power Supply Diagram

H2-7 is the enable pin which can be controlled by the OBC. The enable pin will cut off power in the case that the transceiver must be shut down.

Data Transfer Modes Designed:

1. Downlink Telemetry and Housekeeping data
 - a. Instantaneous Housekeeping data
 - b. List files on Cubesat
 - c. Downlink telemetry and housekeeping data from x time.
2. Downlink Images
 - a. Send image thumbnails
 - b. send specific image
3. Schedule Image Capture
 - a. Capture image(s) with at x time.
4. Onboard Computing Update Command
 - a. Can include system image or select files on Cubesat.
5. Stop all communications (IARU requirement as well)
6. Uplink Telecommands
 - a. Reformat SD card
 - b. Update TLE values
 - c. Delete file(s)
 - d. Reboot system
 - e. Schedule rotation of Cubesat for power generation purposes.

Commented [MK1]: Below it said "Patch command" wording discrepancies, some places say update some say patch. Please check comment in "OBC Update Command" heading

Frame Description:

Start I-Frame:

- Ground station transmits to satellite to Start Information Sending Frame that sets up parameters such as:

- Command for current transmission
- Maximum end transmission time
- Next time to transmit

TT&C I-Frame:

- Frame transmitted by satellite containing Telemetry, Tracking, and Command data

Contains the following sensor information:

- OBC time
- Battery voltages and currents
- Temperature Sensors
- Solar Panel voltages + currents
- Magnetorquer current/volt
- Magnetic compass sensor
- Gyroscope data
- Uptime between last update to the epoch
- Antenna deployment currents
- Payload health information (tbd)
- Etc...

Commented [NM2]: If Steven, Alexis or Matt have more feel free to add

Acknowledge S-Frame (ARQ)

- Supervisory frame used to transmit from the ground station to the satellite to acknowledge frames received and to request next action to be conducted

4 possible actions:

- Acknowledge send next frame
- Acknowledge do not send next frame
- Selective Reject a frame for retransmission
- Reject all frames, retransmit all

VR camera I-Frame

- Information frame transmitting images from the satellite

2 modes to send images in varying qualities:

- Primary mode to send back 12-bit JPEG images at 3000x3000 pixels per image at 13.5MB
- Secondary mode sends back images with up to 15x compression <1MB per image using lower bit depth

OBC (Onboard Computing) Update I-Frame

- Information frame transmitting from ground station up to satellite to update or patch onboard computer flight software

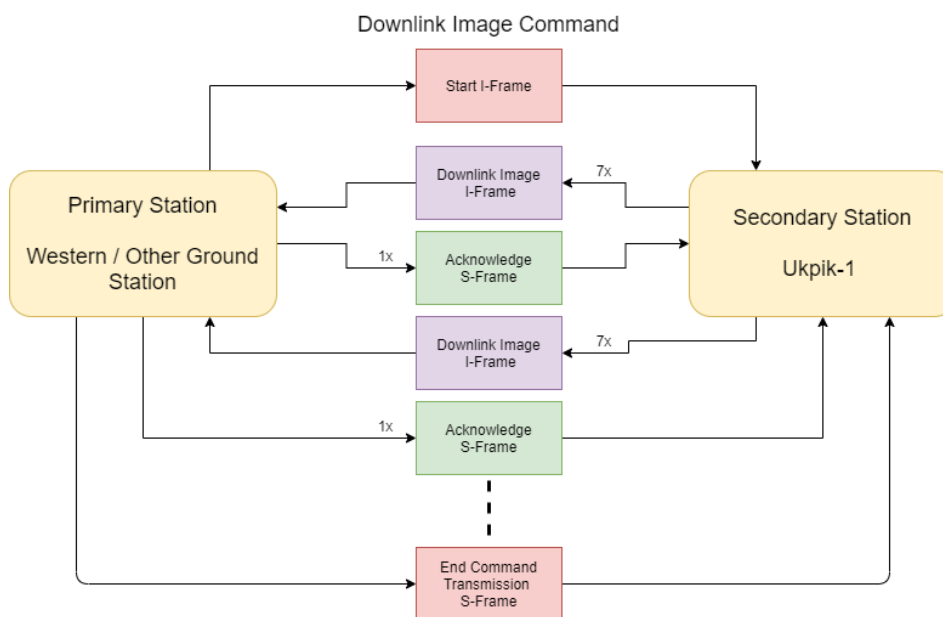
End Transmission S-Frame:

There will be 2 types of end transmission S-Frames:

1. Supervisory frame used to transmit to the Satellite to end transmission at the end of the current command.
2. Supervisory frame used to transmit to the Satellite to end transmission at the end of a transmission period. This happens when the satellite is no longer in communication range of the groundstation.

Downlink Image Command:

Diagram:



Description:

1. 1x Start I-Frame transmitted from ground station to begin a communication sequence with an image transmission command (other functionalities work in similar manor)
2. 7x Camera downlink image I-Frames w from satellite to collect payload camera data.
3. 1x acknowledge S-Frame transmitted from ground station to acknowledge received info and decide what next action is based on the 4 possible actions provided in [HERE](#)
4. Continue with steps 2 and 3 until done with command or time to transmit runs out.
5. 1x End Command Transmission S-Frame that will end the current command sequence.

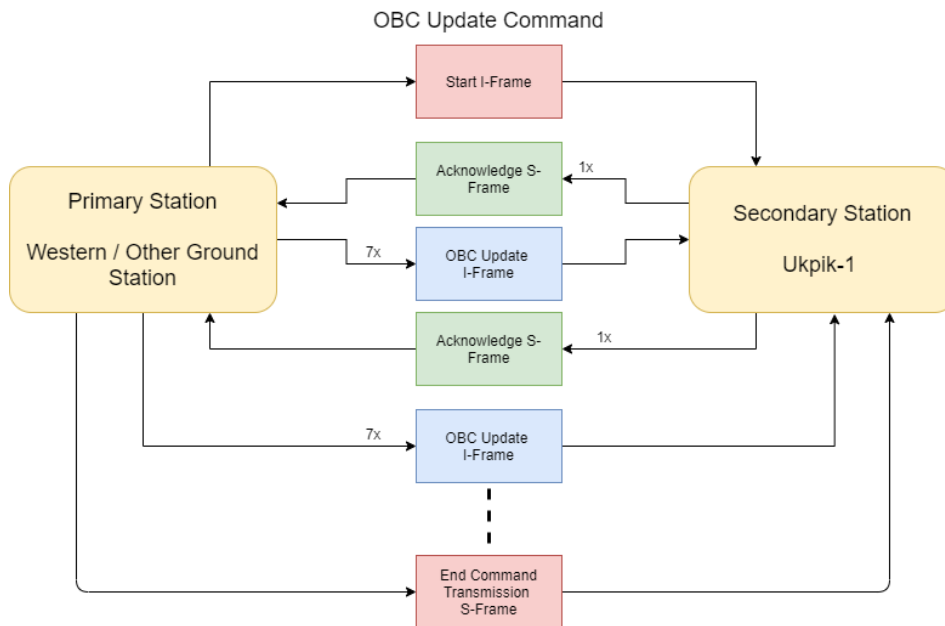
Commented [AP3]: Where did we get the number that we're only sending 7 frames?

Commented [MK4R3]: TBD will be discussed. 7 frames is not a finalized number

Commented [MK5]: Couldn't all these last statements realistically be the end of transmission period type of end transmission command?

OBC Update Command:

Diagram:



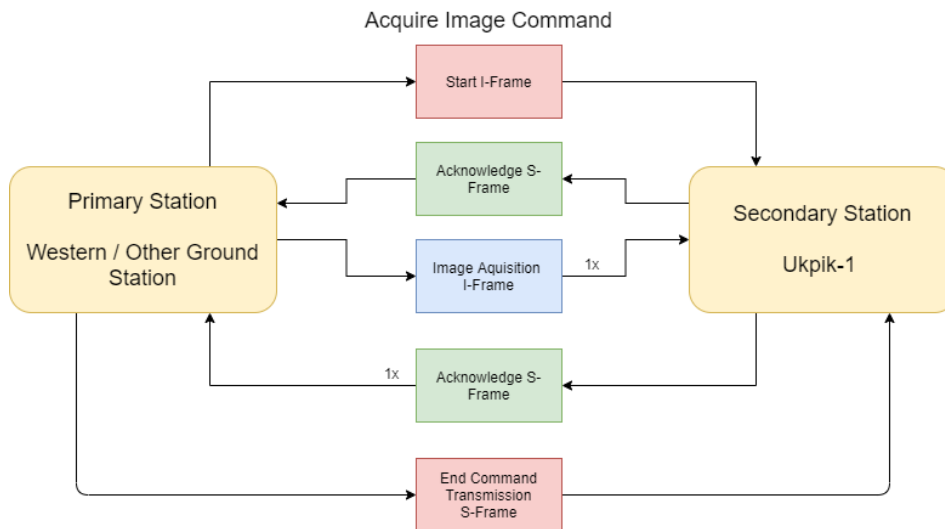
Description:

- 1x Start I-Frame transmitted from ground station to begin a communication sequence with OBC patch command
- 1x acknowledge S-Frame transmitted from satellite to acknowledge received info and decide what next action is based on the 4 possible actions provided in [HERE](#)
- 7x OBC Update I-Frames transmitted by ground station to the satellite to provide data for OBC Patch
- Continue with steps 2 and 3 until done with command or time to transmit runs out.
- 1x End Command Transmission S-Frame that will end the current command sequence.

Commented [MK6]: Patch and update both used in this paragraph, please elaborate.

Image Acquisition Command:

Diagram:

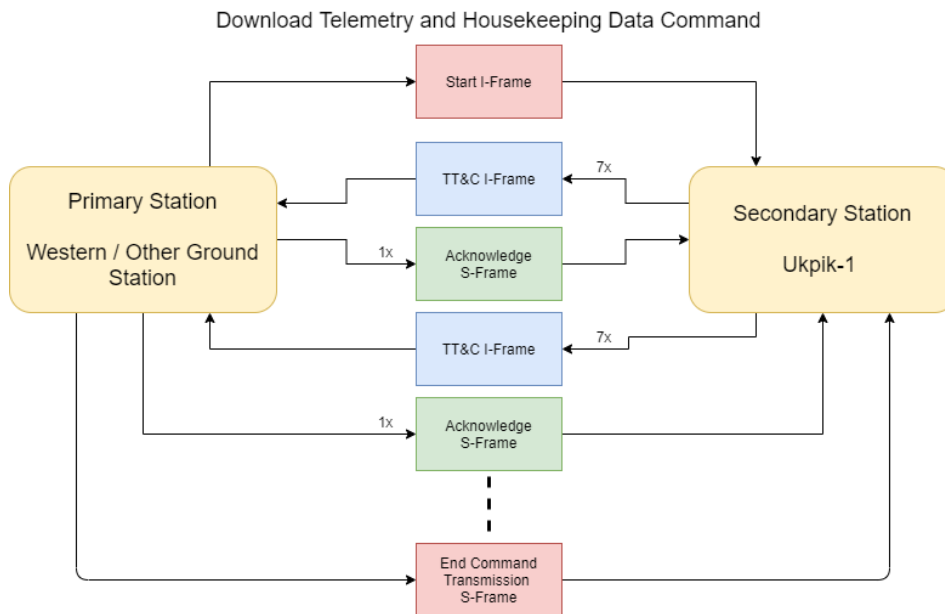


Description:

1. 1x Start I-Frame transmitted from ground station to begin a communication sequence with Acquire Image command
2. 1x acknowledge S-Frame transmitted from satellite to acknowledge received info and decide what next action is based on the 4 possible actions provided in [HERE](#)
3. 1x Image acquisition frame which will detail the next time to image.
4. 1x acknowledge S-Frame transmitted from satellite to acknowledge received info and decide what next action is based on the 4 possible actions provided in [HERE](#)
5. 1x End Command Transmission S-Frame that will end the current command sequence.

Telemetry and Housekeeping Data Command:

Diagram:



Description:

1. 1x Start I-Frame transmitted from ground station to begin a communication sequence with a telemetry and tracking transmission command (other functionalities work in similar manor)
2. 7x TT&C I-Frames transmitted from satellite to collect housekeeping data.
3. 1x acknowledge S-Frame transmitted from ground station to acknowledge received info and decide what next action is based on the 4 possible actions provided in [HERE](#)
4. Continue with steps 2 and 3 until done with command or time to transmit runs out.
5. 1x End Command Transmission S-Frame that will end the current command sequence.

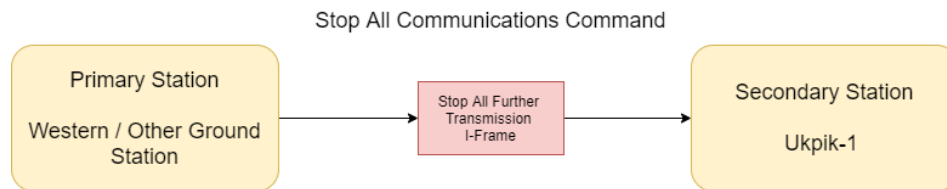
Commented [AP7]: Where did we get the number that we're only sending 7 frames?

Commented [MK8R7]: TBD will be discussed. 7 frames is not a finalized number

Stop Communication Command:

Diagram:

Description:



1. 1x Start I-Frame transmitted from ground station to send Stop Communication command

WILL PERMENANTLY CEASE TRANSMISSION ABILITIES

Ground Station and Operations

Design and Status

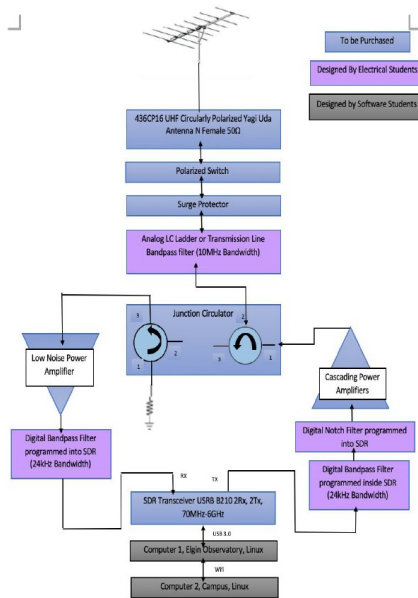


Figure 1: Revised RF Chain

The current design status of the groundstation is that the components have been mostly selected except for the specific power amplifier on the transmit side. The specification that needs to be fit for the power amplifier is that it must achieve a gain of 44.25 dB in order to emit a 1W signal if the SDR is operating at -40 dBm.

TRANSMITTING ARM (Tr) TO CENTRAL								
COMPONENT	SDR	Bandpass (Digital)	Cables	Junction Circulator	Bandpass (Analog)	Surge Protector	Polarized Switch	Antenna
INSERTION LOSS (dBm)	0	0	0.401	0.3	3	0.35	0.2	0
OUTPUT POWER (dBm)	-40	-40	-40.401	-40.701	-43.701	-44.051	-44.25	-44.25

Table 2: Insertion Loss of Components on the Tr Chain

Otherwise the groundstation has been priced out and the components have been selected. The next stage would be to purchase the components that fit as well as start working on an S Band receiver for the groundstation. The S band structure will be designed next for the system.

Payload Data Plan

Commands (rough descriptions, detailed payload bytes lower down):

- List files
 - Just the command, no other payload
- Schedule capture
 - Timestamp, name, size?
- Schedule rotate
 - Timestamp, direction (3 integers for now)
 - Need to ask if we have a format for this yet or how we should go about it
- Delete file
 - File name
- Reformat SD
 - Command only, no payload
- Update TLE values
 - For now just epoch. Do we even need anything else? Perhaps for rotation?
- Get file (CS -> GS) [This is basically just a bulk transmission from a start location]
 - File name, transmission time, start location
 - This will just be the bulk transmission, will use a different command for grabbing missing/bad frames
- Get individual frames (Both ways!) [This is used after a bulk transmission for any missing]
 - File name, list of frames
- Send file (GS -> CS) [Basically tells CS to disable interrupts, get ready to receive a file]
 - Name of file, size of file (knows how many frames), timeout
 - Sets CubeSat into receiving mode (sets flag, stores file name). When set, will read data frames received into the specified file (ignores them otherwise).
 - Should probably leave file open and write it in immediately. Frame data will be written into the file at an index that corresponds to the frame ID.
 - CRC check every frame received, add it to a list.
 - Process:
 - GS issues command to switch to receiving mode, tells it the file name and size.
 - CS switches to receiving mode, status frame OK when ready. In this initial mode, freely records data received until the EOT is issued.
 - GS begins transmitting, CS records any missing/bad frames.
 - GS issues end of transmission. CS proceeds to next part in function.
 - If all frames received fine, CS issues status frame OK. Else issues `get_individual_frames` command to GS. CS still in receiving mode.
 - GS begins transmitting missing frames. CS continues to write it into a file, and record any bad ones.

- Again, GS issues EOT request command. CS replies status OK, or again commands for missing frames, and repeats if necessary.
- Request transmission end [Basically tells CS to reply OK, or send command for individual frames]
 - Tells the Cubesat we're done transmitting. Just a command, no other payload. May be repeated by GS until response.
- How about another command for getting a readout of CubeSat status (not from an SD file)– e.g. `get_status`
 - This will gather some sensor data, power level, voltages, SD mount status, and anything else important. Can use in the case of the SD card not working or if we just want a general status readout? No parameters?
- Reboot
 - Causes a system reset, e.g. we want to apply the software patch

In general for file transmission:

- CubeSat to Ground-station (telemetry, images):
 - GS issues `get_file` with parameters (file name, timeout, start location in file). CS Acks, transmits that data until file finished or timeout. Status frame when done.
 - GS was recording if any missed frames, issues the `get_individual_frames` if so. Repeats if necessary.
 - Repeats over several passes if necessary; we know where the last stopped, can increment the start location in `get_file` command.
- Ground-station to CubeSat (software patches):
 - GS issues command (send file) for CS to listen for data (sets receiving flag, preps new file on SD). CS Acks.
 - GS sends all its data, CS records bad/missing frames, then GS requests for transmission to end (EOT).
 - If CS is missing any frames, issues a `get_individual_frames` command to the GS.
 - GS transmits those frames, and again issues an EOT.
 - Once all frames received, CS sends an OK status frame, and transmission ends.

These files are small enough that they should be able to be transmitted in a single pass easily, so don't need to make GS->CS work over multiple passes. If it fails to do so in one pass for some reason, just issue a command to delete the file then try again.

Command payload parameters

Commands in general will use the first byte to indicate the command, however the rest of the payload is used for parameters and is unique to each command.

Schedule image capture

- Command (1 Byte `uint8_t`), Timestamp (4 Bytes `uint32_t`), File name (11 Bytes char) [8.3 format]

- 11 bytes or 8 bytes for the file name? We can auto append '.jpg' in the code. Maybe leave space for 11 and just ignore the last few, or go 8 and the GS can make sure its correct before sending... Ok, 11 to be consistent? Extension really doesn't matter anyways as long as we know what is in the file.

Schedule rotate

- Command (1 Byte uint8_t), Timestamp (4 Bytes uint32_t), Direction (3x 1 Byte uint8_t)

Update TLE

- Command (1 Byte uint8_t), Epoch (4 Bytes uint32_t), Nothing else yet

List files

- Command (1Byte uint8_t)

Delete file

- Command (1 Byte uint8_t), File name (11 Bytes char) [FAT16 8.3 format file name]

Reformat SD

- Command (1 Byte uint8_t)

Get file

- Command (1 Byte uint8_t), File name (11 Bytes char), Start location (3 Bytes uint24_t), Timeout (2 Bytes uint16_t)
- Timeout must be a positive value, and impose a maximum limit (e.g. max pass length)

Get frames

- Command (1 Byte uint8_t), File name (11 Bytes char), List of frame IDs (n* 3 Bytes uint24_t)
- List of n frame with IDs that are 3 bytes each. We can store each as a uint32_t, won't take up too much memory? We do have 192Kb of ram but I still hate dynamic allocation. Perhaps can declare a big list to hold these beforehand but what happens if we're missing a lot of frame s? Although it's probably the only decent solution anyways. Just make its size 2-3x the expected amount.
- If a value encountered in the list is max value for a 3 byte int (all bits 1), then assume that's the end of the list. Perhaps, either don't even extend the payload that far and just end it at the end of the list, or set the rest of the bits in the payload to 1. Honestly probably easier to just make the frame shorter and don't even deal with the extra case if we don't need to. Make it a GS problem.
- May need to issue several of this command in sequence if there is a lot of frames missing. With a 92 byte max payload, 12 bytes for command and file name parameters, that leaves 80 bytes for the list of IDs. If each ID is 3 bytes, we can fit 26 unique IDs per get_frames command.
- If we can squeeze a single byte out of somewhere, we have room for 27 unique IDs per command.

Send file

- Command (1 Byte uint8_t), File name (11 Bytes char), File size in bytes/max ID (4 Bytes uint32_t)

Get status

- Command (1 Byte uint8_t)

Reboot

- Command (1 Byte uint8_t)

Don't know if this is inside the scope of Comms, this is more the groundstation software I believe?

Ground Station Access Time Analysis

Taken from the Orbital Team analysis Powerpoint:

Min	223 (~4 mins)
Mean	346 (~6 mins)
Max	404 (~7 mins)
Monthly total	58784 (~16 hours)
Monthly mean per day	1896 (~32 mins)
Mean passes per day	5 passes/day

Operation Organization

RF Licensing Status

Current status of the licensing is:

- 1) We have contacted our local RAC organisation, and they are aware of our mission.
- 2) We have completed and sent the first IARU application. We are Currently Waiting on Dr. Jayshri Sabarinathan to get her Advanced amateur radio license and apply for the callsign.
- 3) Dr. Jayshri Sabarinathan has already obtained the basic amateur license.

Link Budgeting and Protocols

Link to the most recent link budget: All protocols and assumption are made in comments or in text.

<https://github.com/cubesat-project/CubeSat/tree/master/Ground%20Segment/COMMS%20--%20GS%202019/Link%20Budget>

Antenna design

The antenna is a COTS component that we purchase. Please see below for the link to the datasheet.

<https://gomspace.com/shop/subsystems/communication-systems/nanocom-ant430.aspx>

Assembly and Integration Plan

Assembly:

- 1) Insert transceiver and antenna into cubesat rails.
- 2) Measure distance between antenna and transceiver ports.
- 3) Gather MMCX, MCX connectors and Coax cables with a sharp knife thin knife.
- 4) Cut coax cable to appropriate length (determined by distance between antenna and transceiver port). Make sure cut is at 90 degrees.
- 5) Solder MMCX and MCX connectors to the ends of the cable.
 - a. Test the insertion loss of cable assembly and update link budget with real values.
- 6) Wrap antenna monopoles with burn wires (4x)
- 7) Attach burn wires to custom board 2 leads.

Integration:

- 1) Test power is not provided to transceiver when enables is turned off.
- 2) Test that power is provided to transceiver when enable is turned on.
- 3) Test that the transceiver can communicate with the endurosat OBC via USART.
- 4) Test that emission is 1W at 436.5 MHz.
- 5) Test that the bandwidth of the signal is 18Mhz as anticipated.
- 6) Test that antenna radiation pattern is isotropic as expected.
- 7) Test that transceiver can operate with groundstation receiver.
 - a. Test that the modulation is GMSK.
 - b. Test that the bitrate is 9.6Kbps.
 - c. Test that the protocol is AX.25.
- 8) Test that the monopole can deploy from endurosat OBC GPIO
 - a. Note that Custom board 2 design is not finished so specific pins cannot be given yet.

Test and Verification Plan

Requirement ID	Verification Strategy	Resources
COM-01	Test that AX.25 Protocol can be received from flatsat to groundstation hardware. Lower output power to reasonable amounts for this test.	Endurosat OBC Endurosat type II transceiver GOMSpace Antenna Coax Cable Assembly Groundstation hardware
COM-02	The Flatsat will send 5 images from onboard memory to groundstation hardware. Decoding the images will verify the design.	Groundstation hardware Flatsat

COM-03	The cubesat shall perform all necessary operational modes and commands list above.	Groundstation hardware Flatsat
COM-04	The cubesat will be able to send mock or real telemetry to the groundstation hardware	Groundstation hardware Flatsat
COM-05	The cubesat shall perform all necessary operational modes and commands list above.	Groundstation hardware Flatsat
COM-06	The flatsat must emit EM signal at 1W, at center frequency 436.5MHz(subject to change) under 24KHz bandwidth. Must be able to shut down power given command from groundstation hardware	Groundstation hardware Flatsat Field / anechoic chamber Spectrum analyser
COM-07	Antenna release structure must be set off at least after 30 minutes. Test each individual monopole itself.	Multimeter Flatsat custom board 2 Nylon burn wire wire Endurosat OBC
COM-08	Test that packetized data is in the form of AX.25	Flatsat Groundstation hardware (alternatively just an SDR)
COM-09	The CubeSat shall turn off transmitter upon request from Groundstation	Flatsat Groundstation hardware
COM-10	The mission design must comply with CCSDS telecommunication standards as outlined in [AD##] (TBD)	NA yet
COM-11	By notification of external sources	
COM-12	The cubesat shall turn off power enable pin when operational mode stop all communication is received from ground station	Flatsat Groundstation
COM-13	A communications link with the satellite must be established when it is at least ## degrees (TBD) above the horizon from the ground segment	NA yet
COM-14	Check that the groundstation access times are as expected. If higher access times then the elevation angle is lower than expected.	Cubesat Groundstation hardware
COM-15	Command data must be uplinked to the spacecraft in S-band Earth Exploration Satellite Service	
COM-16	Must test CRC checksums to make sure that error detection working	Flatsat Groundstation hardware
COM-17	Send a non-operating command to flatsat	Flatsat Groundstation hardware
COM-18	Test that the flatsat can schedule tasks from groundstation to cover 3 days.	Flatsat Groundstation hardware
COM-19	Science and telemetry data must be downlinked by the spacecraft in S-band Earth Exploration Satellite Service	
COM-20	Test that power emitted from flatsat is expected (1W at 436.5MHz).	Flatsat Groundstation
COM-21	The telecommunication system must be capable of simultaneously handling telemetry and science data downlinking and uplinked commands i.e. full duplex (TBC)	Not designed for this.

COM-22	The telecommunication equipment must support the linkage needs of the demo payload as specified in [AD##]. (TBD)	NA Yet
COM-23	Flatsat Test will satisfy this criteria.	Flatsat
COM-24	Flatsat will not draw more than 6W of power at any time when transmitting. Test voltage and current via multimeter during transmission.	Multimeter Transceiver
COM-25	Test antenna, transceiver and nylon wires on a scale to make sure they are less than 500g together	Antenna Nylon wire Transceiver Scale
COM-26	Test that the enable pin works on the transceiver.	Multimeter
COM-27	The communications subsystem must produce RF power output of no greater than 31.8 dBm (TBC) at the transmitting antenna	NA yet
COM-28	The communications subsystem must provide a reliable link to the ground station of at least 3 dB margin under nominal contact geometry of a ## degree (TBD) elevation angle.	NA Yet
COM-29	The communications subsystem downlink frequency range must be 2200 – 2290 MHz (TBC) according to the Canadian Frequency Allocation Table	Not applicable
COM-30	The communications subsystem uplink frequency range must be 2025 – 2110 MHz (TBC) according to the Canadian Frequency Allocation Table	Not Applicable
COM-31	Test that the cubesat emits a GMSK encoded signal	SDR Transceiver and antenna
COM-32	The communications subsystem downlink transmission rate must be at least 250 kbps (TBC)	Not applicable
COM-33	The communications subsystem uplink transmission rate must be at least 120 kbps (TBC)	Not Applicable
COM-34	The XYZ communications subsystem must use Two Line Element values for orbit propagation in order to define RF link periods	
COM-35	The XYZ satellite must provide command channel (uplink) authentication to ensure that only authorized mission control centers have access to the space segment	Not yet Thought of
COM-36	The communications system must utilize ## (TBD) –hand polarization for uplink and ## (TBD) –hand polarization for downlink	Left hand is emitted from the cubesat, Groundsatation is designed for both with polarization switch
COM-37	Characterisation of the COTS 10MHz bandpass filter.	Spectrum analyzer Bandpass filter
COM-38	All intentional receivers must be designed to withstand 160 dBμV/m at any frequency below 40 GHz at the receiving antenna (TBC), without permanent degradation to performance, reliability or life	Not yet Thought of

COM-39	All intentional receivers must be designed to operate without degradation in the presence of ground, sea, and space-based radars as defined in [RD3]	Not yet Thought of
--------	--	--------------------

Schedule and Work Plan for Phase C2 and D

Task Description	Estimation of Time and Human Resources	Required Resources to Complete
S band receiver Groundstation	1.5 Semester, Student team + Nick Mitchell	TBD, Likely close to CSA groundstation design
Groundstation assembly	70 hours, Nick Mitchell, Milpreet, Matt, Jayshri	All COTS components listed below.
Flatsat	50 hours Nick Mitchell, Alexis, Stephen	
SDR Coding	40 hours Student team + Nick Mitchell	USRP SDR
Antenna burnwire testing on custom board 2	20 hours, Nick Mitchell + boards team	Nylon wire, resistors, PCB

Datasheets for COTS Components

Datasheet for MCX and MMCX connectors:

<https://literature.hubersuhner.com/Technologies/Radiofrequency/RFConnectorsEN/?page=118>

Datasheet for Coax Cable:

Not yet chosen.

Datasheet for Transceiver + Antenna:

<https://github.com/cubesat-project/CubeSat/tree/master/Space%20Segment/COMM/COMM%20--%20NickSummer2019/CubeSat>

Datasheet for all groundstation components:

<https://github.com/cubesat-project/CubeSat/tree/master/Ground%20Segment/COMMS%20--%20GS%202019/Link%20Budget/Equipment>