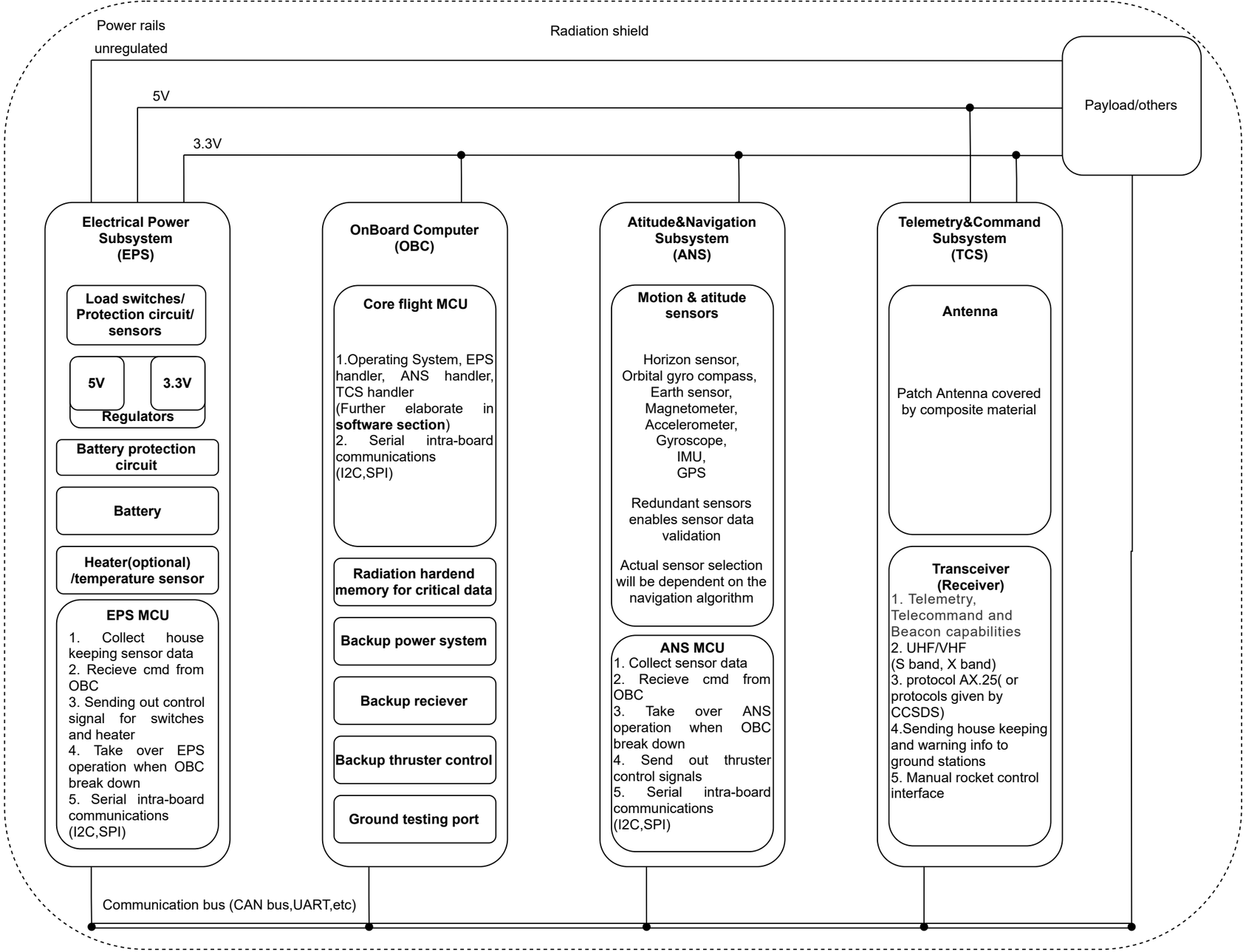

Onboard Software

1)Operating System
(INTEGRITY, FreeRTOS, VxWorks, CMSIS RTX, homemade RTOS)
2)EPS handler
(electrical status monitoring/reporting, switch power modes, operating heater)
3)ANS handler
(Runs a control algorithm such as a kalman filter to estmate current state as well as controlling the thrusters, calculate a local copy of sensor data for validation)
4)TCS handler
(Recieve cmd from ground station, send periodic status report (data from all sensors and ANS algorithm status) to ground station)

Design justifications

1) Electrical power: 3 power rails (3.3V, 5V, unregulated)on the top side of the diagram. Power drawn are limited to the current limite of the regulator and power operation mode. Unregulated power rails are used to meet requirements of payloads of non-standard operation voltage. Battery and load protection circuit prevent core modules gets shocked.

2) Inter-board communications: CAN bus was chosen in this case for its popularity in automobile application. It is noise tolerant, supports multiple device and it has built in CRC in the protocol.
In practice, interboard communication will be dependent on the interface avalible on the actual device.
Boards' internal communication protocol will be dependant on the sensors and MCUs themselves.

3) Data corruption protection: Radiation hardened electronics, metallic casing for radiation shielding, differential data transmission lines, error checking bits in communication packets. (Radiation hardened chips are rather expensive)

4) **Back up subsystem:** If OBC shuts down, EPS,ANS and TCS processor will take over the work load at the price of processing speed. If EPS, ANS or TCS shut down, backup unit on OBC will take over. The backup systems hardware are configurable depending on the emergency strategy.

5) **Sensor data validation:** real-time sensor data calculation from redundant sensor unites as a reference to sensor readings. Reduntant sensors provide multiple ways to solve the curent state of the rocket, which we can cross reference each other to ensure the result make sense.

Power budget: (Assuming 3.3V unless specified)
EPS: 25mA (without the heater)
OBC:200mA (Raspberry pi zero was used as a reference)
ANS: assume 6 sensors with 5mA each + MCU 5mA = 55mA
TCS: TX 606mA(on 5V)20%of flight time, RX 25mA80%of flight time
Heater: 5W
Total: 6.64W, 1.64W without heater

To operate for 5h we need a battery of capacity 33.2Wh estimated weight 200g
or 8.2Wh without heater

Mass budget:
Battery: 15g-200g
EPS: 290g
OBC: 200g
ANS: 200g
TCS: 250g (assuming one antenna board)
radiatiion shield mass included in each subsystem
total weight 955-1140g

Detailed design will vary if we use parts from the market

Links below are the CubeSat part that can be used for our project
(Price range in a few thousand euros)
TCS:
https://nanoavionics.com/cubesat-components/s-band-transceiver/
https://www.isispace.nl/product/isis-uhf-downlink-vhf-uplink-full-duplex-transceiver/
https://www.endurosat.com/cubesat-store/cubesat-communication-modules/uhf-transceiver-ii/
https://gomspace.com/shop/subsystems/communication-systems/nanocom-sr2000.aspx
OBC:
https://nanoavionics.com/cubesat-components/cubesat-on-board-computer-main-bus-unit-satbus-3c2/
https://www.isispace.nl/product/on-board-computer/
https://www.endurosat.com/cubesat-store/cubesat-obc/onboard-computer-obc/
https://gomspace.com/shop/subsystems/command-and-data-handling/nanomind-z7000.aspx
EPS：
https://gomspace.com/shop/subsystems/power/nanopower-bpx.aspx
https://www.endurosat.com/cubesat-store/cubesat-power-modules/eps-power-module-ii/
https://www.isispace.nl/product/ieps-electrical-power-system/
https://nanoavionics.com/cubesat-components/cubesat-electrical-power-system-eps/
ANS system for rockets are very different from the ones in Cube satillites, which needs a specific design
Antenna needs a specific design as well to fit in the rocket geometry.