Main Board:

Power Tests:

1. TPS54226PWPR (3.3V Voltage Regulator)

- ☑ ~~Apply 7.4V (Vsys) to battery terminals using a DC power supply~~
- ☑ ~~Measure output voltage on TPS54226 using a multi~~
- ☑ ~~Result: Output voltage should be 3.3V +/- 0.05V~~

2. TPS7A4501DCQT (5V Voltage Regulator)

- ☑ ~~Apply 7.4V (Vsys) to battery terminals using a DC power supply~~
- ☑ ~~Measure output voltage on TPS7A4501 using a multimeter in voltage mode across the capacitor (C56)~~
- ☑ ~~Result: Output voltage should be 5V +/- 0.1V~~

3. BQ25883RGER (USB Charger)

- ☑ ~~Disconnect battery terminal (J10) and connect USB cable from PC to mainboard~~
- ☑ ~~Measure voltage across capacitor C29/C37 (VBUS), to be about 5V~~
- ☑ ~~When USB is plugged, it can charge the battery and power the system (3.3V and 5V lines are active)~~

MCU Tests:

1. RP2040 (Main MCU)

- ☑ ~~Connect USB cable from PC to mainboard~~
- ☑ ~~RP2040 appears as USB drive on PC~~

2. MR25H40MDF (Flash storage)

- ☑ ~~Install Circuitpython firmware on MCU~~
- ☑ ~~Check size of the MCU USB drive~~
- ☑ ~~Size of drive is 4MB~~

3. MAX706RESA+ (Circuit Watch Dog)

Test 1:

- ☑ ~~Write Circuitpython test script to enable Watch Dog Timer (WDT) by setting the WDT_EN pin high~~
- ☑ ~~Mainboard resets within 1.6 seconds~~

Test 2:

- ☑ ~~Write Circuitpython test script to toggle the WDT_WDI pin between high and low (pulse) every 0.5 seconds~~
- ☑ ~~Set WDT_EN pin high~~
- ☑ ~~Mainboard should not reset~~

Device Tests:

1. SD Card

- ☑ ~~Measure voltage across capacitor C32/C33, to be 3.3V +/- 0.05V~~
- ☑ ~~Write circuitpython script to establish SPI connection. Script should not return errors.~~
- ☑ ~~RP2040 can successfully read and write data to the card~~

2. BNO085 (IMU)

- ☑ ~~Measure voltage across capacitor (C18/C19), to be 3.3V +/- 0.05V~~
- ☑ ~~Write circuitpython script to establish I2C connection. Script should not return errors.~~
- ☑ ~~Accelerometer can respond to linear acceleration changes, and does not return null or invalid values~~
- ☑ ~~Gyroscope can respond to the angular velocity changes and does not return null or invalid values~~
- ☑ ~~Magnetometer can respond to magnetic fields changes and does not return null or invalid values~~

3. DS3231S (RTC)

- [x] ~~Measure voltage across capacitor (C34), to be 3.3V +/- 0.05V~~
- [x] ~~Write circuitpython script to establish I2C connection. Script should not return errors.~~
- [x] ~~Write circuitpython script to set RTC to correct time.~~
- [x] ~~Remove power to the mainboard~~
- [x] ~~Let RTC remain powered through button cell only for 15 minutes~~
- [x] ~~Write circuitpython script to check RTC time~~
- [x] ~~RTC time should be within 1 second of real time~~

Communications Tests:

1. E22-900M30S (LoRa)

- ☑ ~~Write circuitpython script to establish SPI connection. Script should not return errors.~~
- ☑ ~~Set TX_EN pin high, observe power consumption on DC power supply.~~
- ☐ Power consumption should be under 1W
- ☑ ~~Set RX_EN pin high, observe power consumption on DC power supply~~
- ☐ Power consumption should be under 0.25W

2. S1216V8 (GNSS Receiver)

- ☑ ~~Measure voltage across capacitor (C51?), to be 3.3V +/- 0.05V~~
- ☑ ~~Established UART communication~~

~~External Connector~~

~~Camera Board~~

- [ ] ~~Proper power connection~~
- [ ] ~~Successful RX/TX transmission~~
- [ ] ~~Successful I2C communication~~
- [ ]

~~Battery Board~~

- [ ] ~~Proper power connection~~
- [ ] ~~Successful I2C communication~~
- [ ] ~~Successful battery alert signal~~
- [ ]

~~Jetson Board~~

- [ ] ~~Proper power connection~~
- [ ] ~~Successful I2C communication~~
- [ ] ~~Successful SPI communication~~
- [ ] ~~Proper slave select signal from RP2040~~
- [ ]

~~XYZ Boards~~

- [ ] ~~Proper power connection~~
- [ ] ~~Successful I2C communication~~

Battery Board:

Pre-test procedure:

- **Use Power supply as a power source instead of Battery cells**
- **Apply 2 Power supplies in series (suggest):**
  First one 3.7V->left bottom pad-> + (positive)
  ->Right up pad-> - (negative)
  Second one 3.7V->Right up pad-> + (positive)
  ->Left up pad-> - (negative)
  Or
  1 Power supply:
  7.4V->left bottom pad-> + (positive)
  ->Left up pad-> - (negative)

Power Tests:

1. Battery Board 3.3V (from main board)

- ☐ Connect the Mainboard from the port J10
- ☐ Measure output voltage using a multimeter in voltage mode across the capacitor (C16)
- ☐ Result: Output voltage should be 3.3V +/- 0.1V

2. Output Power (J7)

- ☐ Measure output voltage using a multimeter in voltage mode (between SYSTEM_VDD and SYSTEM_VSS)
- ☐ Result: Output voltage should be 7.4V +/- 0.1V

Battery Monitor

MAX17205G+

- ☐ The voltage across C6 shall be 3.7 to 4.2V; Voltage across C5 shall be 3.7 to 4.2V.
- ☐ MCU can detect its I2C address
- ☐ Proper voltage readings and other information like the State of Charge sent through I2C
- ☐ Send Alert signal once bad things happen

Battery Bus Protection

- ☐ If the CHARGE_EN and DISCHARGE_EN are low, the SYSTEM_VDD should be approximately equal to the battery voltage (BATT2).
- ☐ CHARGE_ACTIVE should only be high if charging, and DISCHARGE_ACTIVE should only be high if the battery discharges.
- ☐ When discharging, the U8 Gate-Ground should be High, the U9 Gate-Ground should be

low. When charging, the U8 Gate-Ground should be low, and the U9 Gate-Ground should be high.

Battery Temperature Protection

BTS5090-1EJA

- ☐ C18 measures 7.4V.
- ☐ It shall activate and deactivate based on the signal from the IO Expander
- ☐ It shall cut off the current if the temperature exceeds a certain threshold.

Resistor Array

- ☐ Generate enough heat to warm up the batteries
- ☐ The circuit can withstand the heat without damaging itself

~~I/O Expander~~

~~PCA9536D~~

- ☐ ~~C16 is Properly powered by 3.3V~~
- ☐ ~~Establish I2C connection and send out proper signals~~

- [ ] ~~While discharging, successfully receive the CHARGE_ACTIVE and DISCHARGE_ACTIVE signals from IO0 and IO1~~
- [ ] ~~While discharging, successfully send the HEATER_ENABLE signals from IO2 and IO3~~

Pre-test procedure:

- **Remove the external power source**
- **Use two 18650 batteries and load them on the battery holder**

Battery Test:

- [ ] Measure output voltage using a multimeter in voltage mode
  - 0V at BATT0
  - 3.7V to 4.2V at BATT1
  - 3.7V to 4.2V at BATT2
- [ ] No reverse charging between batteries (Each Battery should have approximately the same voltage)

Output Power (J7):

- ☐ Measure output voltage using a multimeter in voltage mode (between SYSTEM_VDD and SYSTEM_VSS)
- ☐ Result: Output voltage should be 7.4V +/- 0.1V

Pre-test procedure:
2 power sources (suggested):
One on the pad: left down + / left up -
One on J7: SYSTEM_VDD + / SYSTEM_VSS -

Battery Bus Detection:

1. S-8209AAA-T8T1U (Discharge/Over Power Protection)

- ☐ Change the Pad power source to 8.4V
- ☐ Measure DISCHARGE_EN voltage using a multimeter in voltage mode, It should be HIGH
- ☐ Measure the voltage between two sides of U4 to see if it is the same

2. S-8209AAA-T8T1U (Changing/Under Power Protection)

- ☐ Change the Pad power source to 4.8V
- ☐ Measure CHARGE_EN voltage using a multimeter in voltage mode, It should be HIGH

- [ ] Measure the voltage between two sides of U6 to see if it is the same

3. **LTC4412xS6 (No need To test)**

~~Interfaces~~

~~J10~~

- [ ] ~~Send out the correct alert signal~~
- [ ] ~~Send proper I2C data~~
- [ ] ~~Send 3.3V power~~

~~J7~~

- [ ] ~~Send out the correct VDD power~~

~~J8~~

- [ ] ~~Send out the correct alert signal~~
- [ ] ~~Send proper I2C data~~
- [ ] ~~Send 3.3V power~~

X/Y Board:

Pre-test procedure:

- ☐ Connect with Deployables with J1, and connect with Mainboard with J2.
- ☐ The mainboard should also connect to the Batteryboard and follow the pre-test procedure of the Batteryboard.
- ☐ Deployables should face the sun or artificial sunlight for power generation.

Power Tests:

1. DRV8235RTER (Coil Driver)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure output voltage on DRV8235RTER using a multimeter in voltage mode across the capacitor (C26/C27)
- ☐ Result: Input voltage should be 7.4V +/- 0.1V

2. OPT4003DNPRQ1 (Light Sensor)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure input voltage on OPT4003DNPRQ1 using a multimeter in voltage mode across the capacitor (C36).
- ☐ Result: Input voltage should be 3.3V +/- 0.05V

3. LTC3130 (MPPT)

- ☐ Measure the voltage generated (Vin) from the deployable using a multimeter in voltage mode across the capacitor (C13-C16).
- ☐ Input voltage between Vin and Ground should be at least higher than 10V as a result of the MPPC.
- ☐ Ideally, the input voltage between Vin and Ground should be about 16V as a result of the MPPC.
- ☐ Output voltage between Vout and Ground should be higher than 7.4V as a result of FB.

4. LTC4412xS6 (Or-ing)

- ☐ Measure the voltage using a multimeter in voltage mode across the capacitor (C1).

- [ ] Result: Input voltage should be ?V +/- ?V

5. ADM1176-1ARMZ-R7 (Power Monitor)

- [ ] Measure the voltage using a multimeter in voltage mode between R124 and the Ground.
- [ ] Result: Input voltage should be 7.4V +/- 0.1V

Device Tests:

1. DRV8235RTER (Coil Driver)

- [ ] (If needed) Switch PERI_3V3 to the ground, the driver should be in low-power sleep mode.
- [ ] IPROPI should increase and decrease according to COIL_P and COIL_N.
- [ ] Write circuitpython script to establish I2C connection. Script should not return errors. (Detect address)

2. OPT4003DNPRQ1 (Light Sensor)

- [ ] Write circuitpython script to establish I2C connection. Script should not return errors.
- [ ] Complete/Partial/Not block the sensor, should see significant increases in lux value.

3. LTC3130 (MPPT)

- ☐ Write circuitpython script to establish I2C connection. Script should not return errors.
- ☐ Input voltage between Vin and Ground should be at least higher than 10V as a result of the MPPC.
- ☐ Ideally, the input voltage between Vin and Ground should be about 16V as a result of the MPPC.
- ☐ Output voltage between Vout and Ground should be higher than 7.4V as a result of FB.

Solar Cell Tests:

1.

-Z Board:

Pre-test procedure:

- ☐ Connect with Mainboard with J1.
- ☐ The mainboard should also connect to the Batteryboard and follow the pre-test procedure of the Batteryboard.
- ☐ Tying the fishing line on the burn wires.

Power Tests:

1. DRV8235RTER (Coil Driver)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure output voltage on DRV8235RTER using a multimeter in voltage mode across the capacitor (C10/C9)
- ☐ Result: Input voltage should be 7.4V +/- 0.1V

2. OPT4003DNPRQ1 (Light Sensor)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.

- ☐ Measure input voltage on OPT4003DNPRQ1 using a multimeter in voltage mode across the capacitor (C5).
- ☐ Result: Input voltage should be 3.3V +/- 0.05V

3. PCA9633DP2 (Burn Wire Controller)

- ☐ Measure input voltage on PCA9633DP2 using a multimeter in voltage mode across the capacitor (C13)
- ☐ Result: Input voltage should be 3.3V +/- 0.05V

4. IM11DGR (Power Relay)

- ☐ Measure input voltage on IM11DGR using a multimeter in voltage mode crossing C16.
- ☐ Result: Input voltage should be 3.3V +/- 0.05V
- ☐ Measure load on IM11DGR Pin 4 and Pin 5 using a multimeter in voltage mode.
- ☐ Result: The outcome should be 7.4V +/- 0.1V

Device Tests:

1. DRV8235RTER (Coil Driver)

- ☐ (If needed) Switch PERI_3V3 to the ground, the driver should be in low-power sleep mode.

- ☐ IPROPI should increase and decrease according to COIL_P and COIL_N.
- ☐ Write circuitpython script to establish I2C connection. Script should not return errors. (Detect address)

2. OPT4003DNPRQ1 (Light Sensor)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure input voltage on OPT4003DNPRQ1 using a multimeter in voltage mode across the capacitor (C5).
- ☐ Result: Input voltage should be 3.3V +/- 0.05V

3. PCA9633DP2 (Burn Wire Controller)

- ☐ Write circuitpython script to establish I2C connection. Script should not return errors. (Detect address)
- ☐ By soldering or desoldering R11, R12, R13, and R37, the detected address should change according to the datasheet.
- ☐ Write circuitpython script to manipulate the output of LED0 to LED3 pins.

- [ ] The signal output from LED0 to LED3 should correspond to circuitpython code.

4. IM11DGR (Power Relay)

- [ ] When the BURN_RELAY_A signal from LED3 pins of PCA9633DP2 is high, the switch shall move to pin4, driving VBURN_A_IN high.
- [ ]

5. Burn Wire

- [ ] While the VBURN_A_IN is high, if the BURN_EN is low, the corresponding burn wire should be heated.
- [ ] Once the burn wire increases temperature, the fishing line should melt.

~~**6. DXW21HN5011BL (Transformer)**~~

~~**7. CGGBP.18.4.A.02 (GPS Antenna)**~~

+Z Board:

Pre-test procedure:

- ☐ Mount all four sun sensor micro PCBs on each edge at 45 degrees.
- ☐ Connect with Mainboard with J1.
- ☐ The mainboard should also connect to the Batteryboard and follow the pre-test procedure of the Batteryboard.

Power Tests:

1. DRV8235RTER (Coil Driver)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure output voltage on DRV8235RTER using a multimeter in voltage mode across the capacitor (C11/C12)
- ☐ Result: Input voltage should be 7.4V +/- 0.1V

2. Micro PCBs (Sun Sensors)

- ☐ If the Solar Panel cannot support itself or sunlight isn't available, apply 7.4V (Vsys) to battery terminals using a DC power supply.
- ☐ Measure output voltage on Micro PCBs using a multimeter in voltage mode across the capacitor (C8/C5/C6/C7).

- [ ] Result: Input voltage should be 3.3V +/- 0.05V

3. LTC3130 (MPPT)

- [ ] Measure the voltage generated (Vin) from the deployable using a multimeter in voltage mode across the capacitor (C18-C21).
- [ ] Input voltage between Vin and Ground should be at least higher than 10V as a result of the MPPC.
- [ ] Ideally, the input voltage between Vin and Ground should be about 16V as a result of the MPPC.
- [ ] Output voltage between Vout and Ground should be higher than 7.4V as a result of FB.

4. LTC4412xS6 (Or-ing)

- [ ] Measure the voltage using a multimeter in voltage mode across the capacitor (C10).
- [ ] Result: Input voltage should be ?V +/- ?V

5. ADM1176-1ARMZ-R7 (Power Monitor)

- [ ] Measure the voltage using a multimeter in voltage mode between R55 and the Ground.
- [ ] Result: Input voltage should be 7.4V +/- 0.1V

Device Tests:

1. DRV8235RTER (Coil Driver)

- ☐ (If needed) Switch PERI_3V3 to the ground, the driver should be in low-power sleep mode.
- ☐ IPROPI should increase and decrease according to COIL_P and COIL_N.
- ☐ Write circuitpython script to establish I2C connection. Script should not return errors. (Detect address)

2. Micro PCBs (Sun Sensors)

- ☐ Write circuitpython script to establish I2C connection. Script should not return errors.
- ☐ Complete/Partial/Not block the sensor, should see significant increases in lux value.

3. LTC3130 (MPPT)

- ☐ Write circuitpython script to establish I2C connection. Script should not return errors.
- ☐ Input voltage between Vin and Ground should be at least higher than 10V as a result of the MPPC.

- ☐ Ideally, the input voltage between Vin and Ground should be about 16V as a result of the MPPC.
- ☐ Output voltage between Vout and Ground should be higher than 7.4V as a result of FB.

Solar Cell Tests:

1.