OSPRE
Critical Design Review

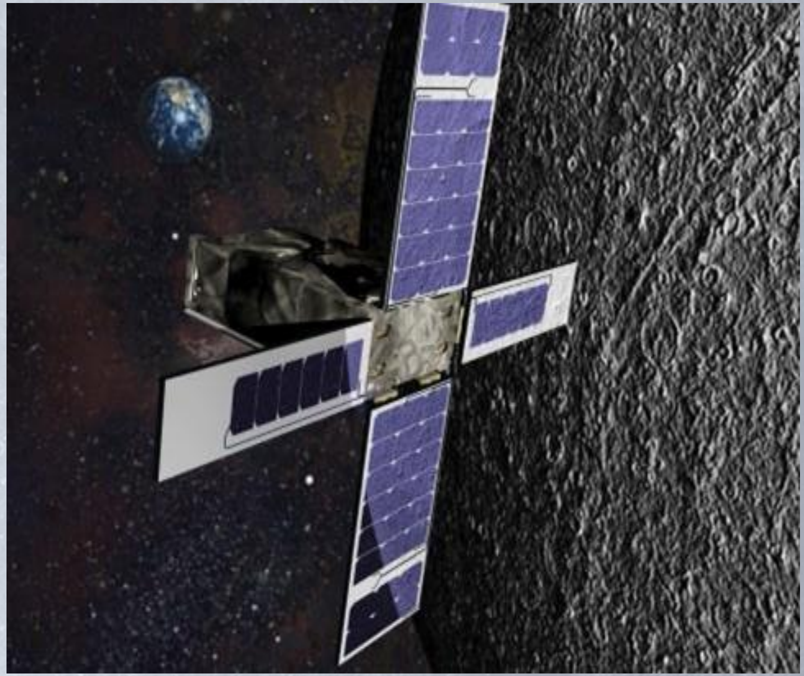
Project Purpose and Objectives

Mission Description

- To use optical relative navigation to determine a spacecraft's state vector and state vector error during a lunar transit

- CubeSat based on NASA CubeQuest challenge
  - Lunar Mission
  - Launch on SLS EM-1

- Customer: Lockheed Martin

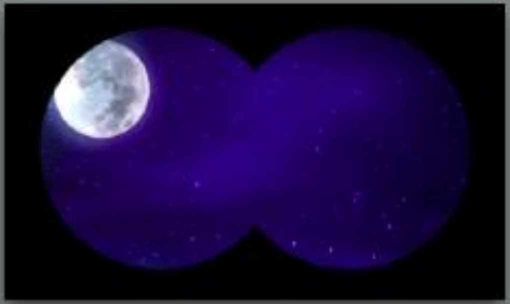


Lunar cubesat

Relative Navigation

1

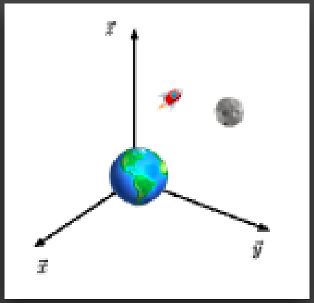
WHAT I SEE



3

WHAT I CAN FIND OUT

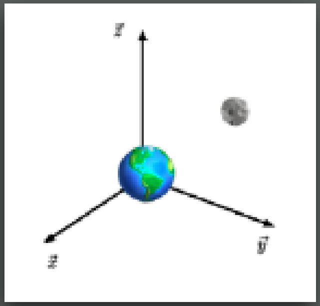
Spacecraft Location



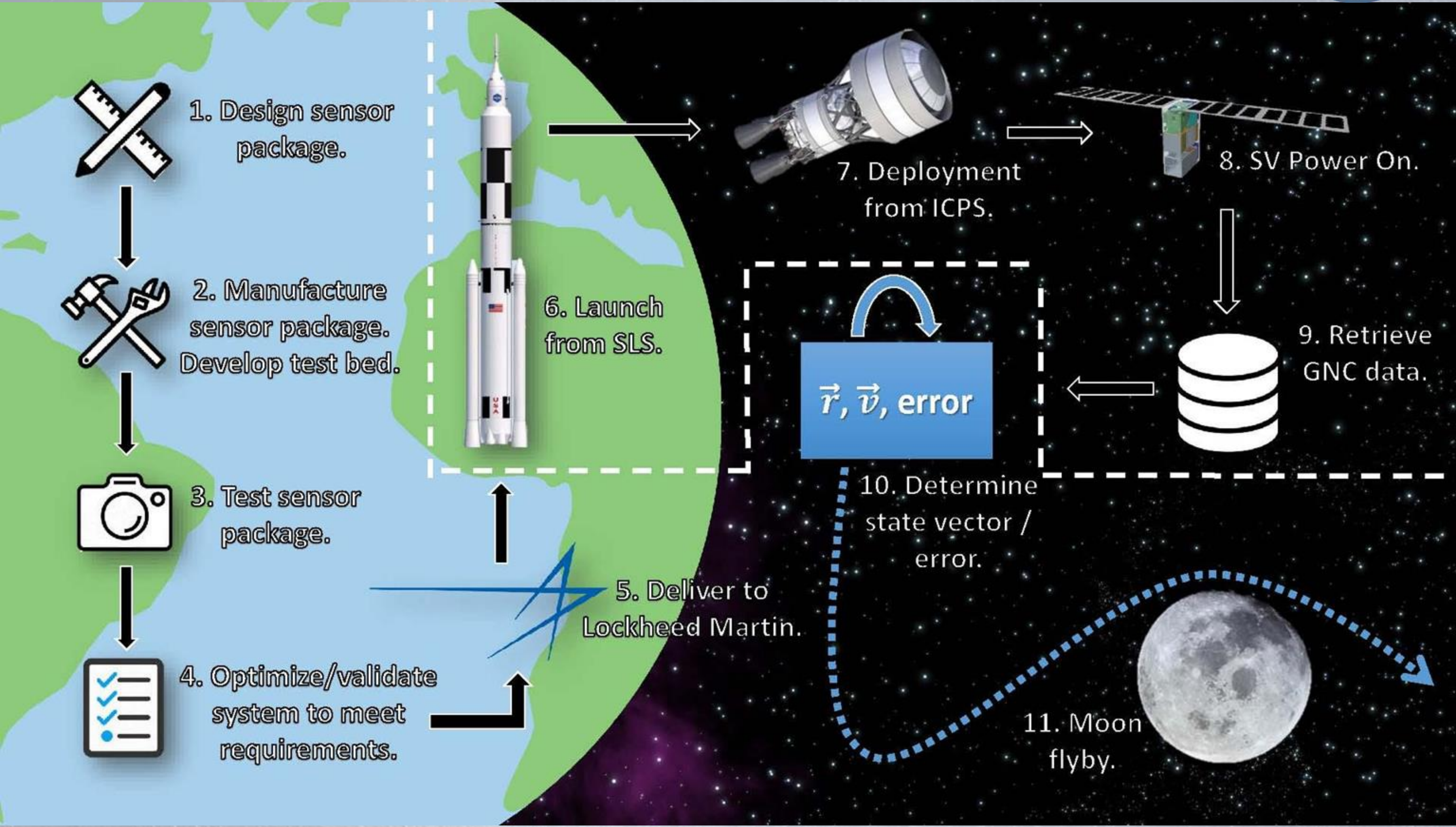
2

WHAT I KNOW

Location of the Earth & Moon



Mission CONOPS



1. Design sensor package.

2. Manufacture sensor package. Develop test bed.

3. Test sensor package.

4. Optimize/validate system to meet requirements.

5. Deliver to Lockheed Martin.

6. Launch from SLS.

7. Deployment from ICPS.

8. SV Power On.

9. Retrieve GNC data.

10. Determine state vector / error.

11. Moon flyby.

Functional Requirements

FR 0.0 -	Provide relative navigation from an image sensor package on a lunar trajectory.
FR 1.0 -	Provide state vector within desired error bounds: • Velocity: ± 250 m/s Position: ±1000 km
FR 2.0 -	Meet dimensional requirements: • 50 x 50 x 10 mm
FR 2.1 -	Meet electrical requirements: • 3.3, 5, or 12VDC, 0.5A (max), 3W (max)
FR 2.2 -	Meet interfacing requirements: • SPI or I^2C

Critical Project Elements

Critical Project Elements

Solution Accuracy

- State vector must be determined to within the required accuracy

Camera resolution	Finite camera resolution limits the information we can obtain from an image
Image Processing	Imperfect image processing introduces error
Navigation algorithms	Imperfect navigation algorithms introduce error

Critical Project Elements

Testing Accuracy

- Solution accuracy must be verified in testing
- Measures solution accuracy via scaling of the Earth-Moon system

Measurement of distance between camera and target	Imprecise measurements introduces error
Measurement of the location of the center of the target	

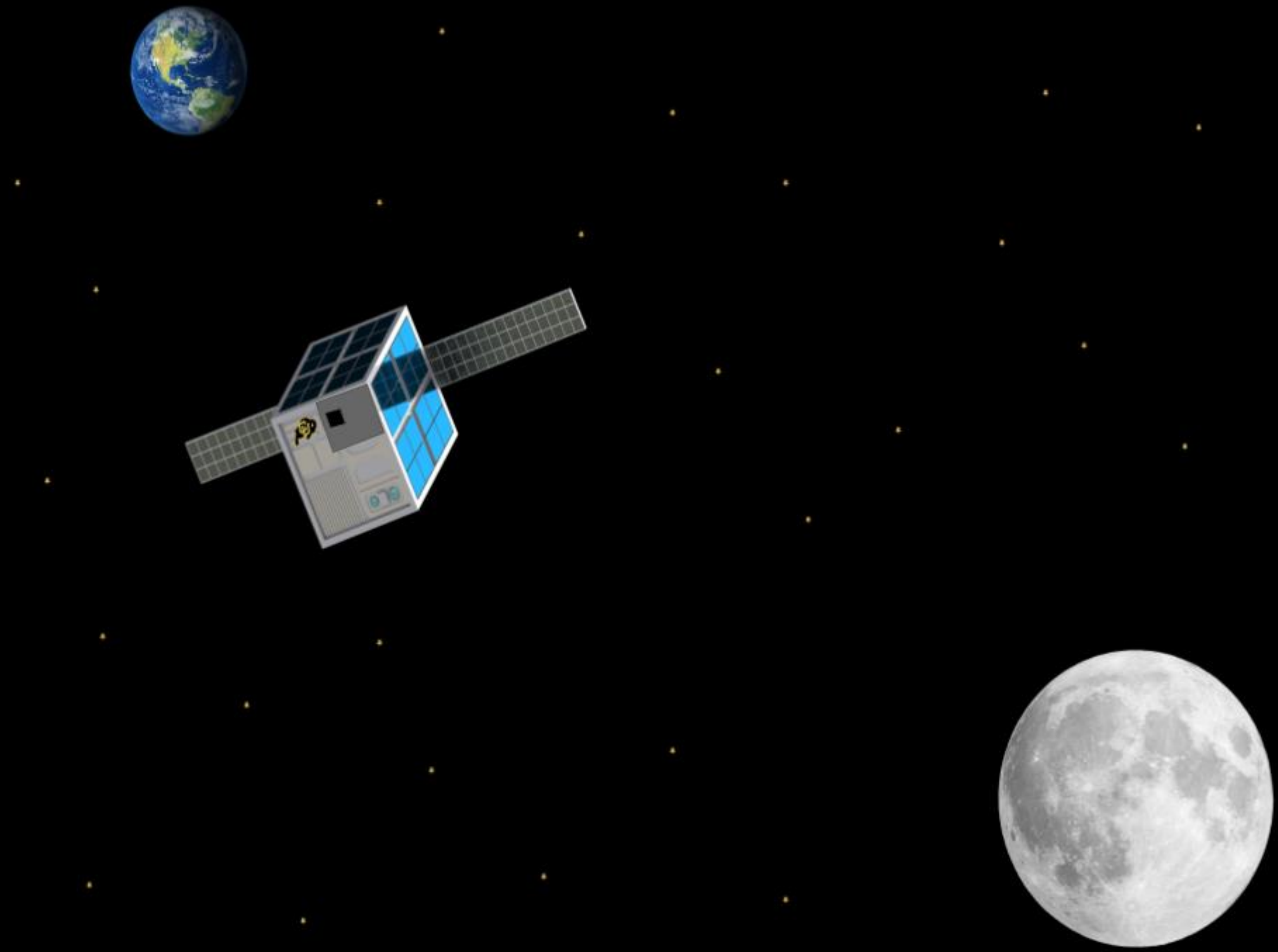
Critical Project Elements

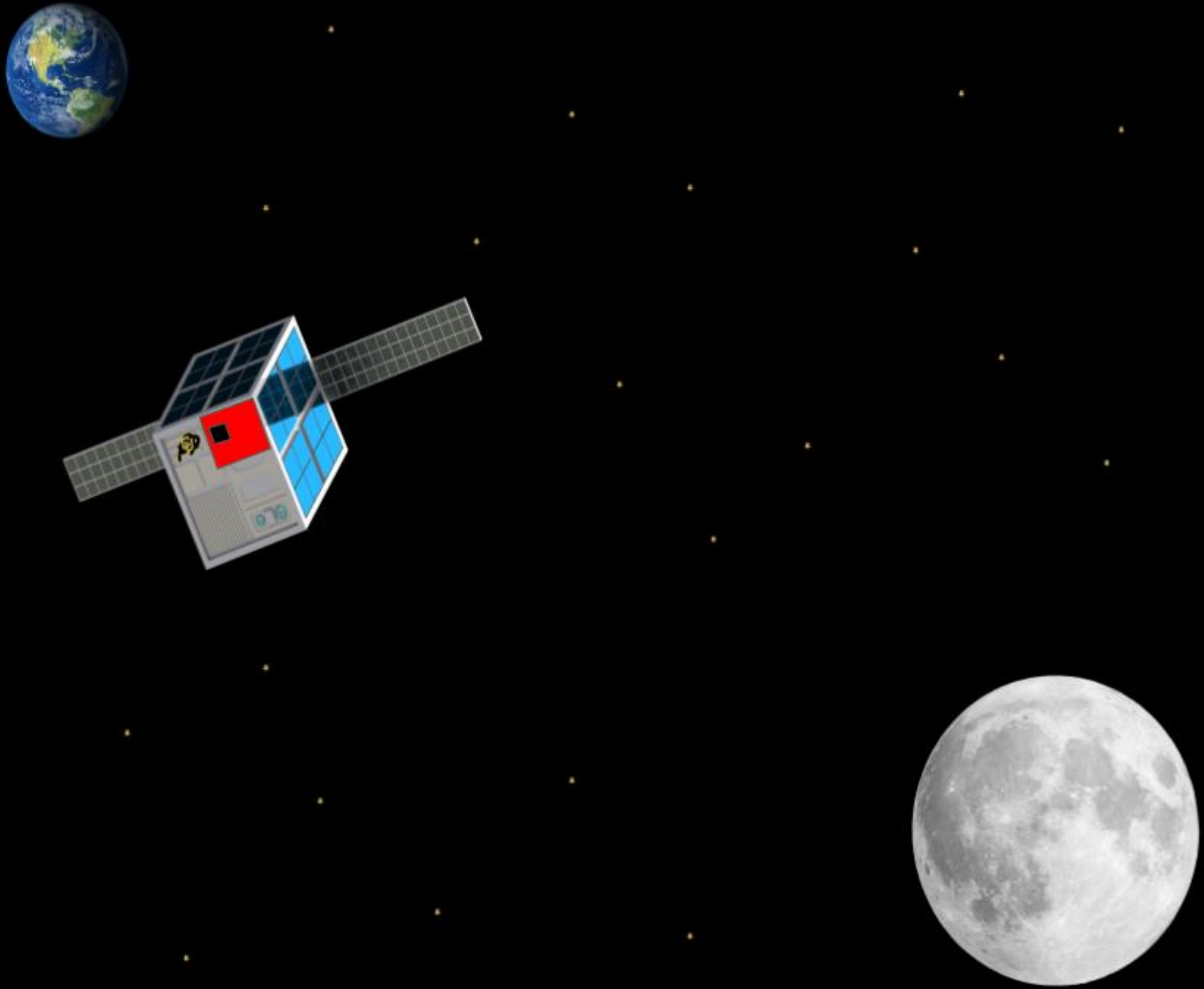
SWAP

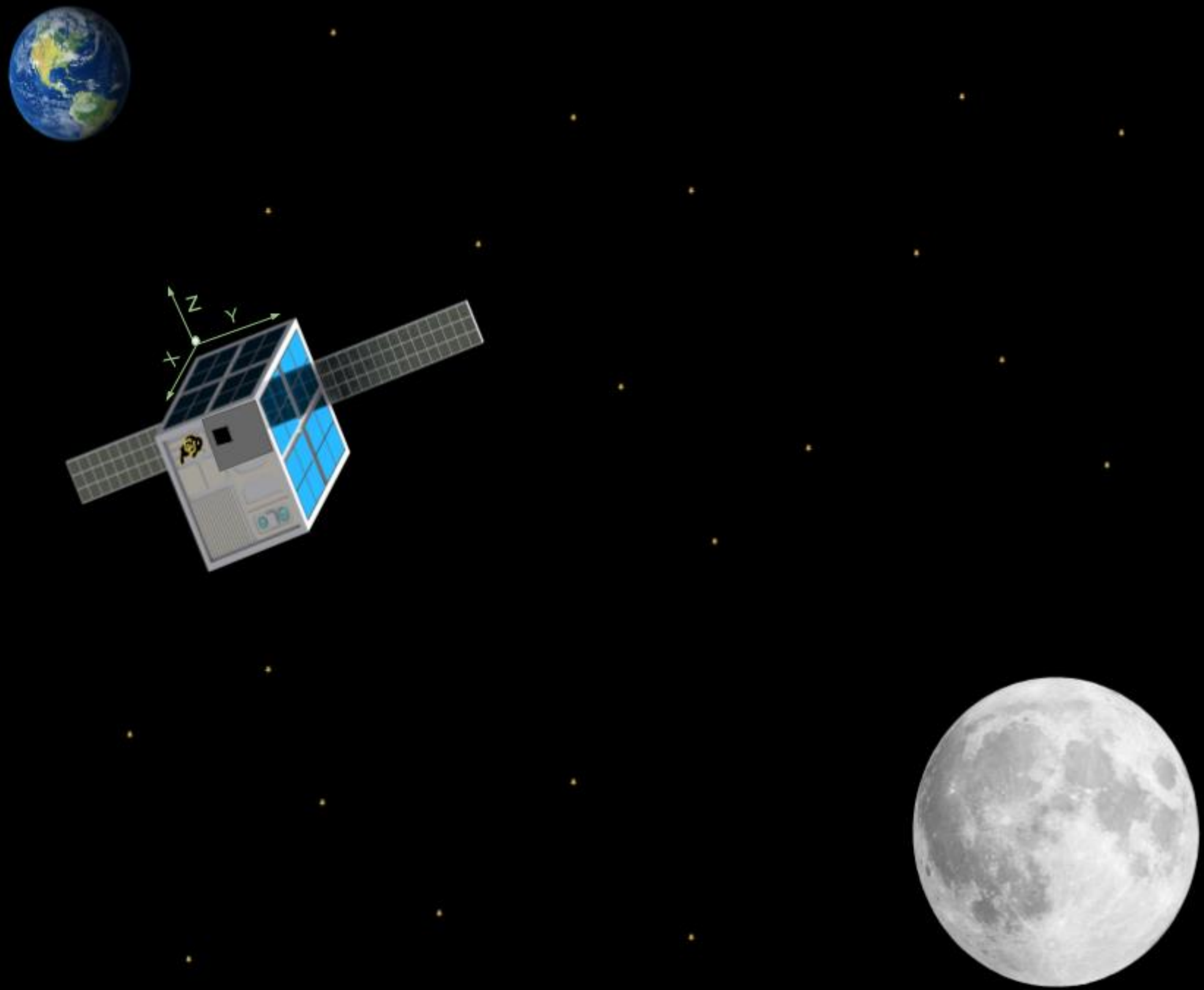
- Size, Weight, And Power requirements must be met

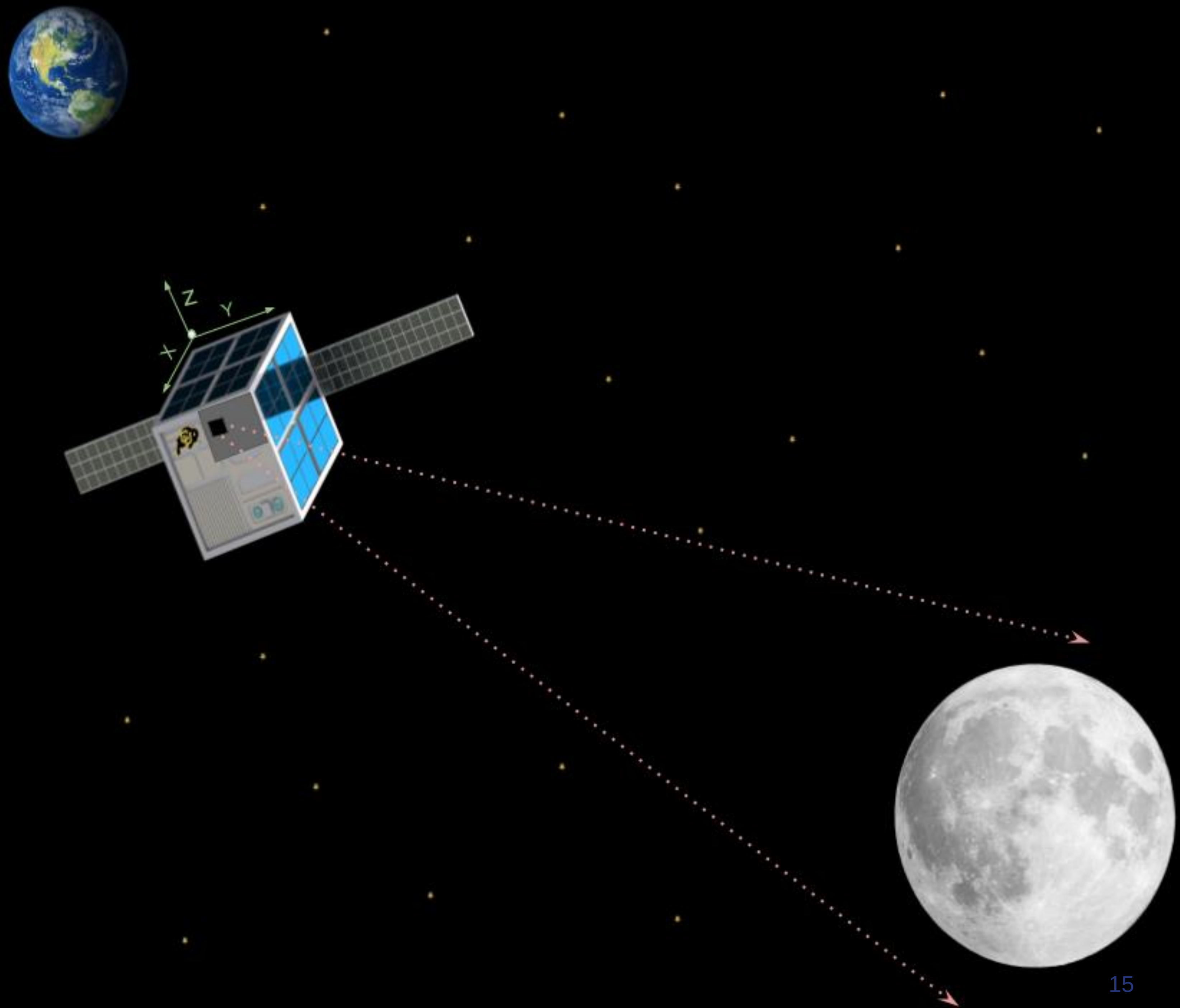
Component size	Components that are too large violate volume constraints - this driver has limited camera options
Component power draw	Components that draw too much power violate power constraints
Component weight	Components that are too heavy violate weight constraints

Design Solution

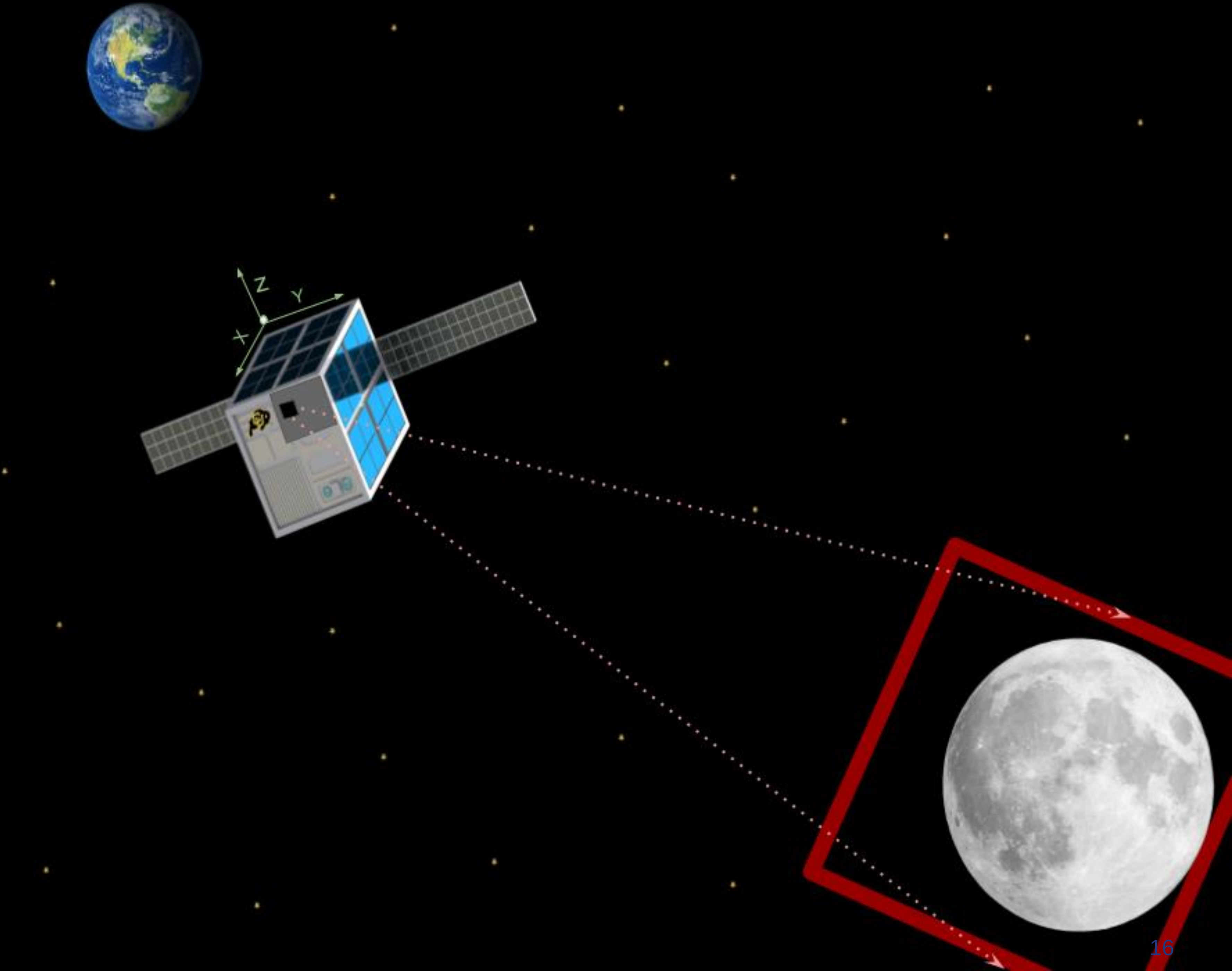


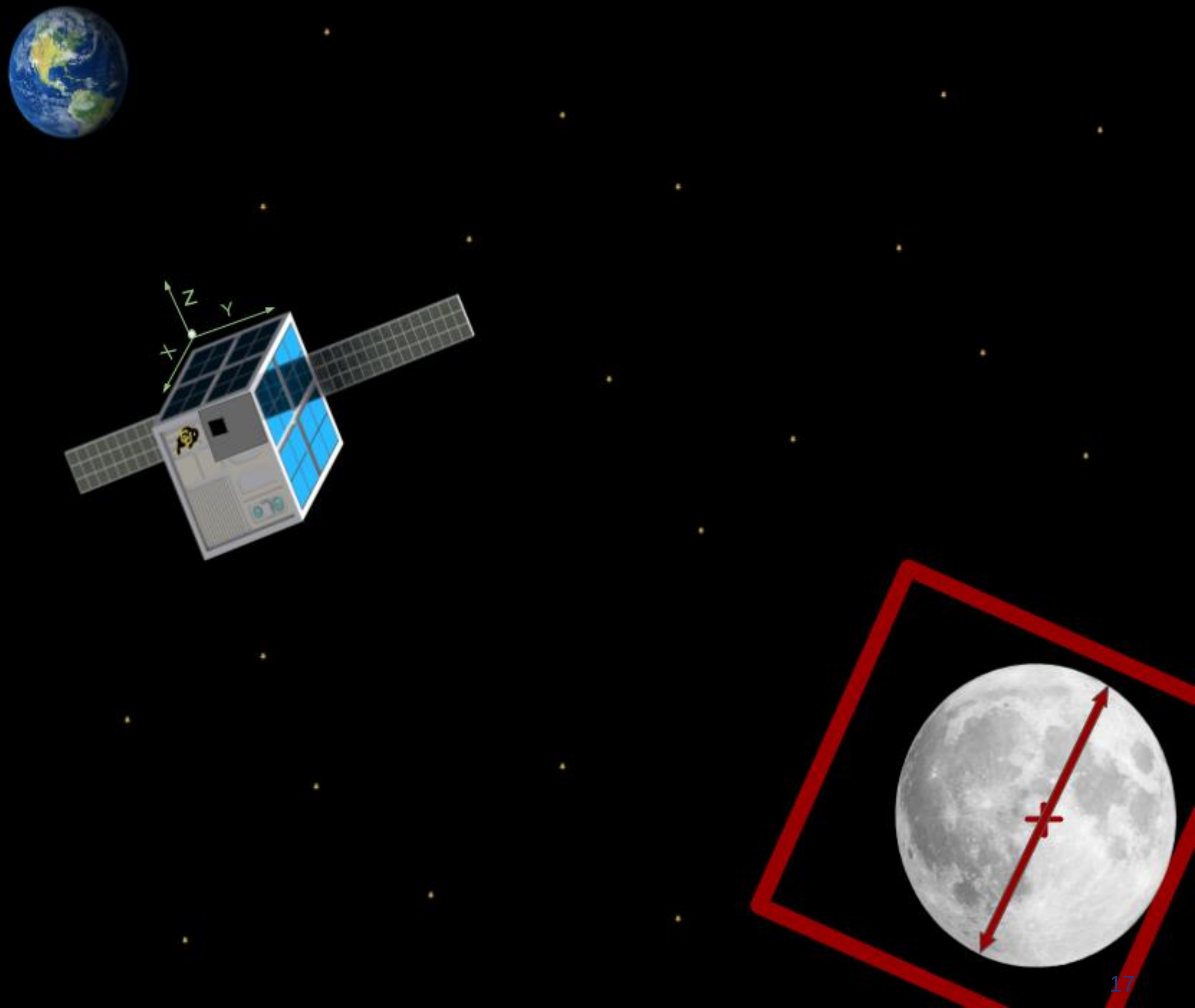


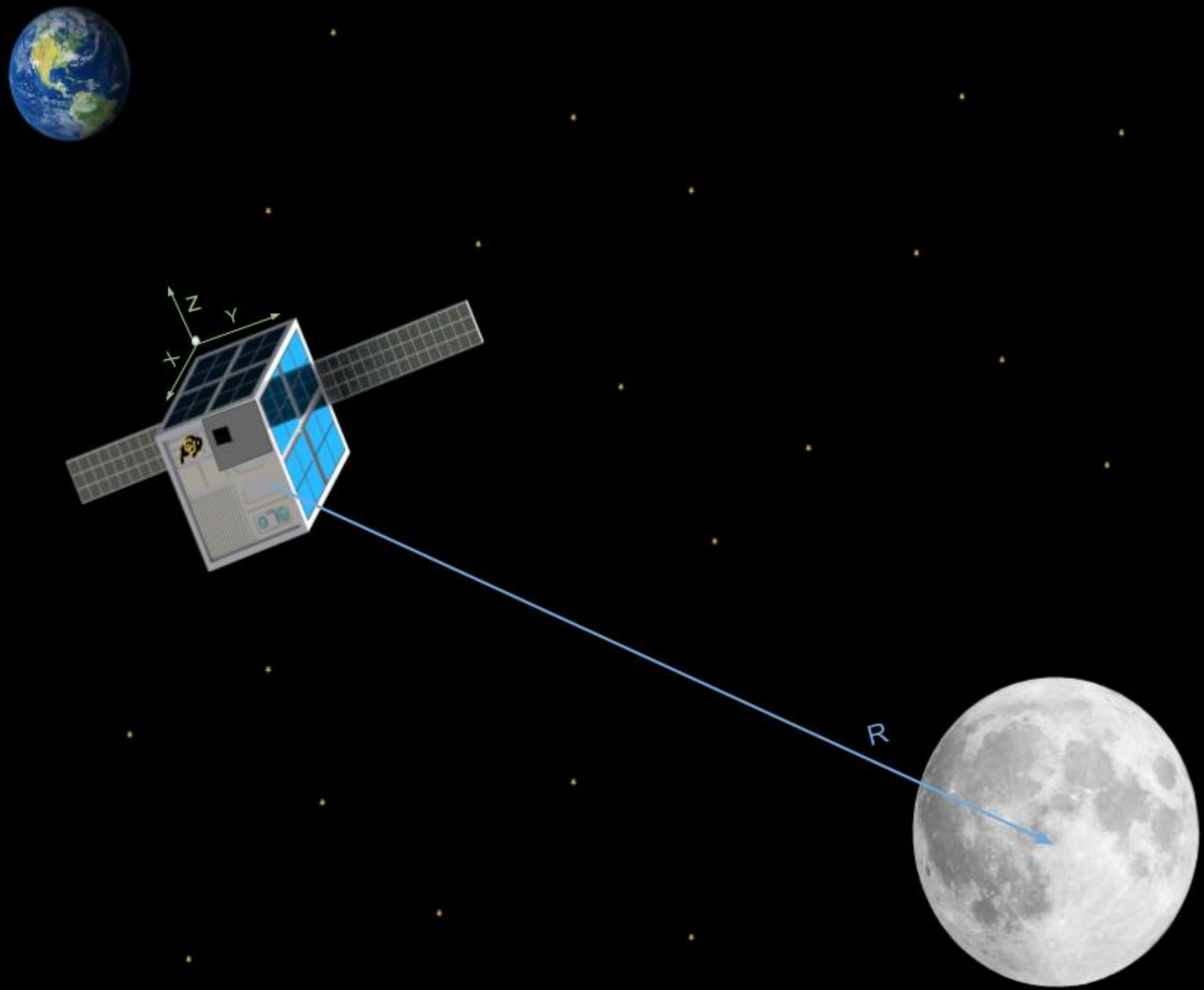

Z
Y
X

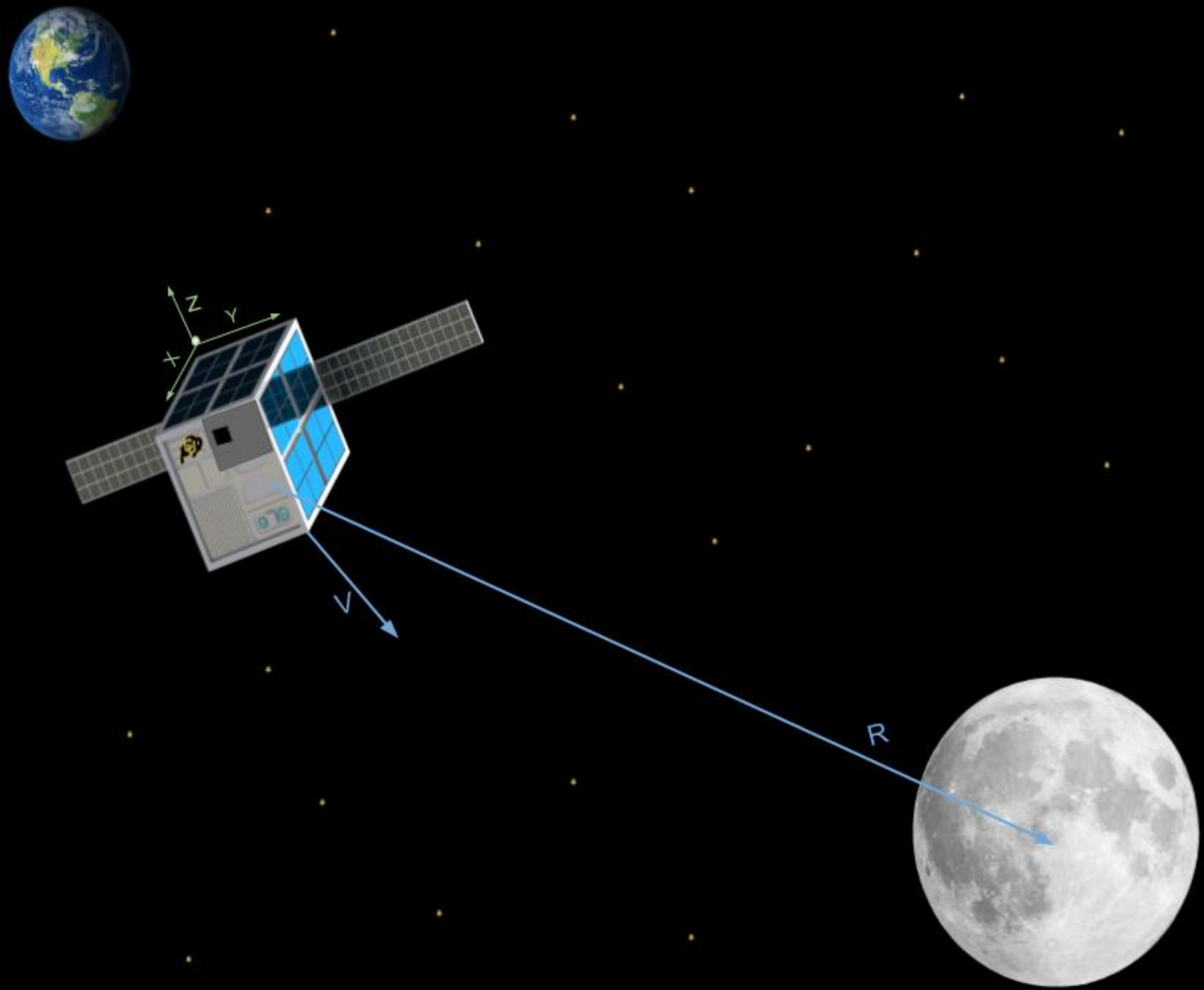


Z
Y
X

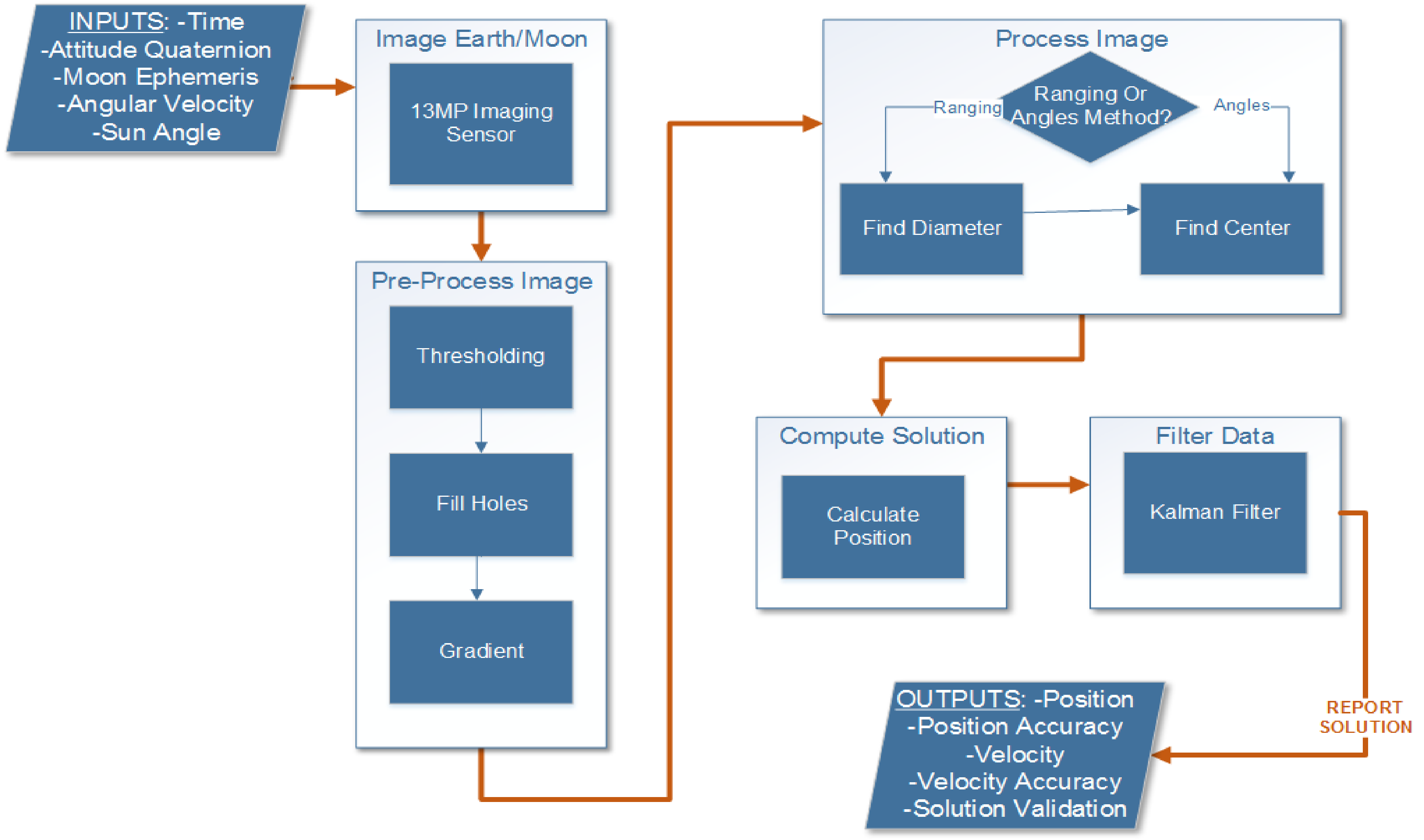

Z
Y
X


Z
Y
X

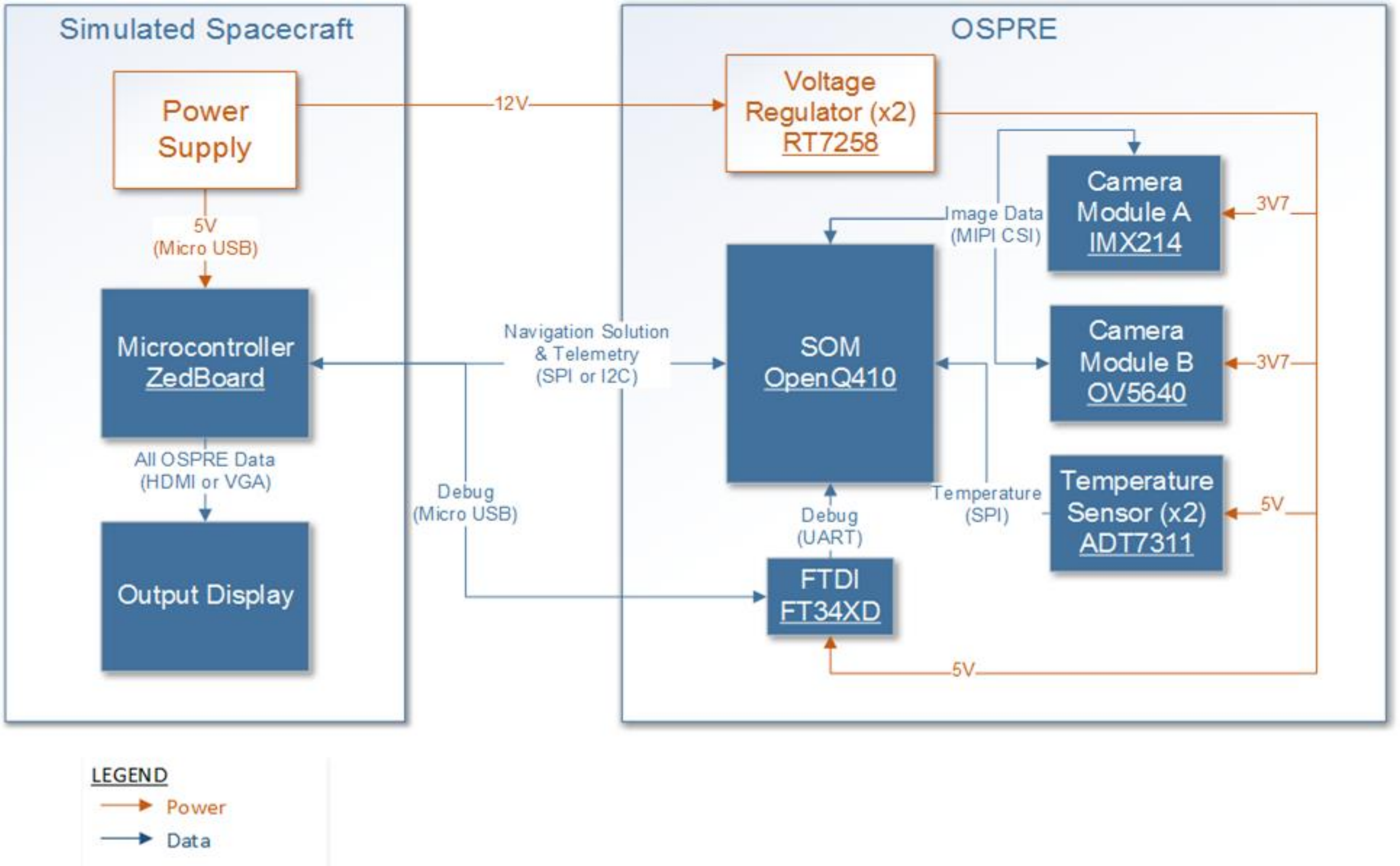

Z
Y
X
R


Z
Y
X
V
R

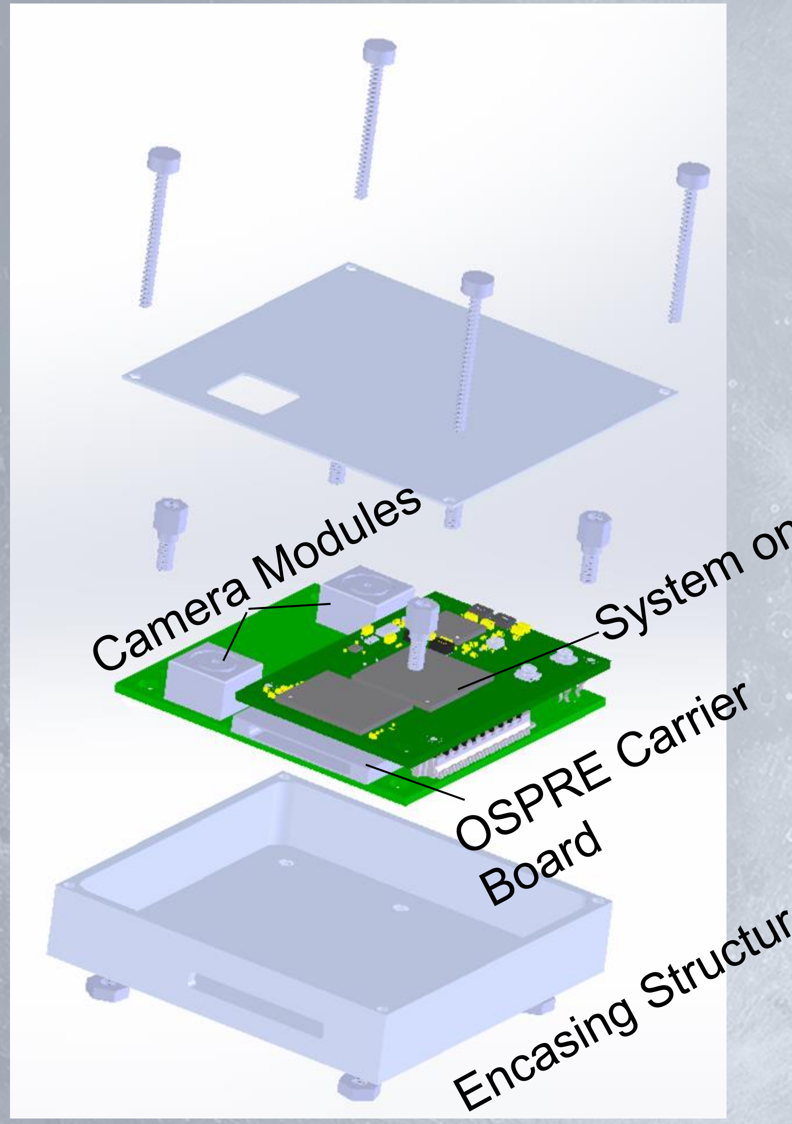
Functional Flow Diagram

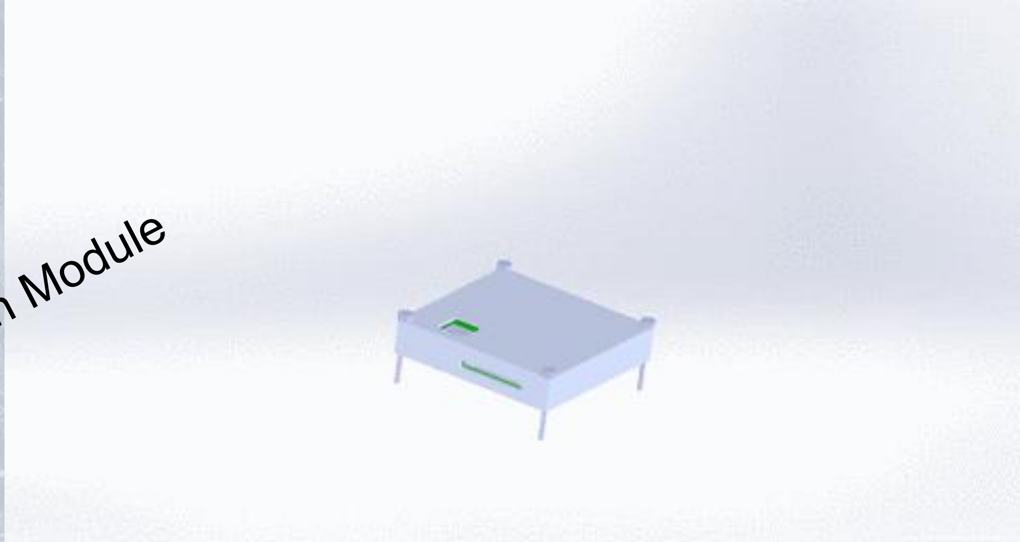


FBD


Simulated Spacecraft
Power Supply
5V (Micro USB)
Microcontroller ZedBoard
All OSPRE Data (HDMI or VGA)
Output Display
12V
Navigation Solution & Telemetry (SPI or I2C)
Debug (Micro USB)
OSPRE
Voltage Regulator (x2) RT7258
Image Data (MIPI CSI)
Camera Module A IMX214
3V7
SOM OpenQ410
Camera Module B OV5640
3V7
Temperature (SPI)
Temperature Sensor (x2) ADT7311
5V
Debug (UART)
FTDI FT34XD
5V
LEGEND
Power
Data

Final Design

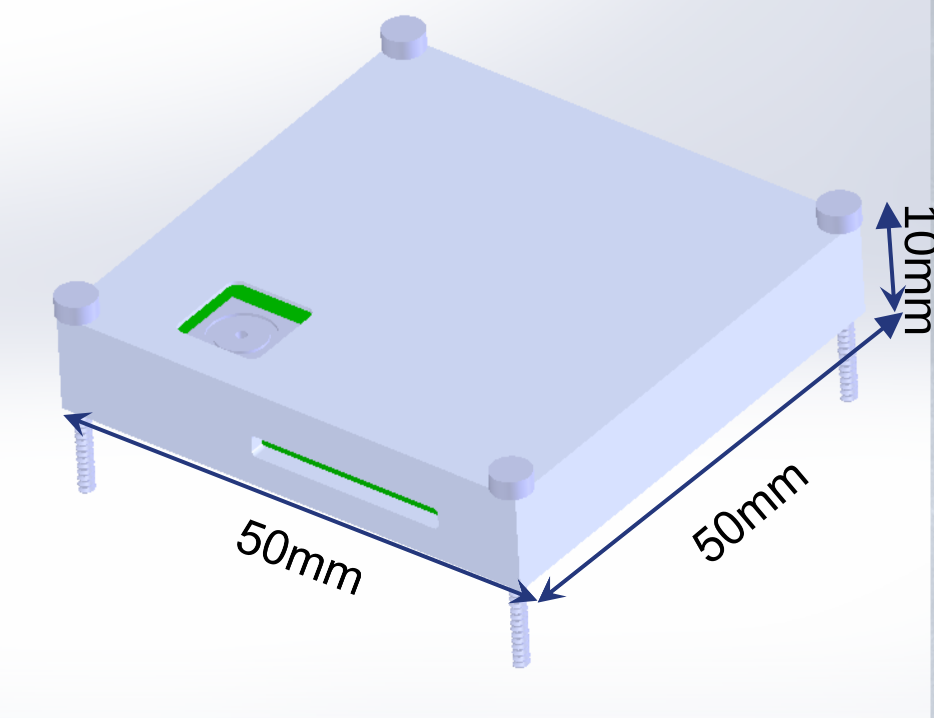




Volume	50 x 50 x 10 mm
Weight	30 g

Design Requirements and Their Satisfaction

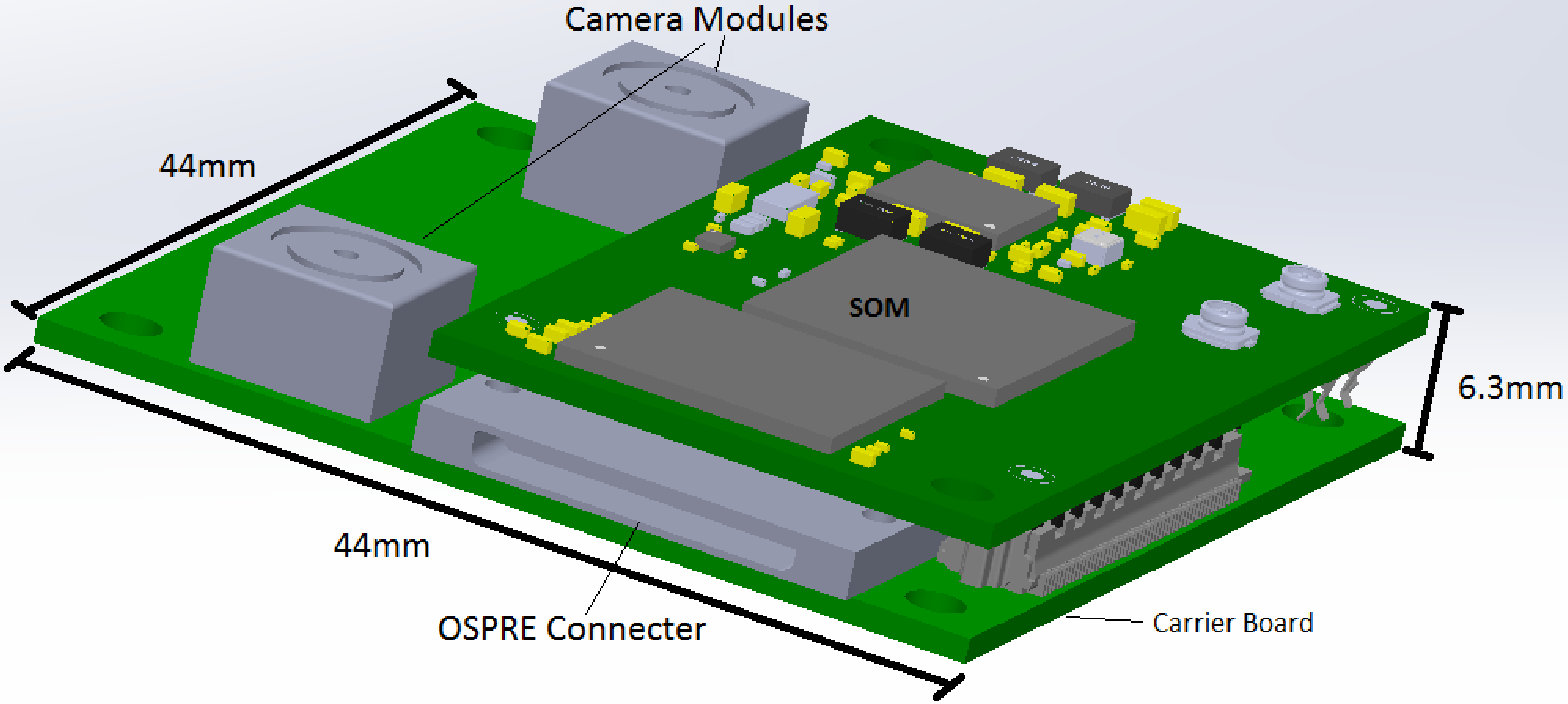
Mechanical Architecture



OSPRE Sensor Package ~30 grams

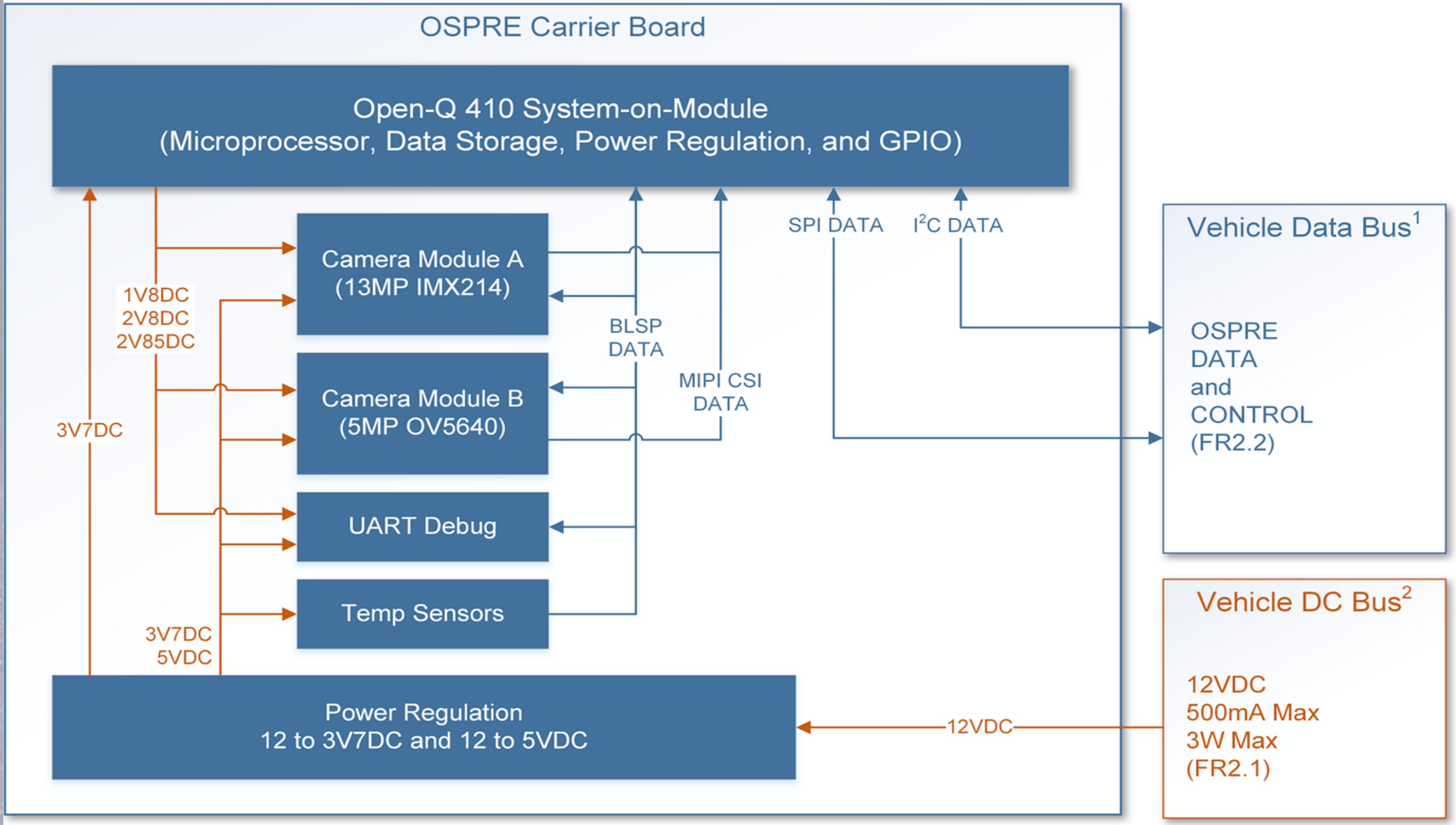
Requirement			OSPRE Design
FR2.0	DR2.6	Total Mass ≤ 800g	Approx. 30g
	DR2.7	Total Volume 50 x 50 x 10 mm	50 x 50 x 10 mm

Electrical Architecture



Electrical Architecture

Simplified Block Diagram

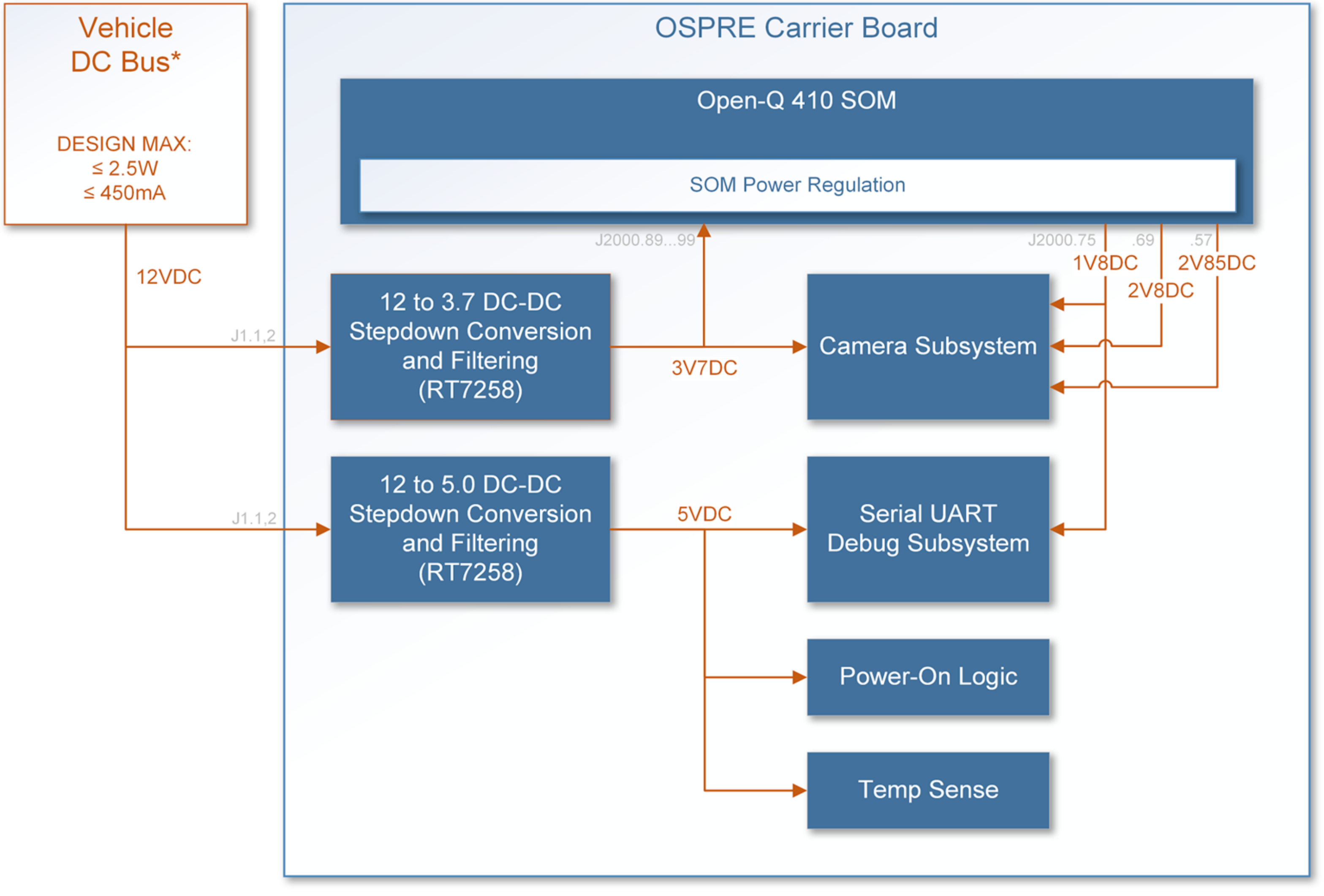


[1] Simulated with ZedBoard

[2] Simulated with bench Power Supply

Electrical Architecture

Power Regulation and Distribution



Vehicle DC Bus*

DESIGN MAX:
≤ 2.5W
≤ 450mA

12VDC

J1.1,2

J1.1,2

OSPRE Carrier Board

Open-Q 410 SOM

SOM Power Regulation

J2000.89...99

J2000.75

.69

.57

1V8DC

2V85DC

2V8DC

12 to 3.7 DC-DC Stepdown Conversion and Filtering (RT7258)

3V7DC

Camera Subsystem

12 to 5.0 DC-DC Stepdown Conversion and Filtering (RT7258)

5VDC

Serial UART Debug Subsystem

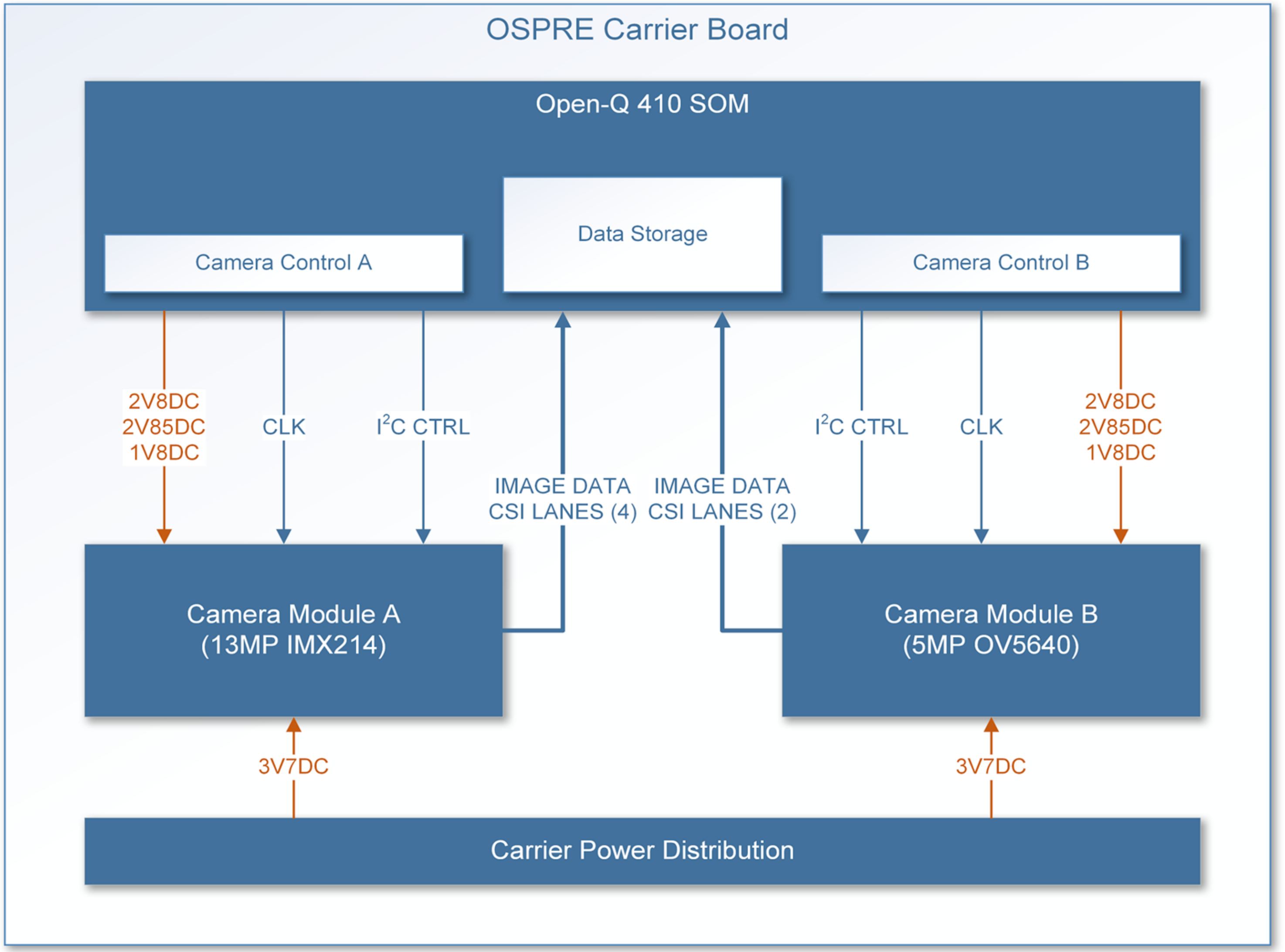
Power-On Logic

Temp Sense

*Simulated with bench DC supply

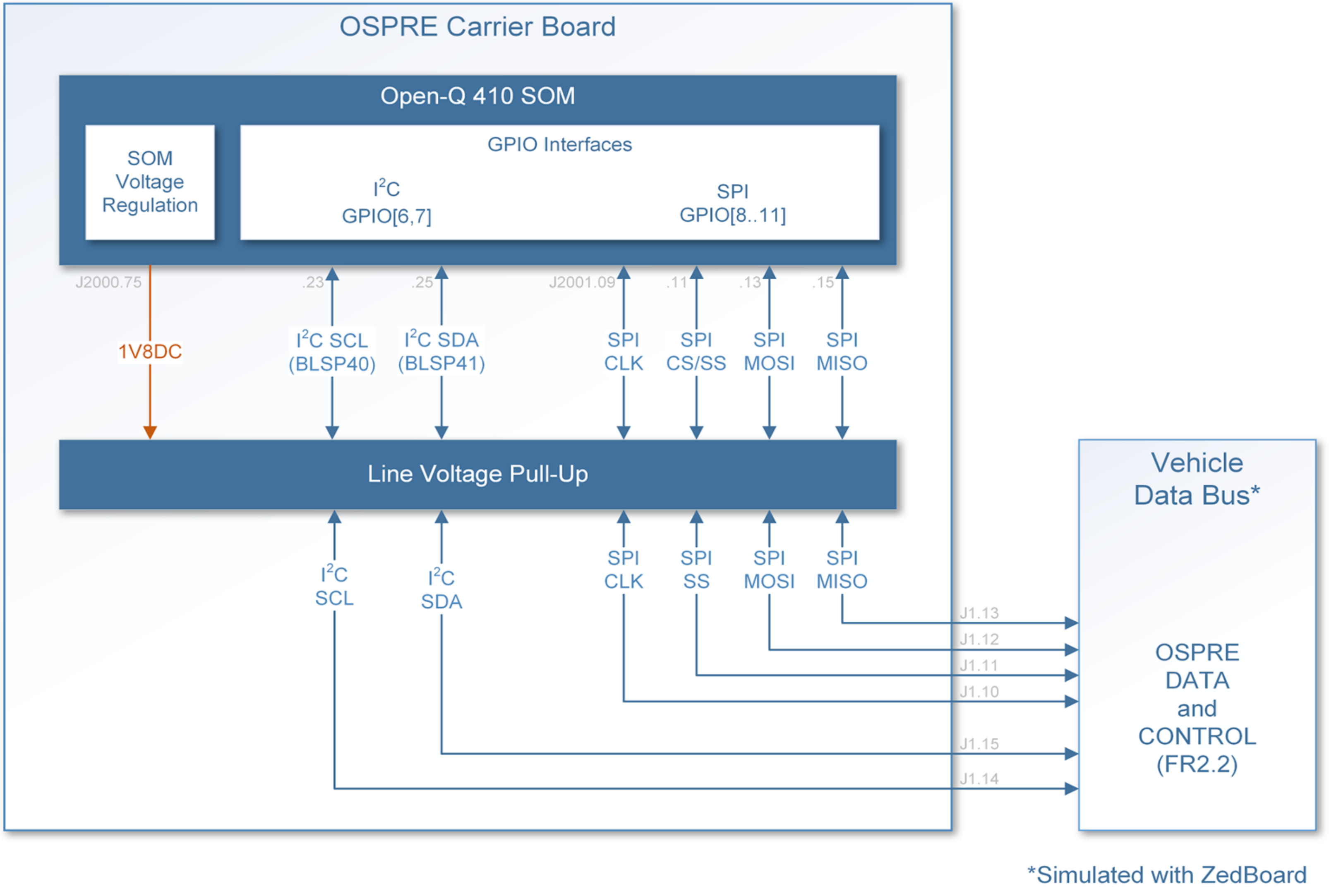
Electrical Architecture

Camera Power, Control, and Data



Electrical Architecture

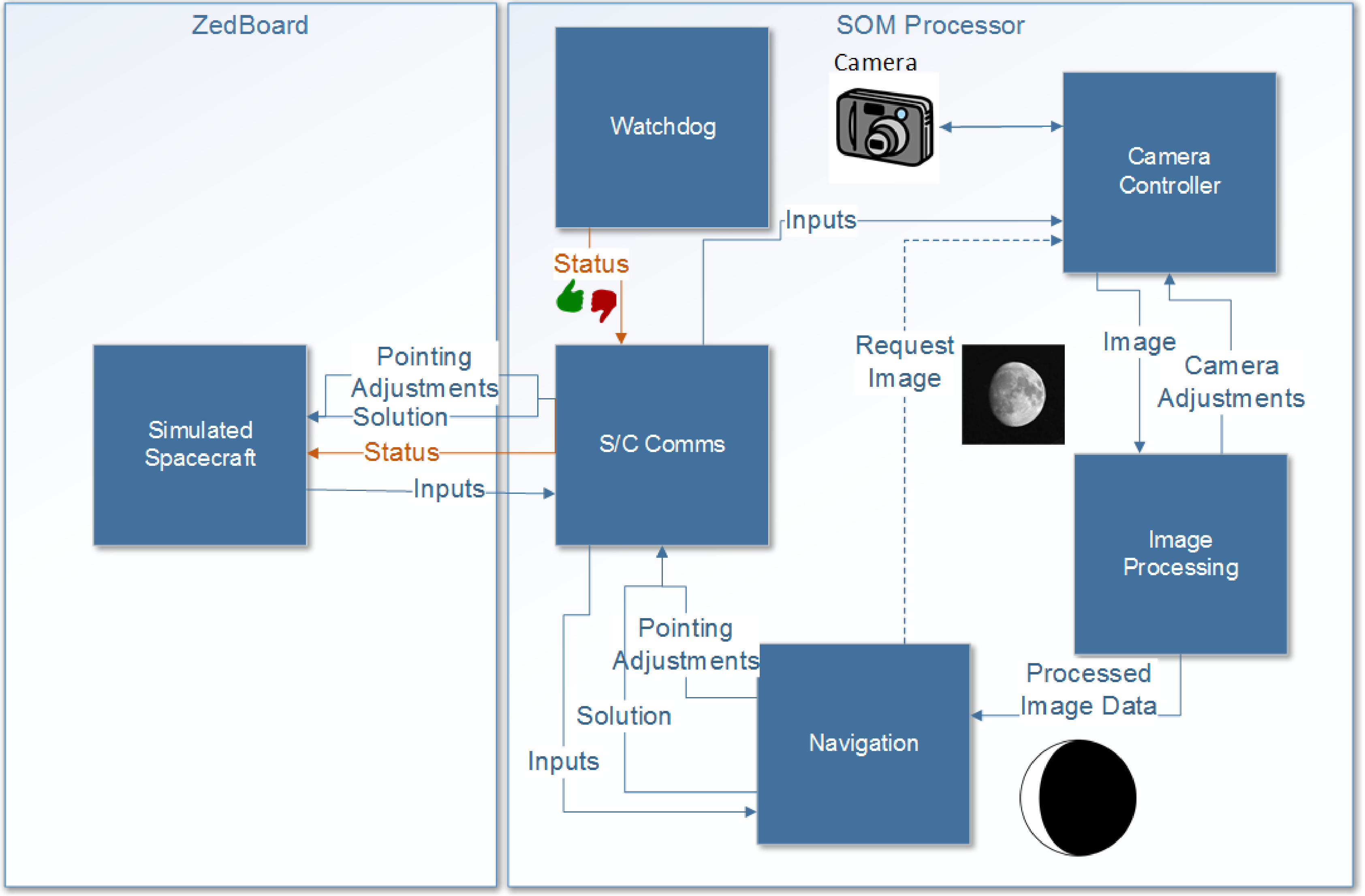
I^{2}C and SPI Data



Electrical Architecture

Requirement			OSPRE Design	Verification
FR0.0	DR0.0	Capability to acquire images of Earth, Moon	Supports up to two MIPI Camera Modules	SW & HW Inspection, Operational Test
FR2.1	DR2.1.1	Operating Voltage of 3.3, 5, or 12 VDC	12VDC	Electrical Test
	DR2.1.2	Peak Current NGT 500mA	≤ 450 mA	Electrical Test
	DR2.1.3	Peak Power NGT 3W	≤ 2.5W	Electrical Test
FR2.2	DR2.2.1	Comms via SPI, I2C, and/or CamLink	SPI & I^2C	SW & HW Inspection, Operational Test

Software Architecture



Software Architecture

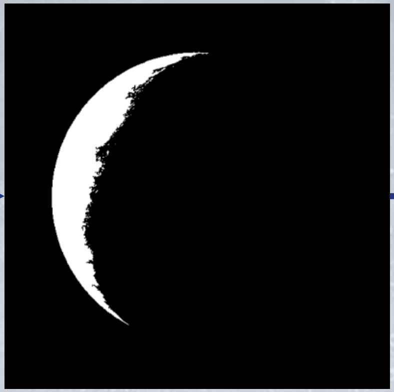
Requirement			OSPRE Design	Verification
FR1.0	DR1.0	Process image and compute a state vector update.	Camera Controller, Image Processing, Navigation Processes	Software Functional Test/ Operational Systems test
FR1.1	DR1.2	Receive required Inputs outlined by Lockheed Martin.	Simulated S/C with controlled inputs	Software Functional Test
FR2.3	DR2.3	Operate on a ZedBoard controller.	Simulated S/C (ZedBoard) Process	Software Functional Test
FR2.4	DR2.4	Output telemetry to the simulated spacecraft.	Watchdog Process	Software Functional Test/ Operational Systems Test

Image Processing

Original Image

Thresholding

Fill Holes



Gradient

Coherent Circular Hough Transform

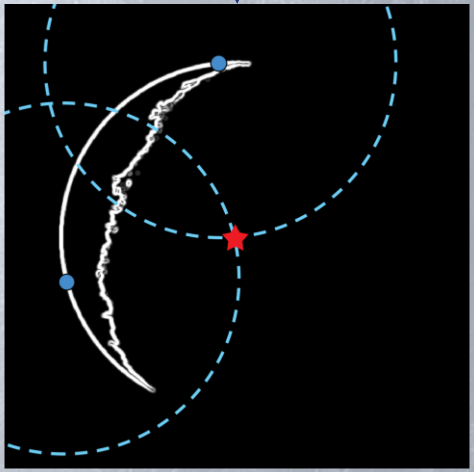
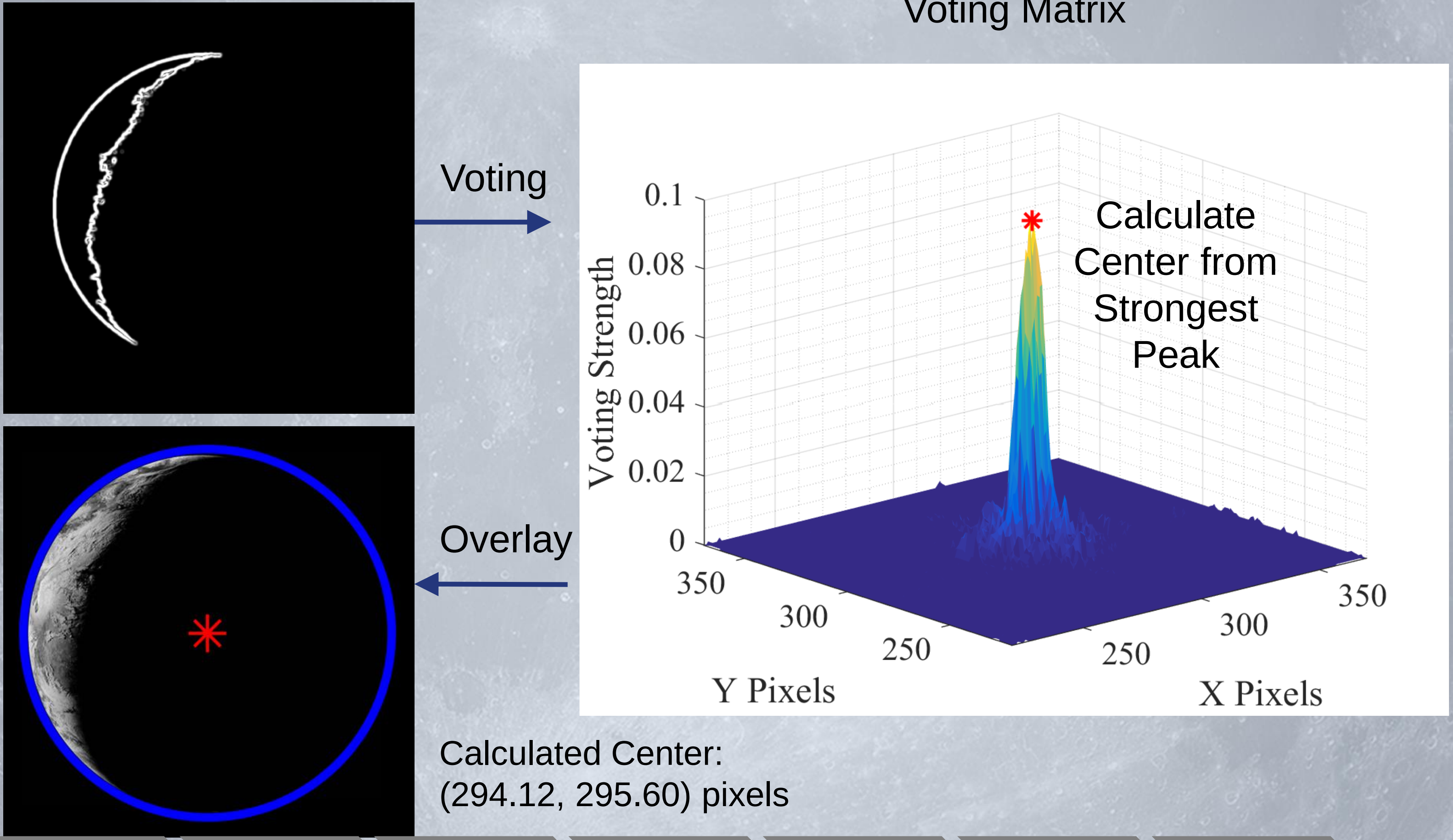


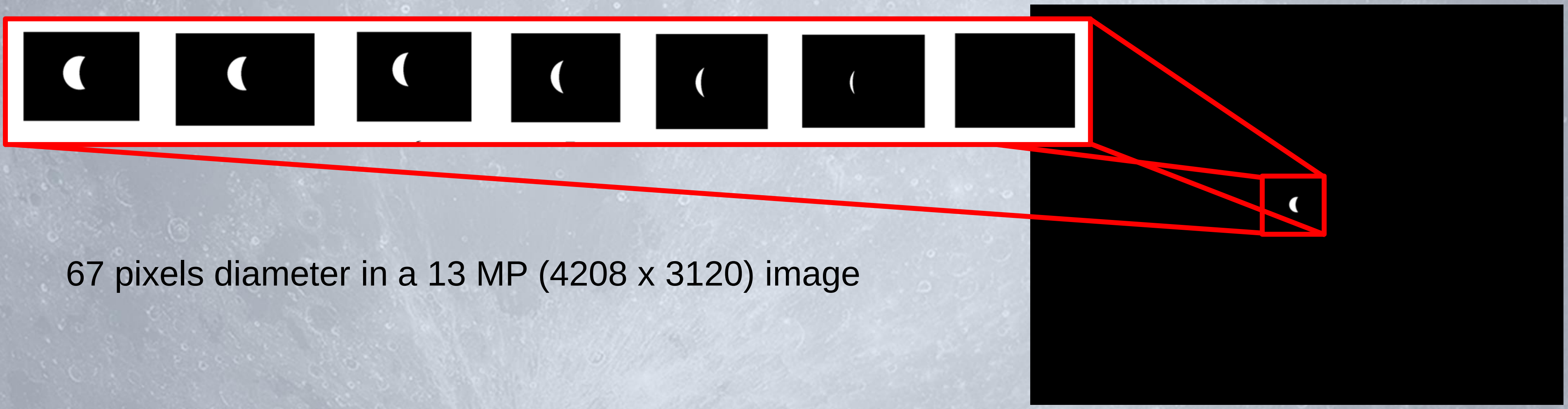
Image Processing



Calculate Center from Strongest Peak

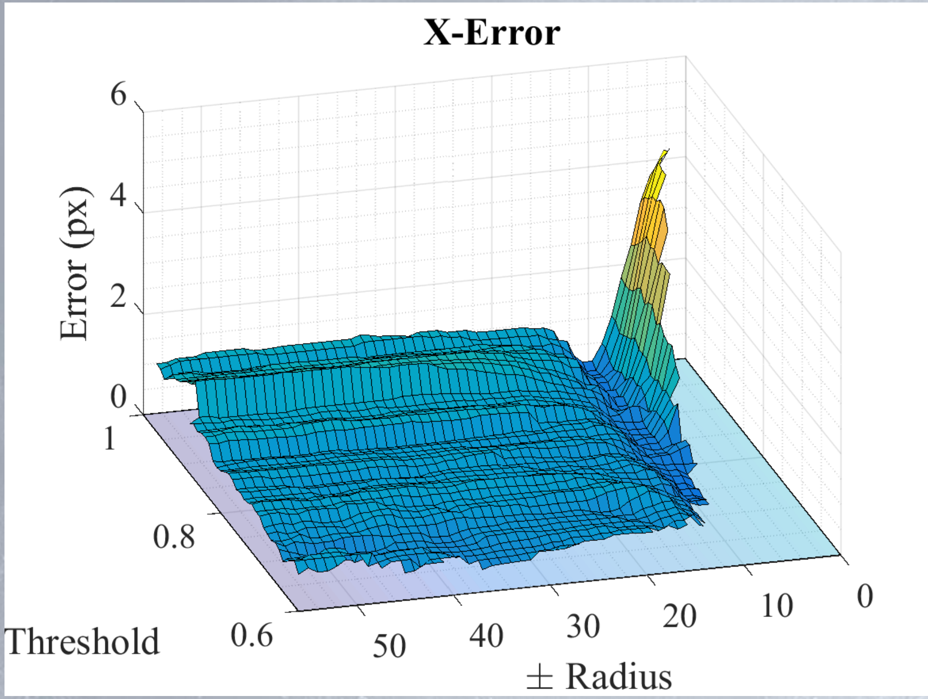
Calculated Center:
(294.12, 295.60) pixels

Image Processing

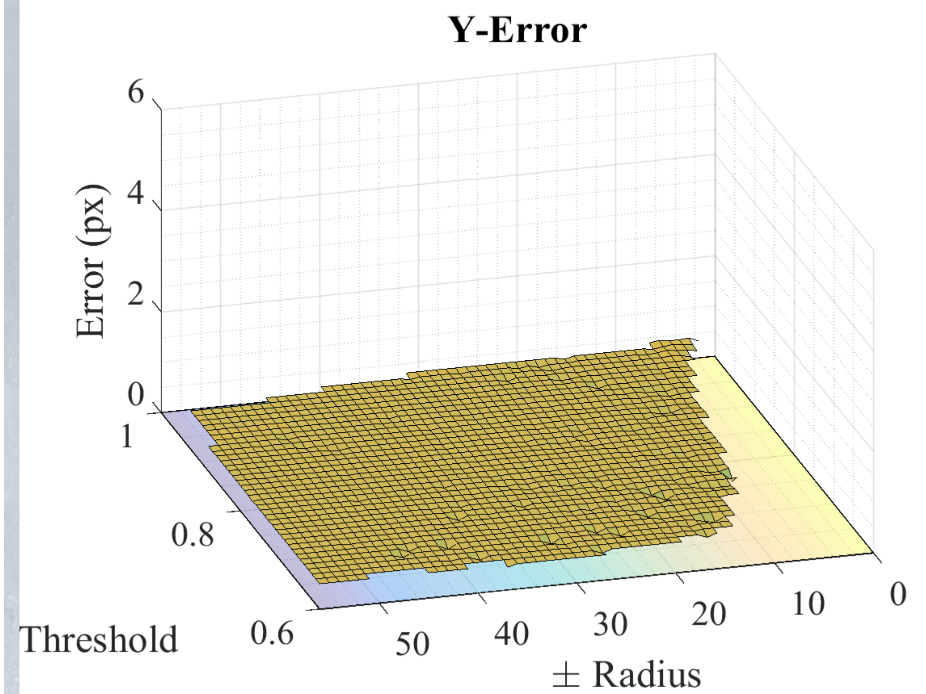


67 pixels diameter in a 13 MP (4208 x 3120) image

Image Processing



X-Error



Y-Error

- Average of 0.7 px error for most parameters

- Average of 0.5 px error
- Symmetry explains flat average

Threshold - voting strength kept, lower = less votes kept, higher = more votes kept
±*Radius* - Plus and minus range of radius to analyze; more information in backup

Navigation

- Three position algorithms
  - require images of Earth, Moon, or both
  - most accurate one is chosen at each point in time during flight
- Run through a data filter to increase accuracy and compute velocity and error.

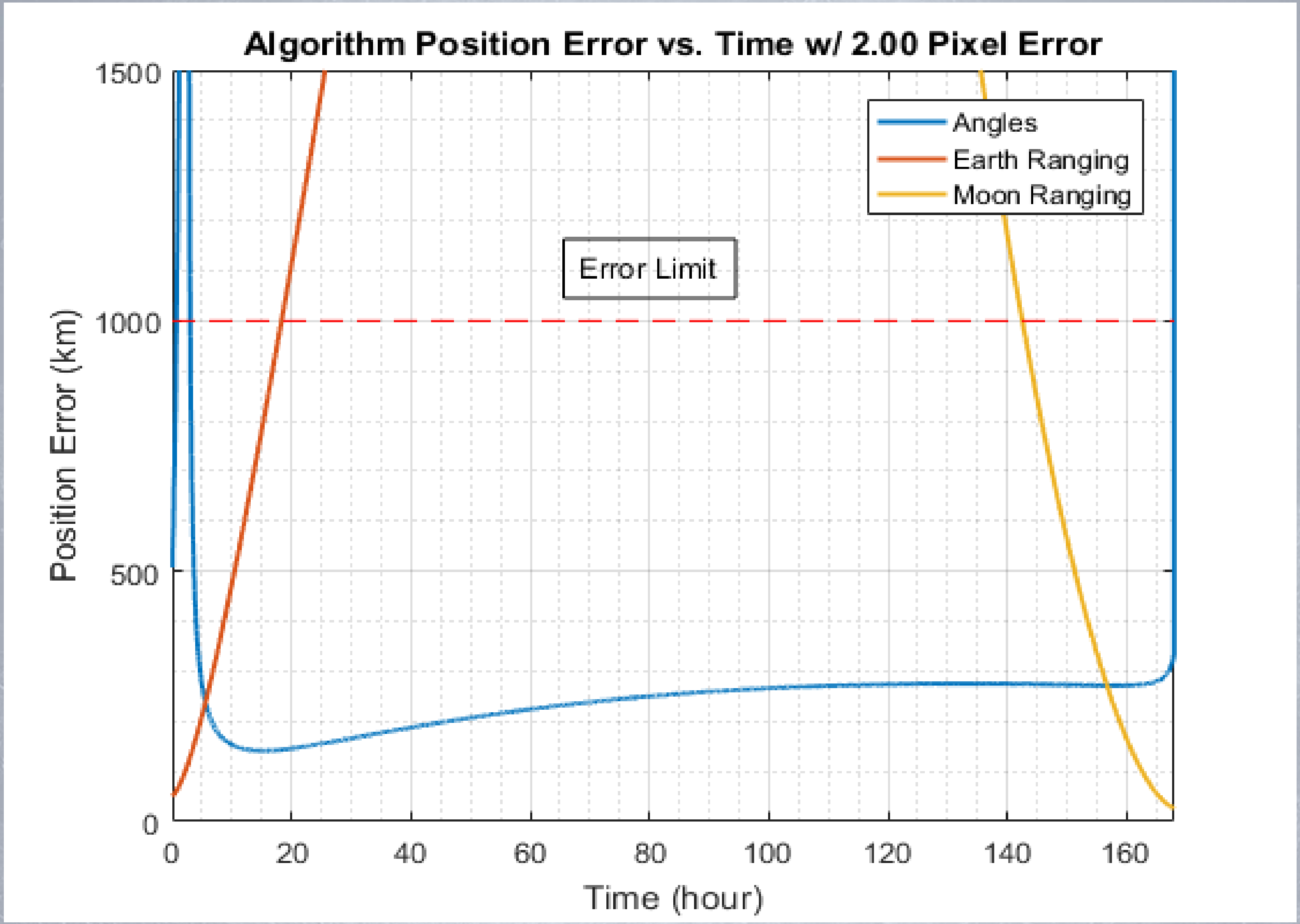
Navigation

Algorithms

Ranging Algorithms (Earth and Moon)	Angles Algorithm
1. Ranging from Earth or Moon 2. Calculates attitude to body center. 3. Calculates angular diameter from edge-finding and camera specifications. 4. Calculates position.	1. Calculates attitude to first body. 2. Calculates attitude to second body. 3. Propagates second position from first position velocity. 4. Calculates first position from relative geometry.

Navigation

Algorithms



Navigation

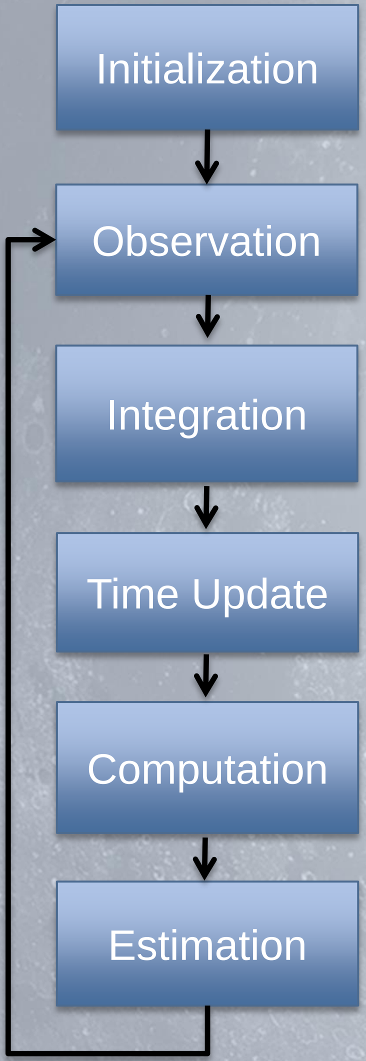
Kalman Filter

Kalman Filter

- Mitigates error using estimated dynamics and errors
  - – inherent measurement noise may be high
  - – dependent on orbit and image processing
- Provides velocity
- Provides error in both position and velocity

Navigation

Kalman Filter



Initialization

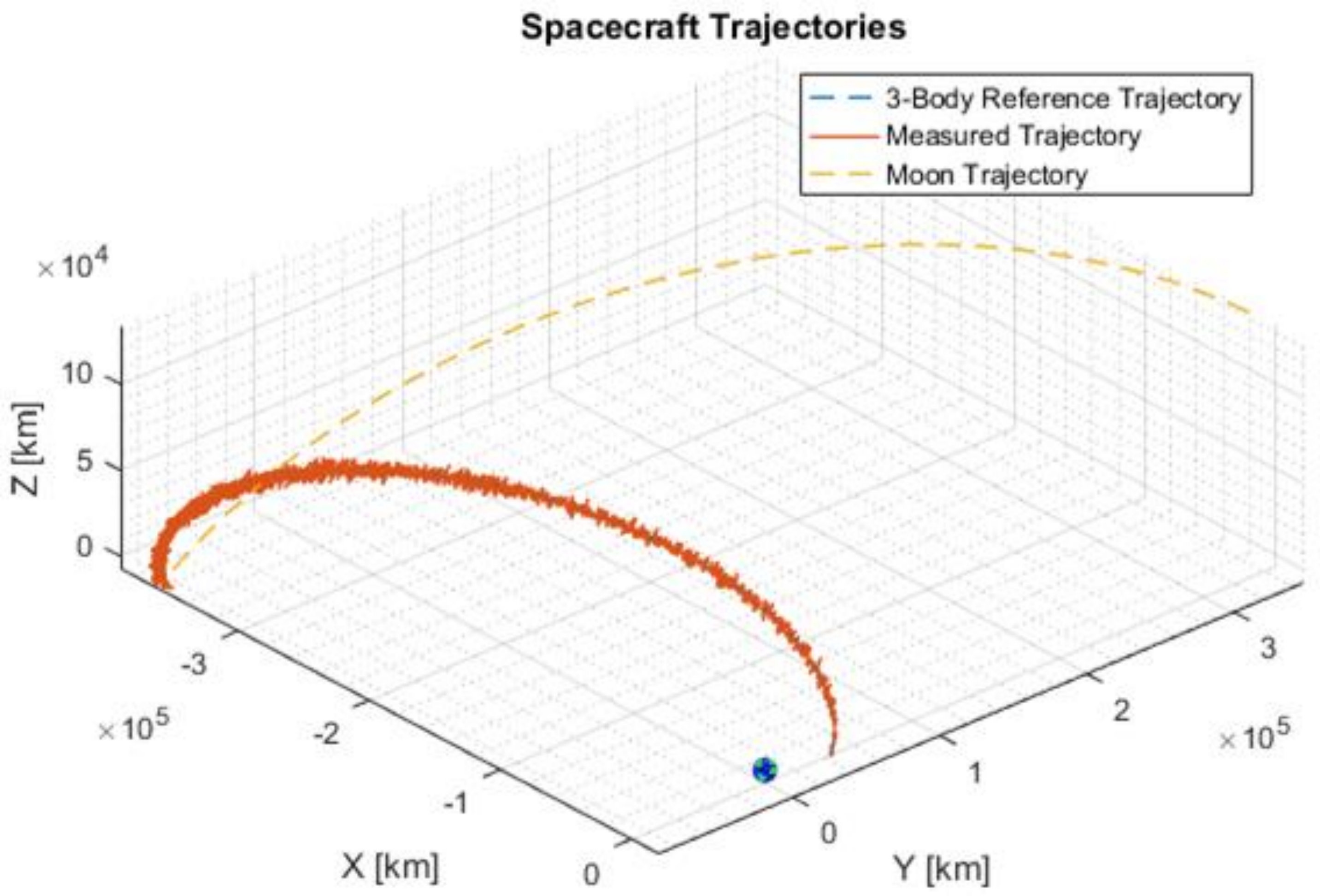
Observation

Integration

Time Update

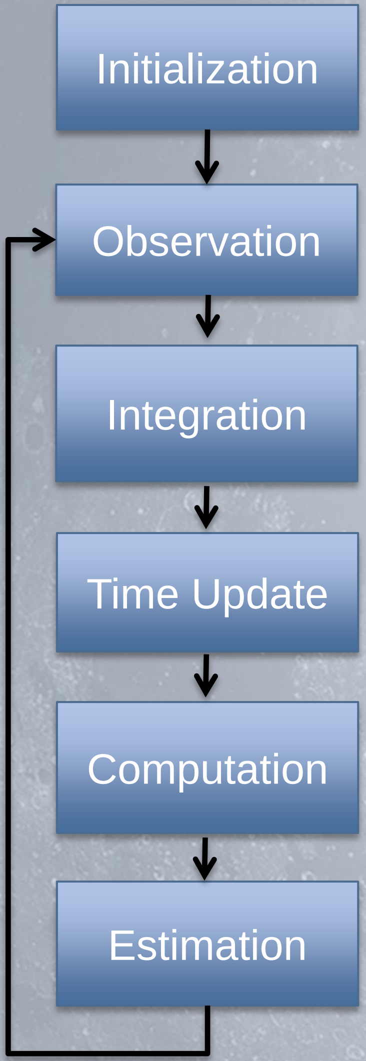
Computation

Estimation



Navigation

Kalman Filter



Initialization

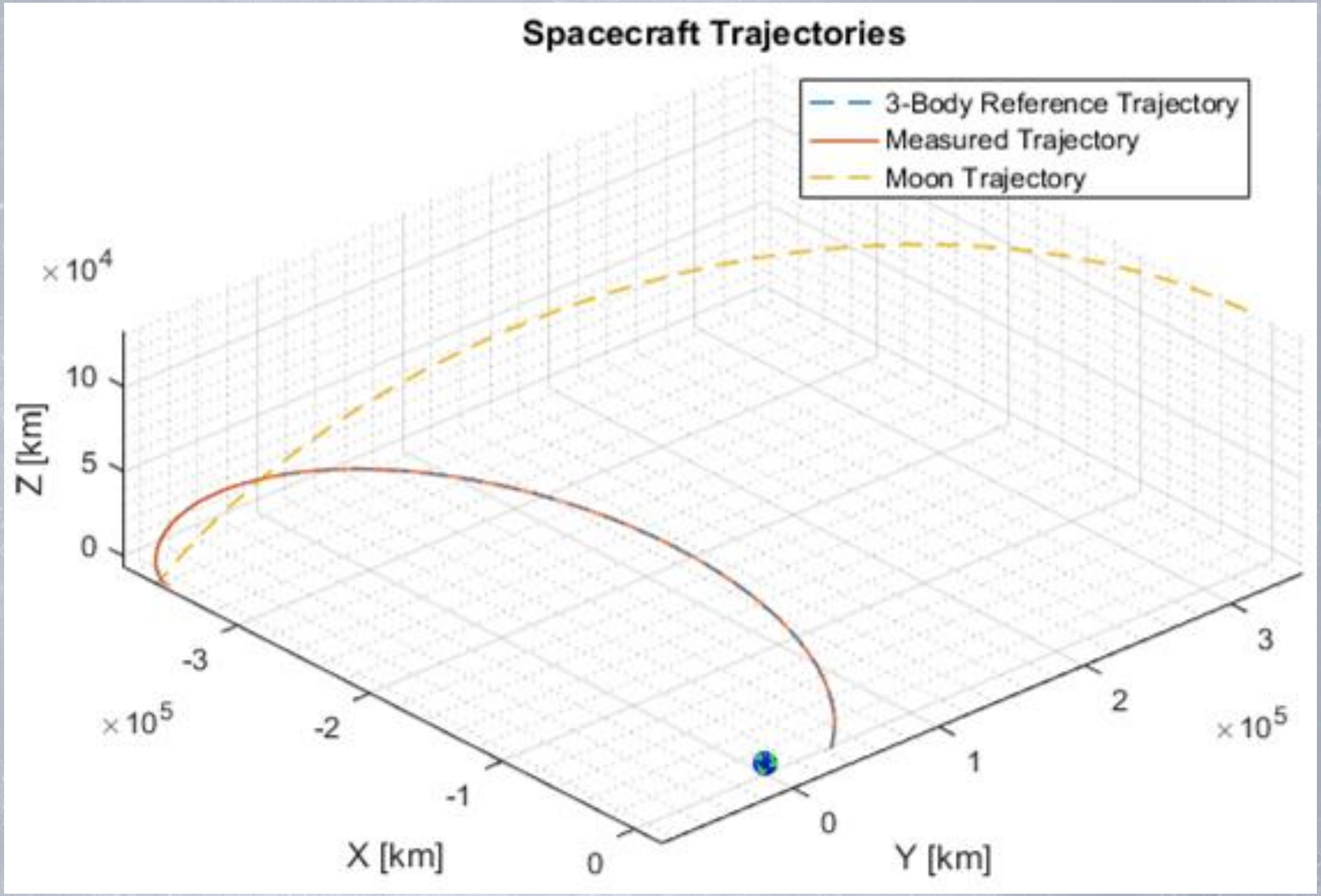
Observation

Integration

Time Update

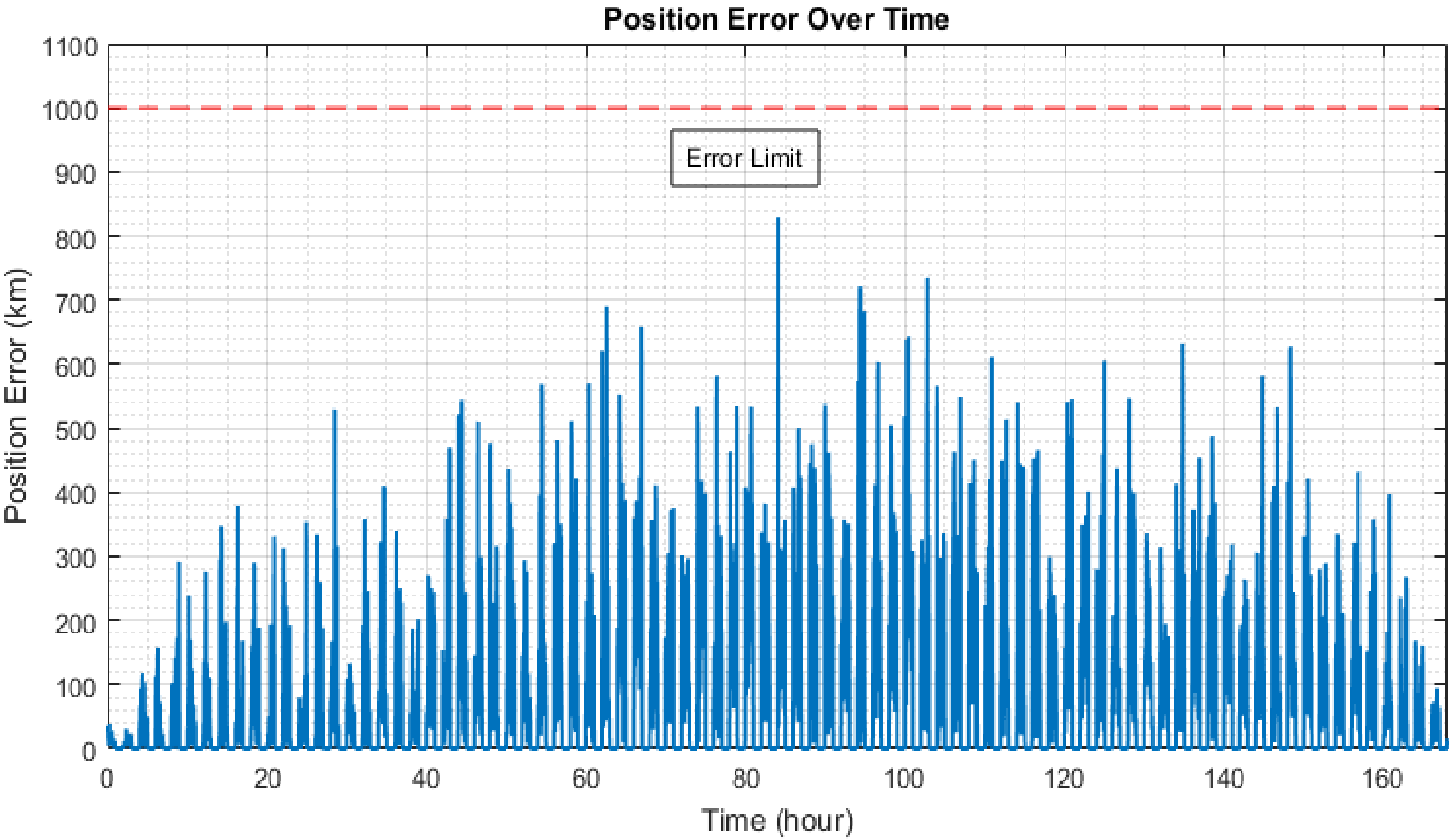
Computation

Estimation



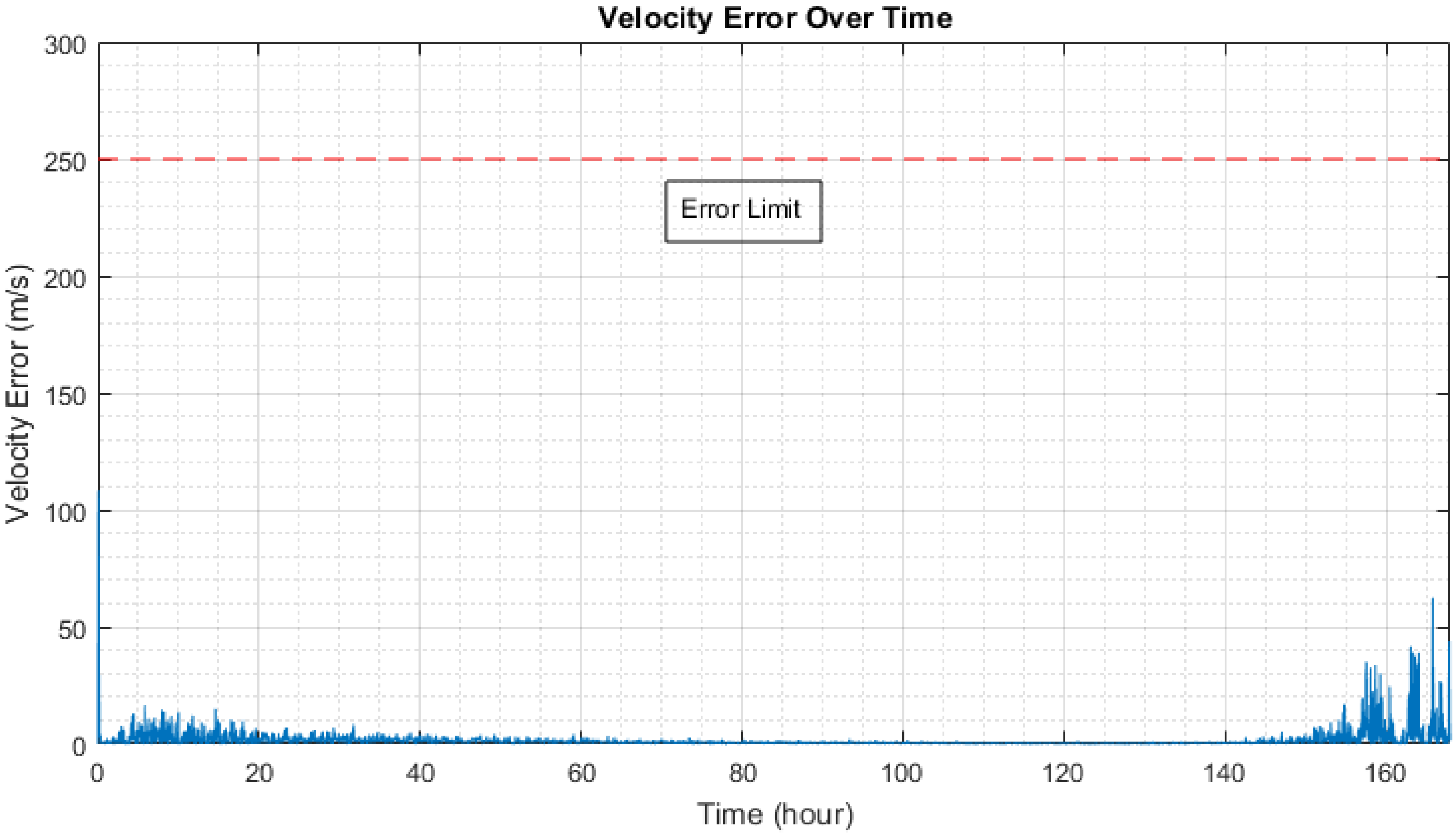
Navigation

Kalman Filter



Navigation

Kalman Filter



Navigation

Requirement			OSPRE Design	Verification
FR1.1	DR1.1.3	Computation of the angles between the simulated spacecraft, Earth, and Moon	3 Positional algorithms	Software Test
	DR1.1.4	Include the estimation of the range to the Earth and Moon from the simulated spacecraft	3 Positional algorithms	Software Test
	DR1.1.5	Output the computed state vector update and error in the ECI reference frame.	Kalman Filter	Software Test
FR1.2	DR1.2	Computation of the position and the position error of the simulated spacecraft and be capable of achieving an error of less than 1000km from actual position.	3 Positional algorithms and Kalman Filter	Software Test
FR1.3	DR1.3	Computation of the velocity and velocity error of the simulated spacecraft and be capable of achieving an error of less than 250 m/s from actual velocity.	Kalman Filter	Software Test

Verification and Validation

Testing

1. **Operational Systems Test**

- **FR 1.1** - State Vector Calculation
- **FR 1.2** - Position Accuracy
- **FR 1.3** - Velocity Accuracy
- **FR 1.4** - ESM Angle
- **FR 1.5** - Solution Validity
- **FR 2.5** - Data Collection
- Satisfies customer requirement

Takeaway - Test is Vital for Project Success

Key Testing Aspects

RANGING METHOD:

How large does the celestial body appear in the image?

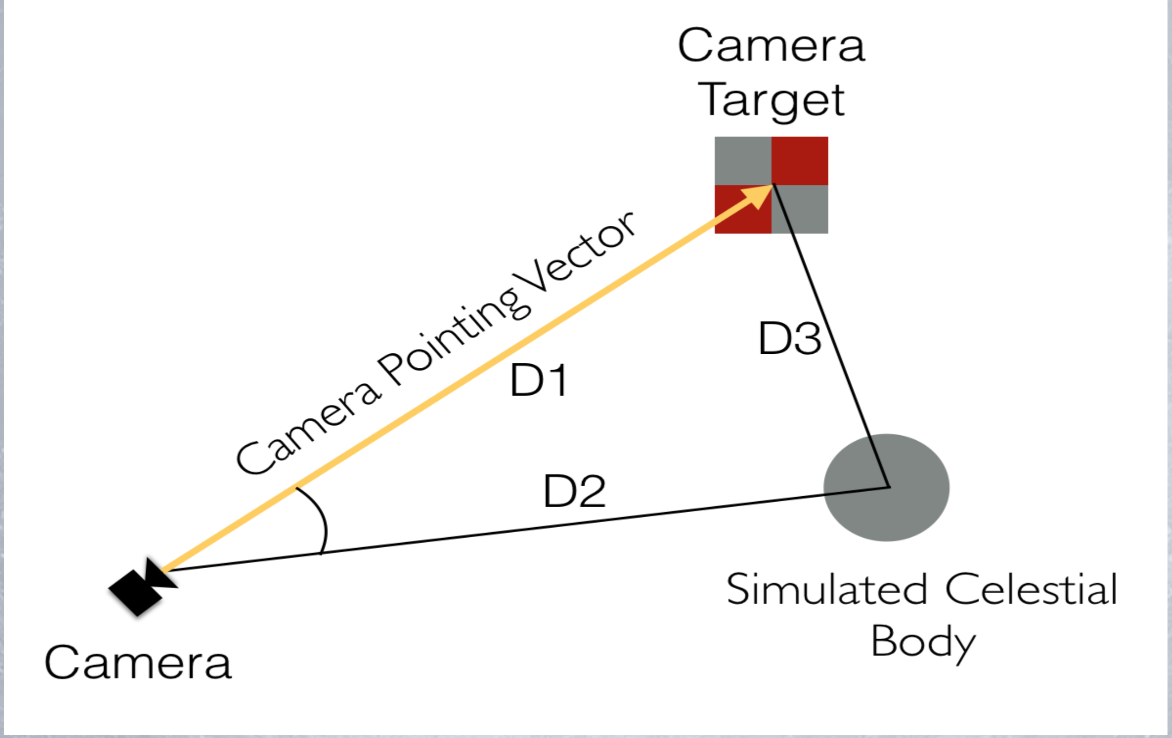
ANGLES METHOD:

How large does the celestial body appear in the image?

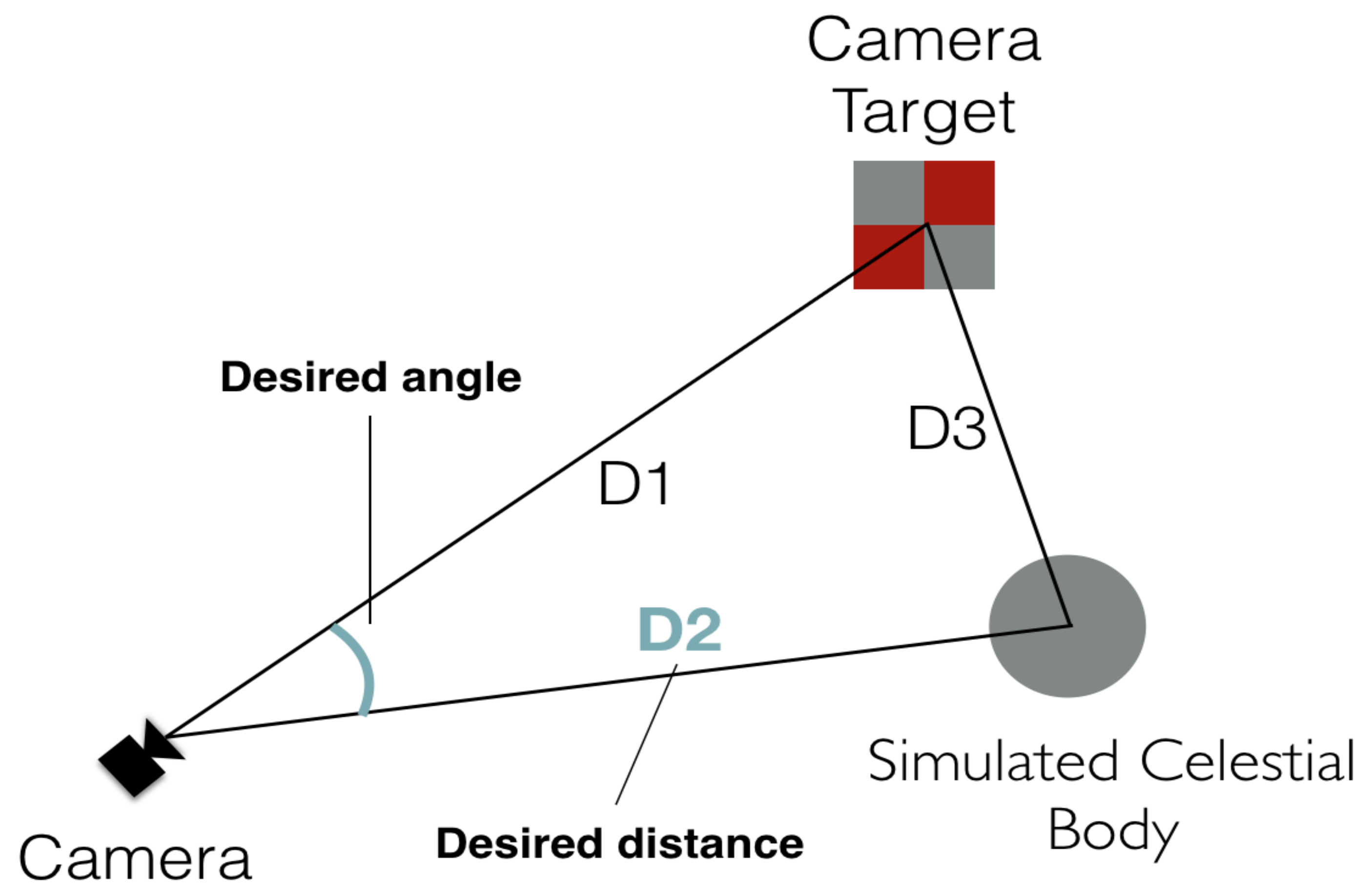
Where exactly does the celestial body appear in the image?

Operational Systems Testing

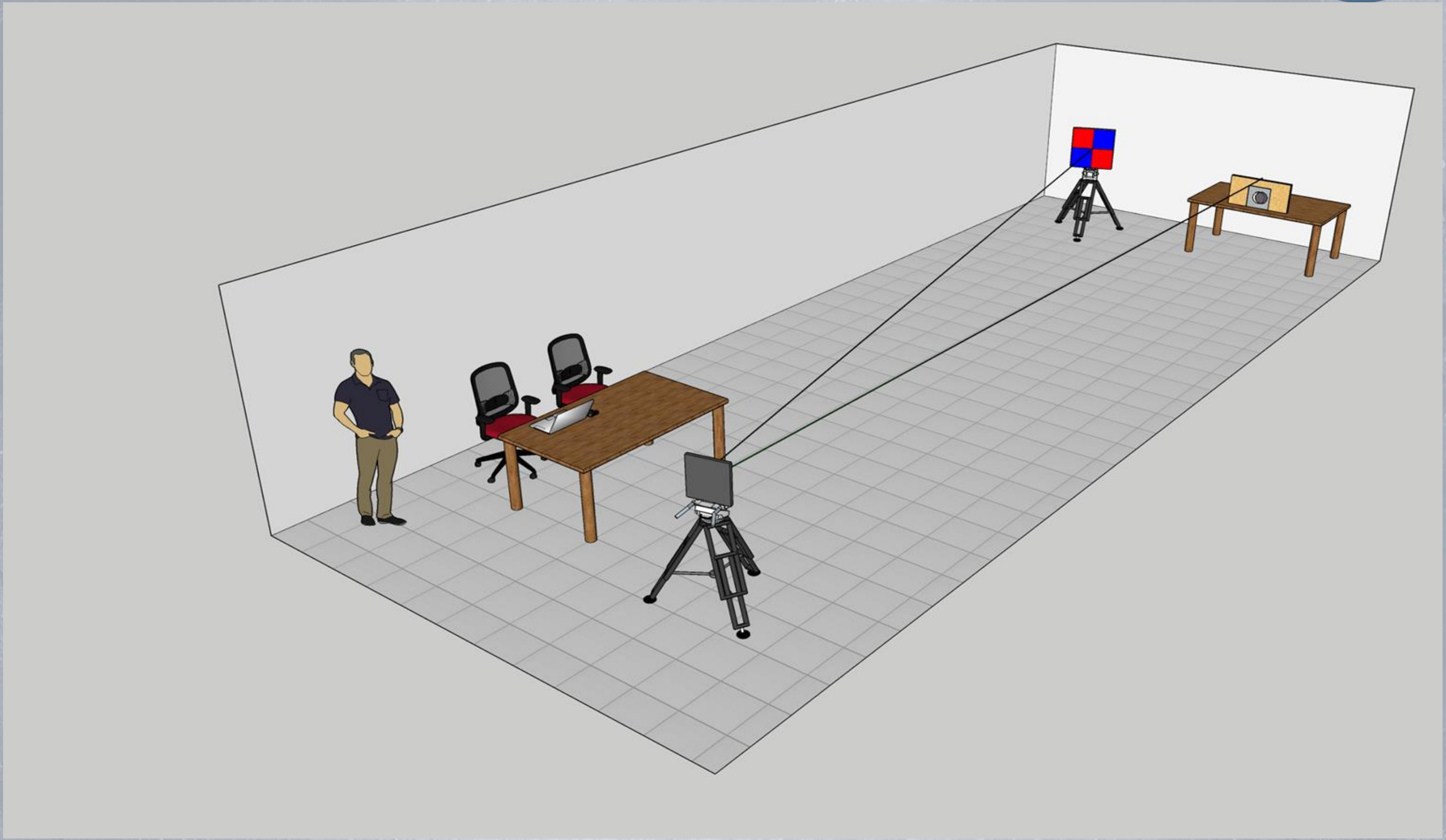
- Will allow position calculation
- Here’s what it looks like:



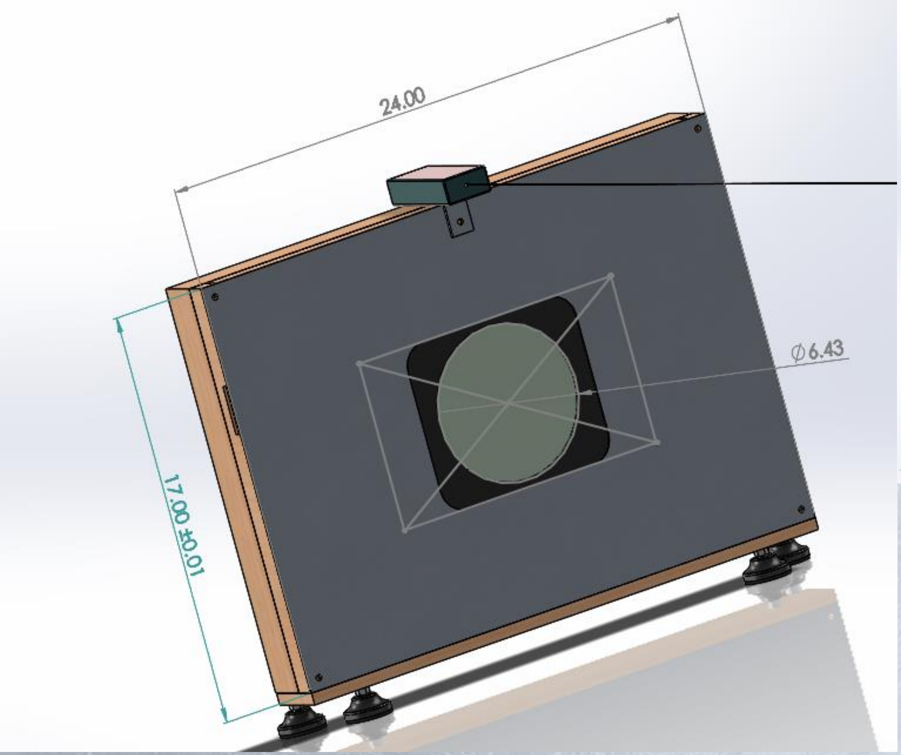
Operational Systems Testing

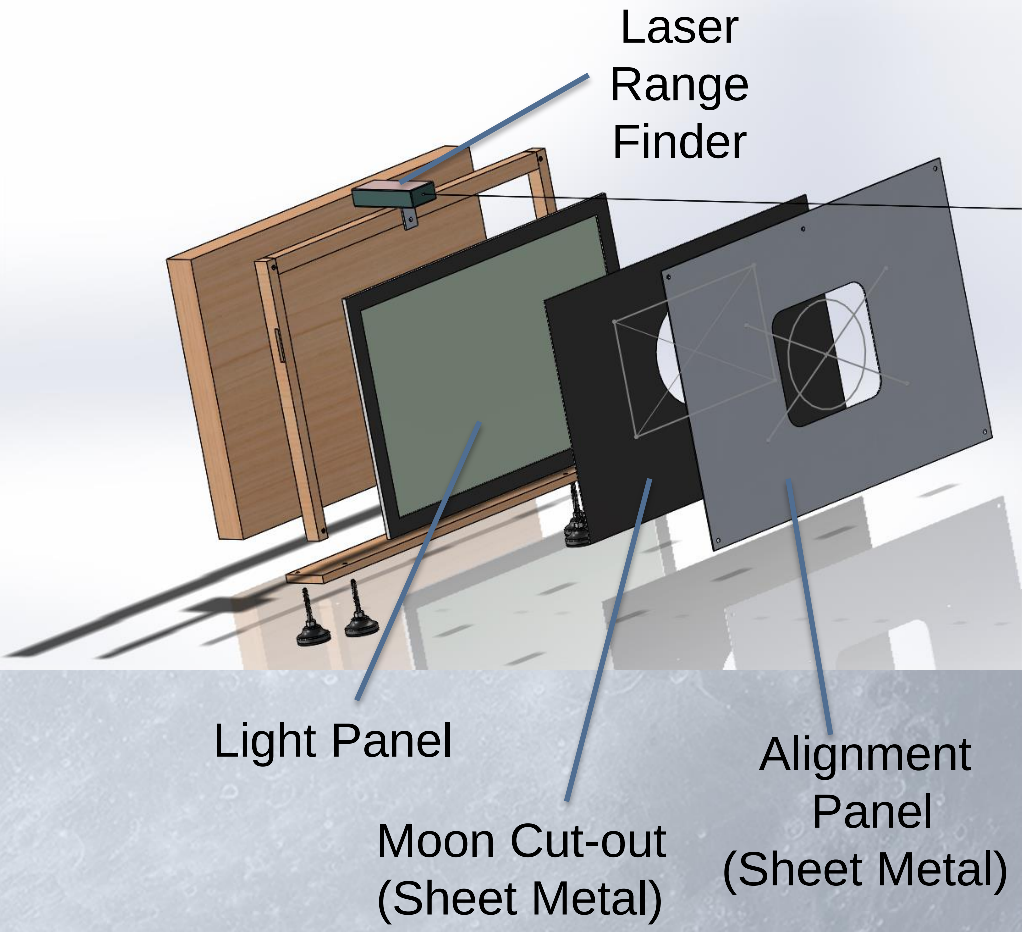


Operational Systems Test

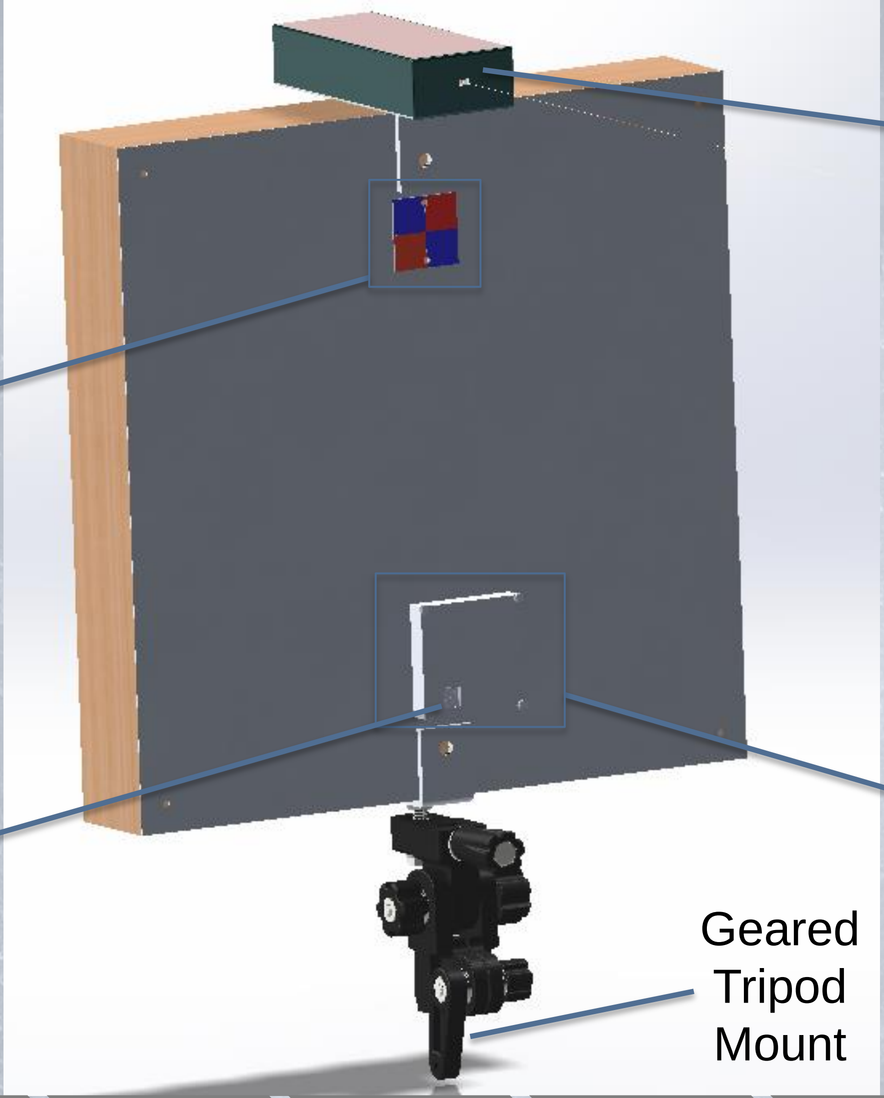


LightBox


24.00
∅6.43
17.00 ±0.01


Laser Range Finder
Light Panel
Moon Cut-out (Sheet Metal)
Alignment Panel (Sheet Metal)

Camera Mount



Laser Range Finder

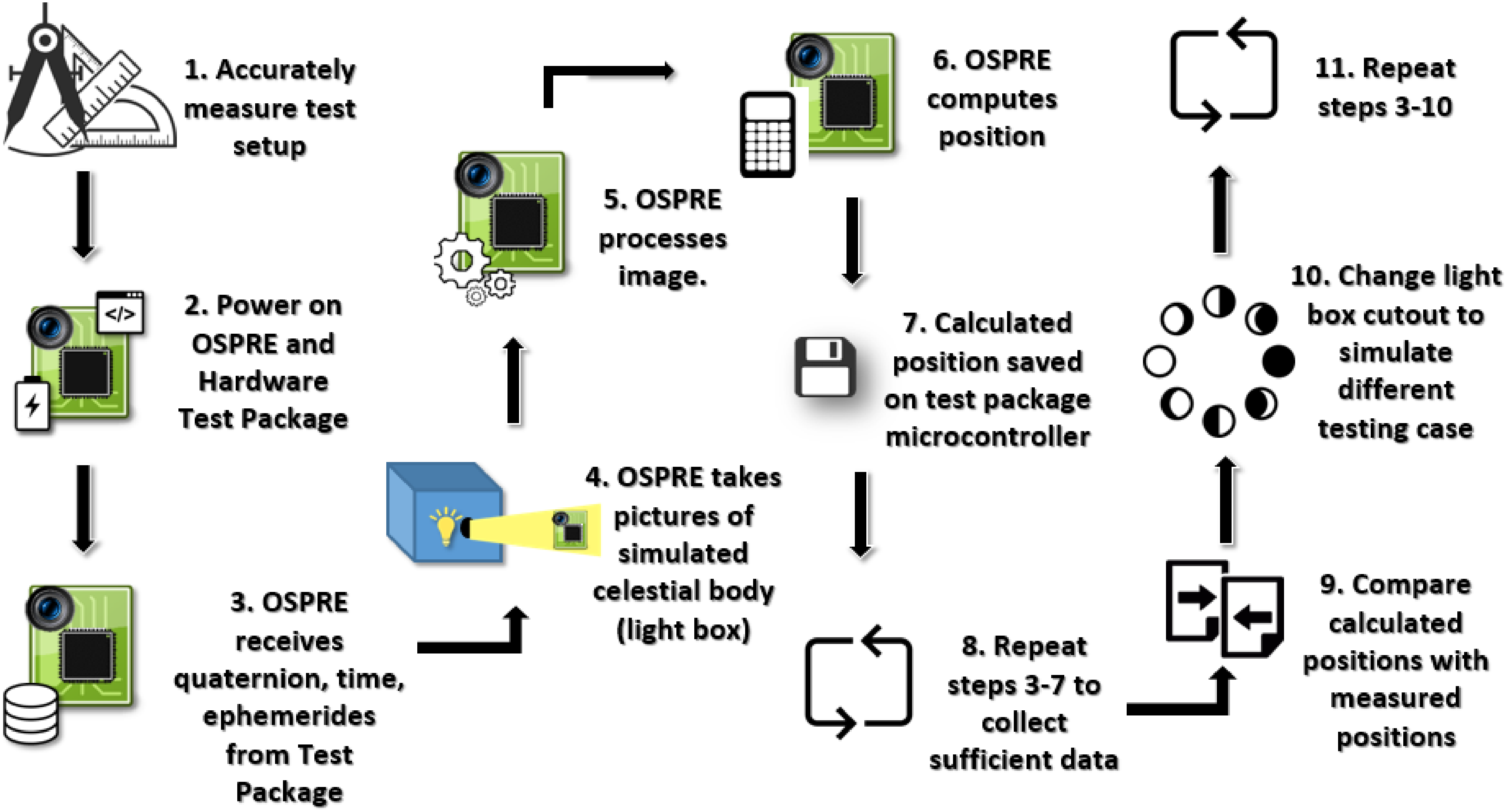
Laser Target (Acrylic)

OSPRE Camera

OSPRE System

Geared Tripod Mount

Testing CONOPS



1. Accurately measure test setup

2. Power on OSPRE and Hardware Test Package

3. OSPRE receives quaternion, time, ephemerides from Test Package

4. OSPRE takes pictures of simulated celestial body (light box)

5. OSPRE processes image.

6. OSPRE computes position

7. Calculated position saved on test package microcontroller

8. Repeat steps 3-7 to collect sufficient data

9. Compare calculated positions with measured positions

10. Change light box cutout to simulate different testing case

11. Repeat steps 3-10

Operational Systems Testing Error

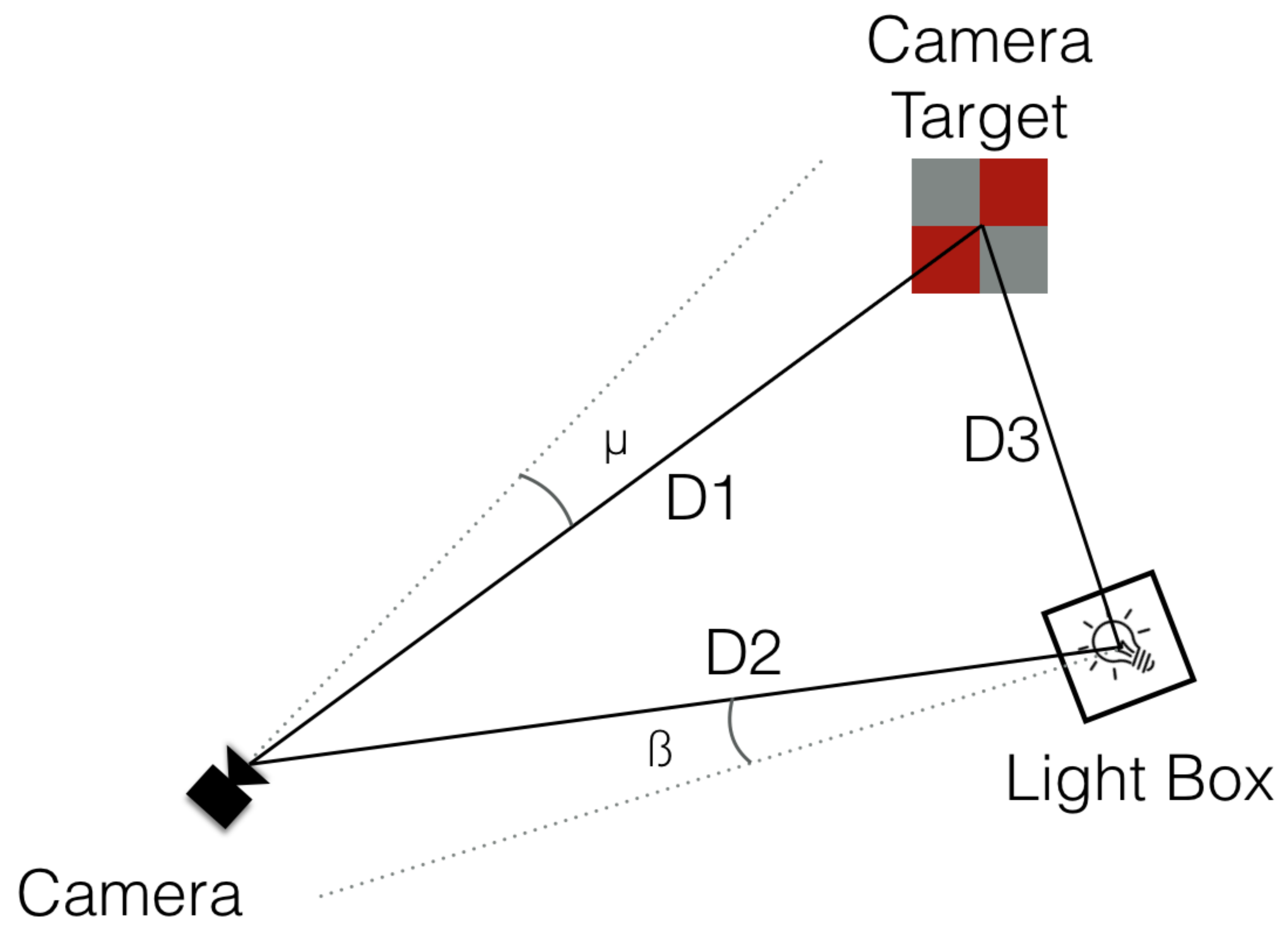
OBJECTIVE

Make test-attributed error lesser than 5% (50 km scaled) for all mission test cases to allow for statistically significant OSPRE position error quantification

IMPLICATIONS

For the worst-case mission scenario the test must introduce less than 0.0221% error (50 km position error over 226,030 km scaled distance)

Operational Systems Testing Error



Operational Systems Testing Error

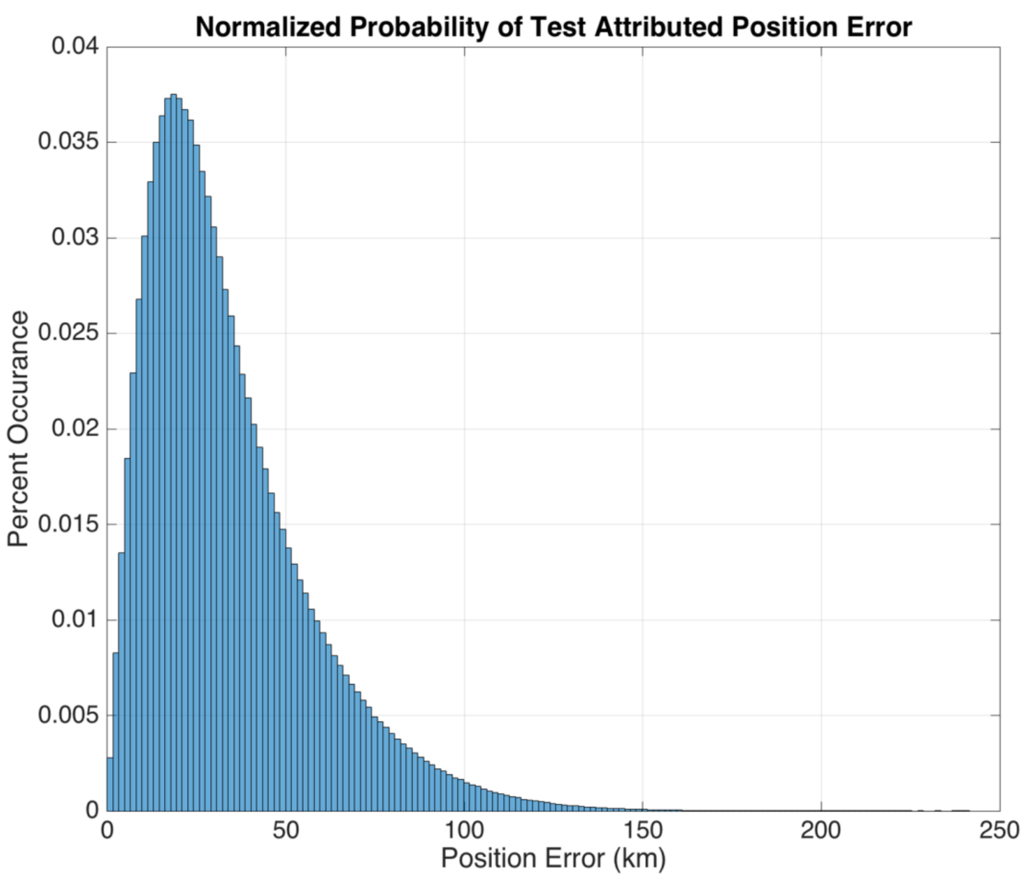
Source	Associated Error	Source	Human Error Margin
Lightbox Pointing	± 0.2˙	Alignment error due to equipment	x 1.5 margin
Camera Pointing	± 0.007˙	0.5 pixel error	x 1.1 margin
Calipers	± 0.025 mm	Advertised equipment error	x 1.5 margin
Steel Tape Measure	± 1.1 mm	Advertised equipment error	x 1.5 margin
Laser Ranger	± 1.5 mm	Advertised equipment error	x 1.2 margin
Center Finding	± 0.05 mm	Machining accuracy	x 1.2 margin

Operational Systems Testing Error

Worst Case Error Scenario

5,000,000 trials

Gaussian randomized error



< 50 km Error Goal

79.9% of the time

< 100 km Error

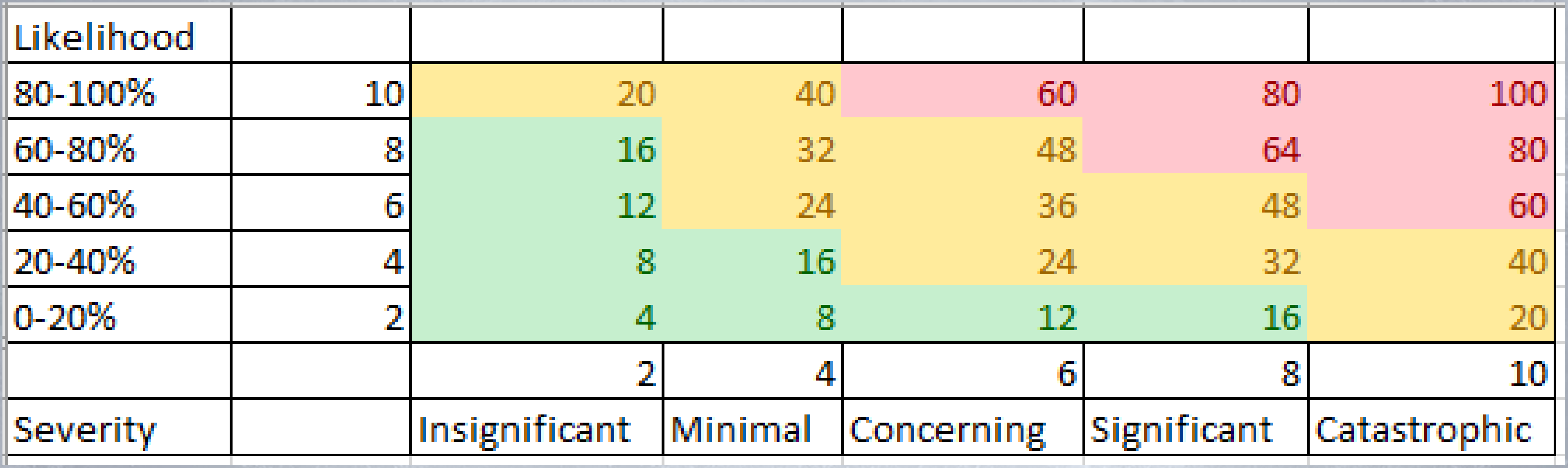
98.31% of the time

Validation & Verification Takeaway

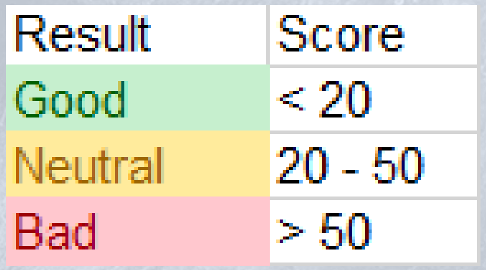
- Test setup currently has an 79.9% chance of providing a statistically significant state error quantification
  - Validates **FR 1.1, 1.2, 1.3, 1.4,** & **1.5**
- Test setup satisfies the customer requirement of developing a system-level testbed
- Sub-system level tests will validate all other sub-system level functional requirements

Risk Analysis

Risk Matrix

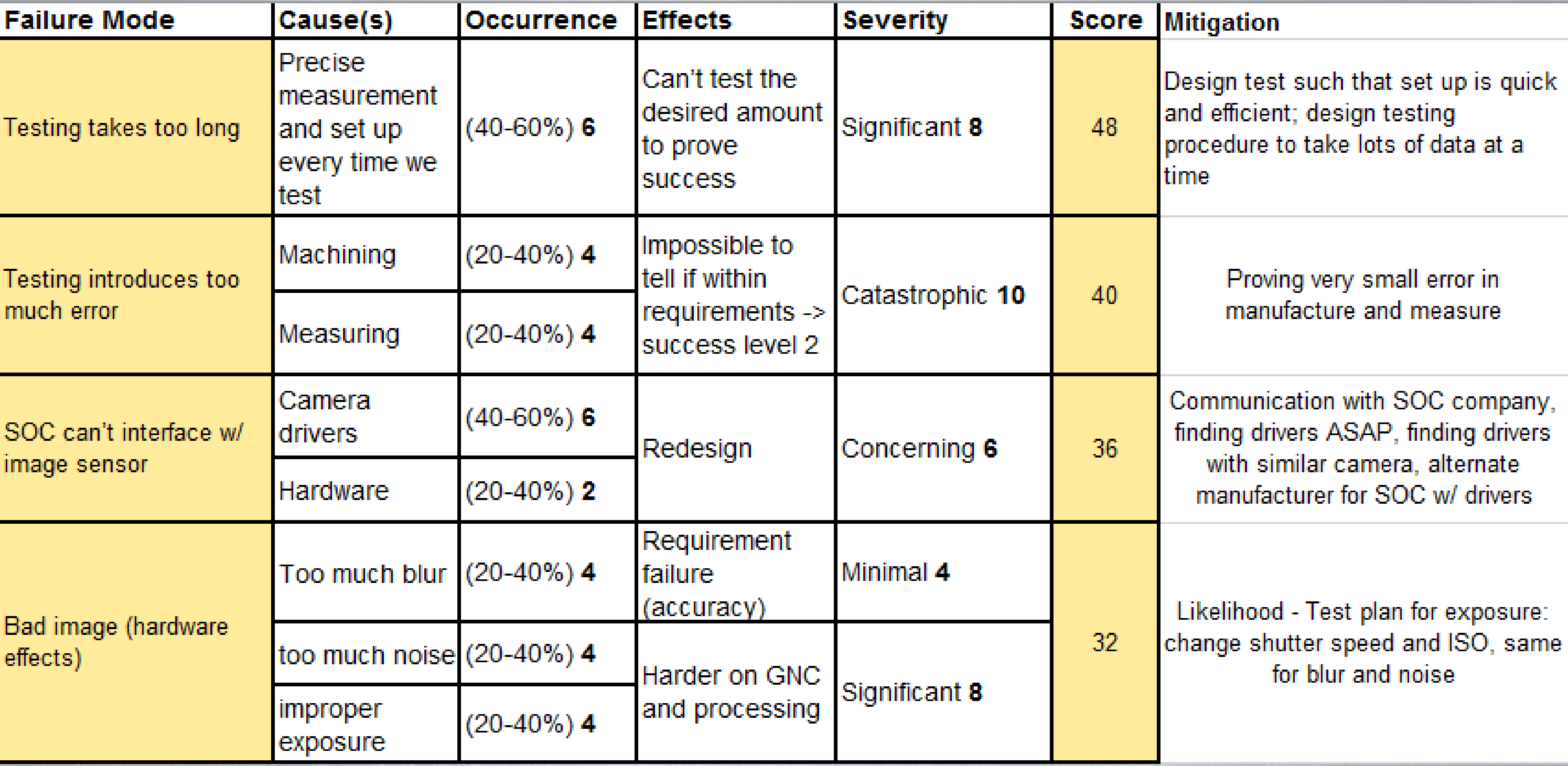


Likelihood						
80-100%	10	20	40	60	80	100
60-80%	8	16	32	48	64	80
40-60%	6	12	24	36	48	60
20-40%	4	8	16	24	32	40
0-20%	2	4	8	12	16	20
		2	4	6	8	10
Severity		Insignificant	Minimal	Concerning	Significant	Catastrophic



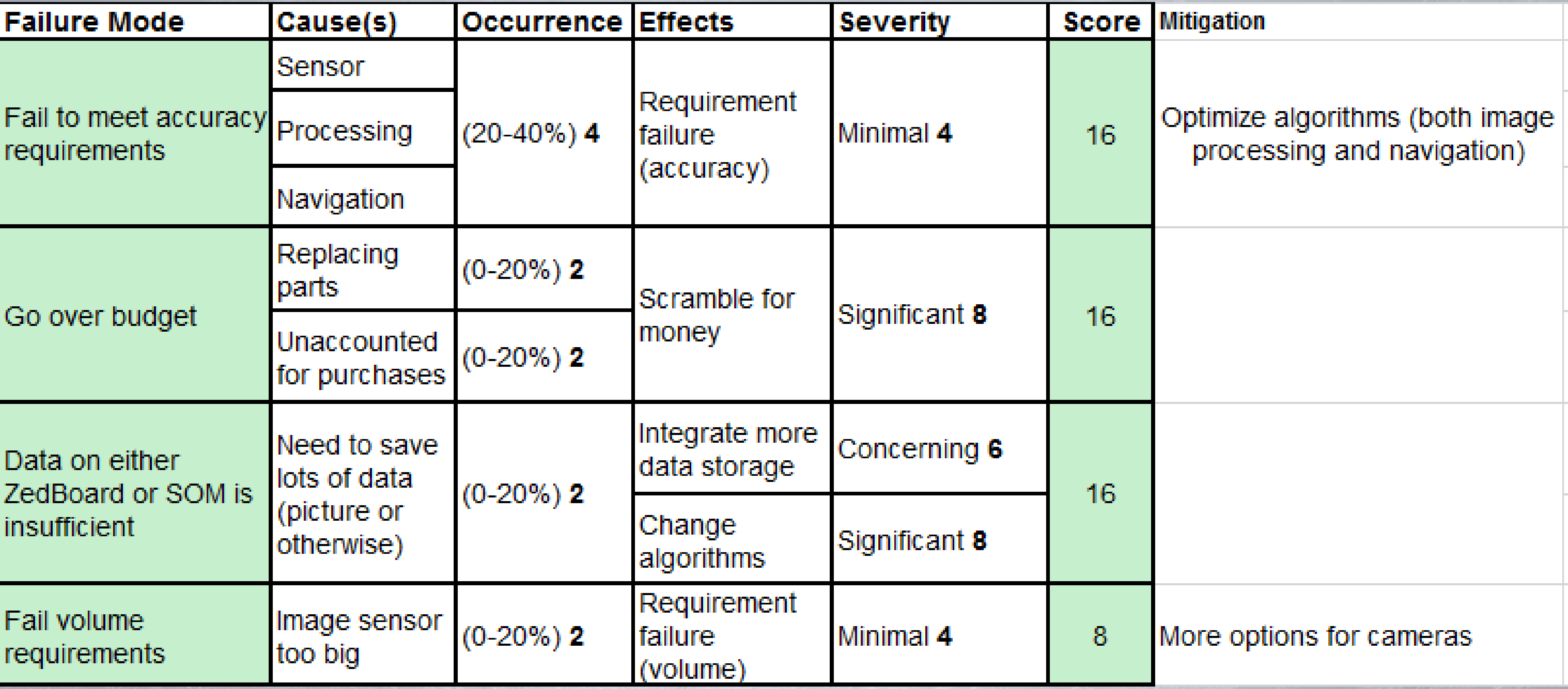
Result	Score
Good	< 20
Neutral	20 - 50
Bad	> 50

Risk Analysis



Failure Mode	**Cause(s)**	**Occurrence**	**Effects**	**Severity**	**Score**	**Mitigation**
Testing takes too long	Precise measurement and set up every time we test	(40-60%) **6**	Can't test the desired amount to prove success	Significant **8**	48	Design test such that set up is quick and efficient; design testing procedure to take lots of data at a time
Testing introduces too much error	Machining	(20-40%) **4**	Impossible to tell if within requirements -> success level 2	Catastrophic **10**	40	Proving very small error in manufacture and measure
	Measuring	(20-40%) **4**				
SOC can't interface w/ image sensor	Camera drivers	(40-60%) **6**	Redesign	Concerning **6**	36	Communication with SOC company, finding drivers ASAP, finding drivers with similar camera, alternate manufacturer for SOC w/ drivers
	Hardware	(20-40%) **2**				
Bad image (hardware effects)	Too much blur	(20-40%) **4**	Requirement failure (accuracy)	Minimal **4**	32	Likelihood - Test plan for exposure: change shutter speed and ISO, same for blur and noise
	too much noise	(20-40%) **4**	Harder on GNC and processing	Significant **8**		
	improper exposure	(20-40%) **4**				

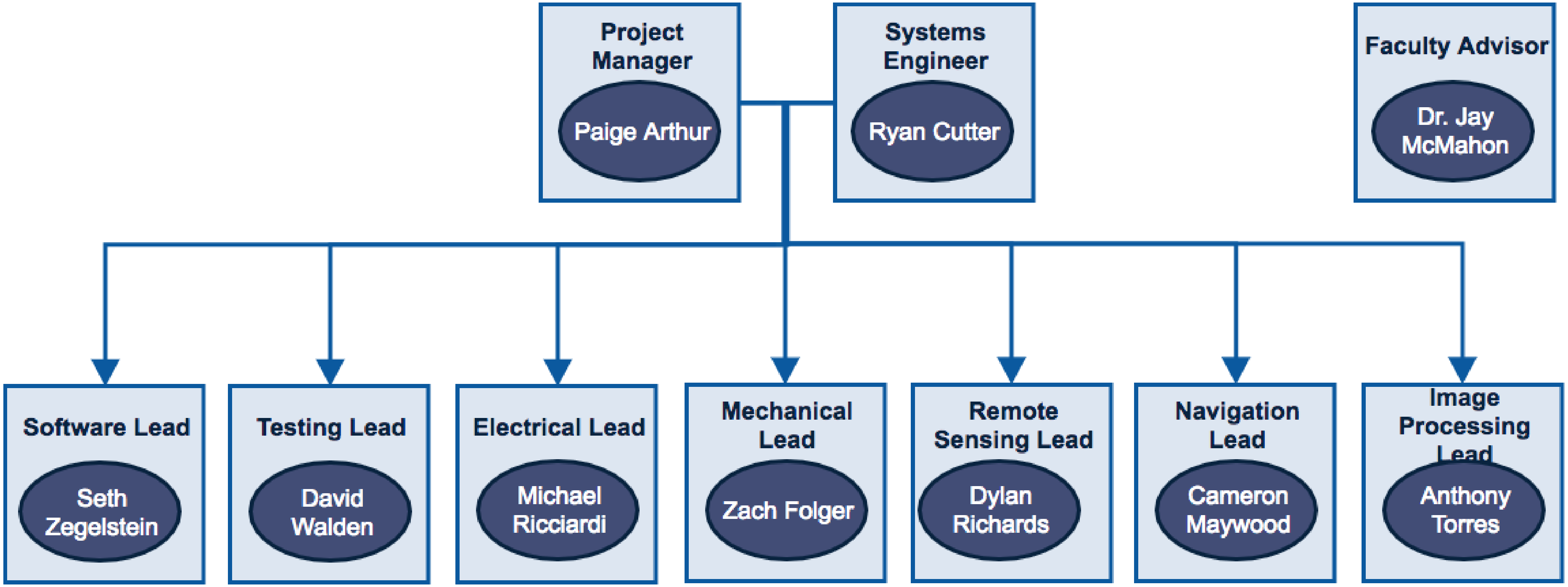
Risk Analysis



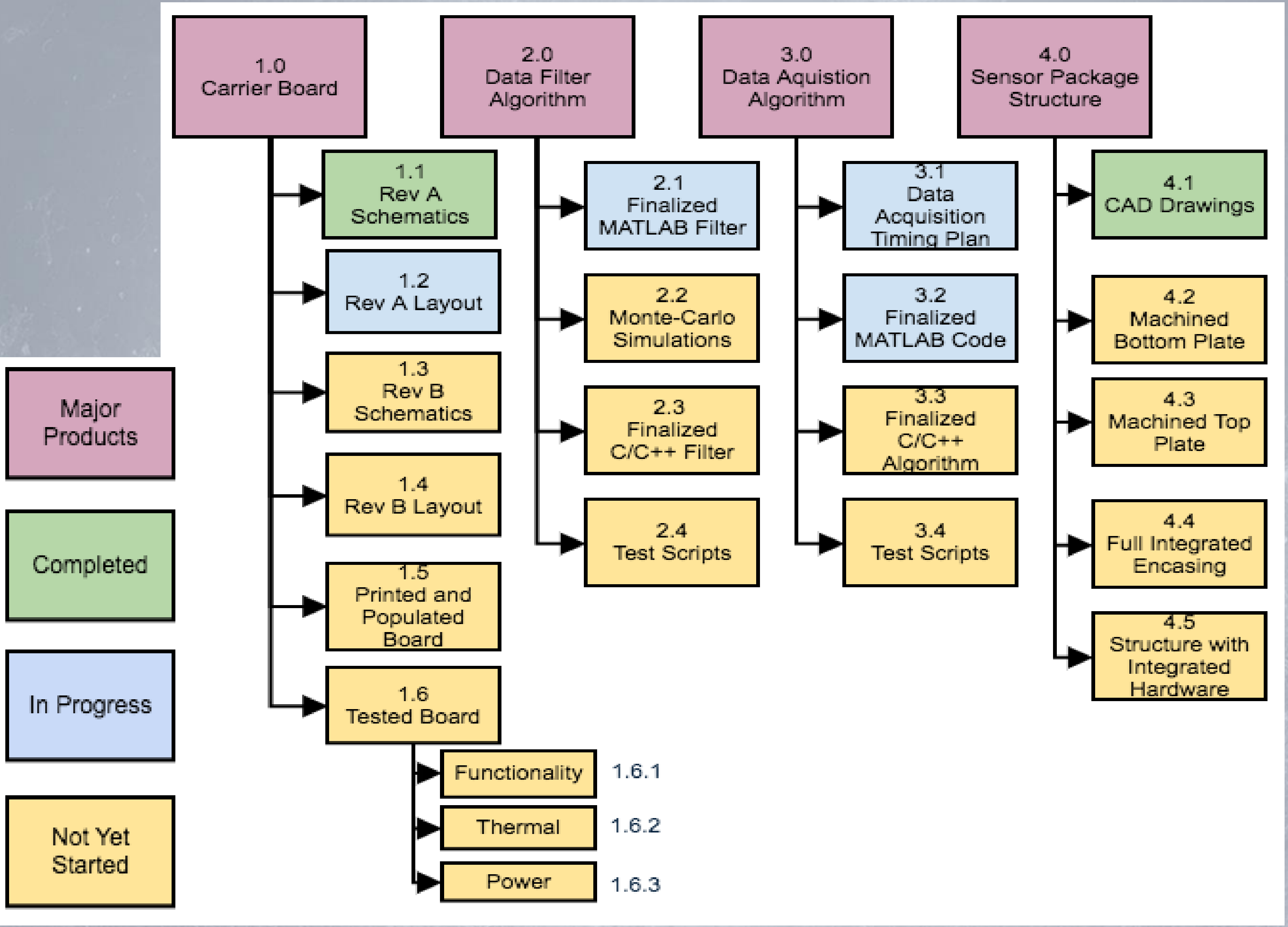
Failure Mode	**Cause(s)**	**Occurrence**	**Effects**	**Severity**	**Score**	**Mitigation**
Fail to meet accuracy requirements	Sensor	(20-40%) **4**	Requirement failure (accuracy)	Minimal **4**	16	Optimize algorithms (both image processing and navigation)
	Processing					
	Navigation					
Go over budget	Replacing parts	(0-20%) **2**	Scramble for money	Significant **8**	16	
	Unaccounted for purchases	(0-20%) **2**				
Data on either ZedBoard or SOM is insufficient	Need to save lots of data (picture or otherwise)	(0-20%) **2**	Integrate more data storage	Concerning **6**	16	
			Change algorithms	Significant **8**		
Fail volume requirements	Image sensor too big	(0-20%) **2**	Requirement failure (volume)	Minimal **4**	8	More options for cameras

Project Planning

Organizational Chart



Work Breakdown Structure



- 1.0 Carrier Board
  - 1.1 Rev A Schematics
  - 1.2 Rev A Layout
  - 1.3 Rev B Schematics
  - 1.4 Rev B Layout
  - 1.5 Printed and Populated Board
  - 1.6 Tested Board
    - Functionality 1.6.1
    - Thermal 1.6.2
    - Power 1.6.3
- 2.0 Data Filter Algorithm
  - 2.1 Finalized MATLAB Filter
  - 2.2 Monte-Carlo Simulations
  - 2.3 Finalized C/C++ Filter
  - 2.4 Test Scripts
- 3.0 Data Aquistion Algorithm
  - 3.1 Data Acquisition Timing Plan
  - 3.2 Finalized MATLAB Code
  - 3.3 Finalized C/C++ Algorithm
  - 3.4 Test Scripts
- 4.0 Sensor Package Structure
  - 4.1 CAD Drawings
  - 4.2 Machined Bottom Plate
  - 4.3 Machined Top Plate
  - 4.4 Full Integrated Encasing
  - 4.5 Structure with Integrated Hardware

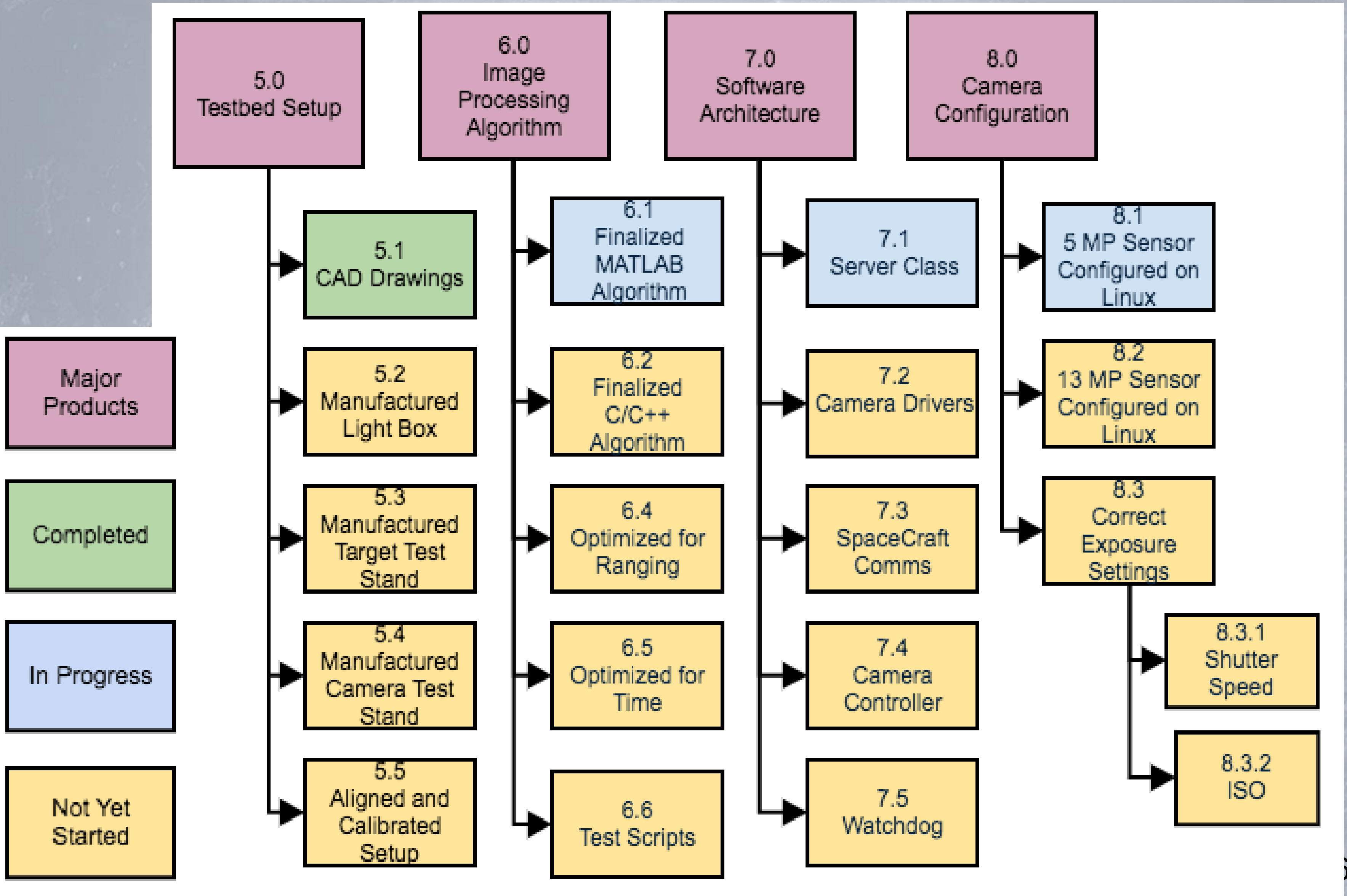
Major Products

Completed

In Progress

Not Yet Started

Work Breakdown Structure



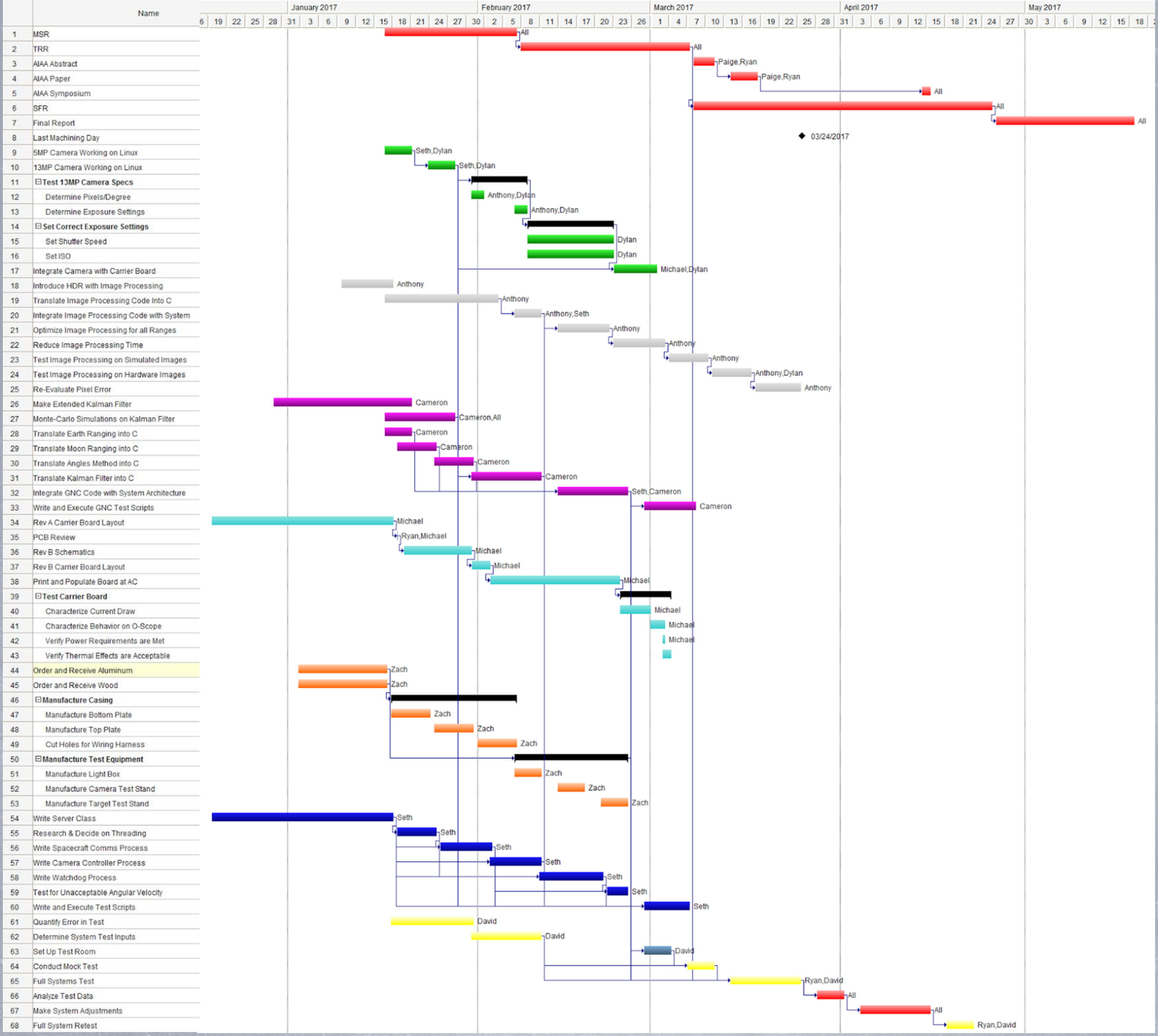
- 5.0 Testbed Setup
  - 5.1 CAD Drawings
  - 5.2 Manufactured Light Box
  - 5.3 Manufactured Target Test Stand
  - 5.4 Manufactured Camera Test Stand
  - 5.5 Aligned and Calibrated Setup
- 6.0 Image Processing Algorithm
  - 6.1 Finalized MATLAB Algorithm
  - 6.2 Finalized C/C++ Algorithm
  - 6.4 Optimized for Ranging
  - 6.5 Optimized for Time
  - 6.6 Test Scripts
- 7.0 Software Architecture
  - 7.1 Server Class
  - 7.2 Camera Drivers
  - 7.3 SpaceCraft Comms
  - 7.4 Camera Controller
  - 7.5 Watchdog
- 8.0 Camera Configuration
  - 8.1 5 MP Sensor Configured on Linux
  - 8.2 13 MP Sensor Configured on Linux
  - 8.3 Correct Exposure Settings
    - 8.3.1 Shutter Speed
    - 8.3.2 ISO

Major Products

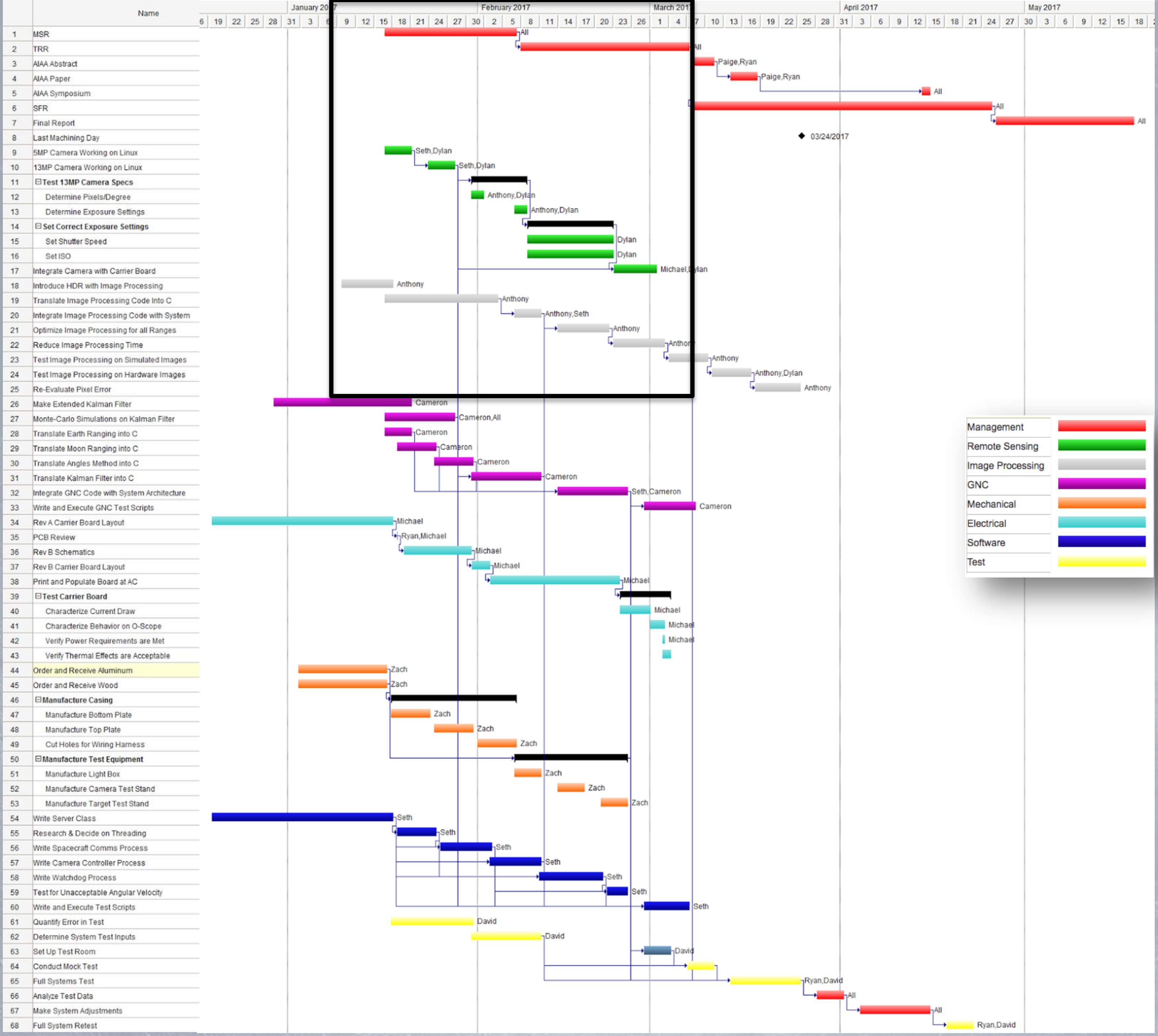
Completed

In Progress

Not Yet Started

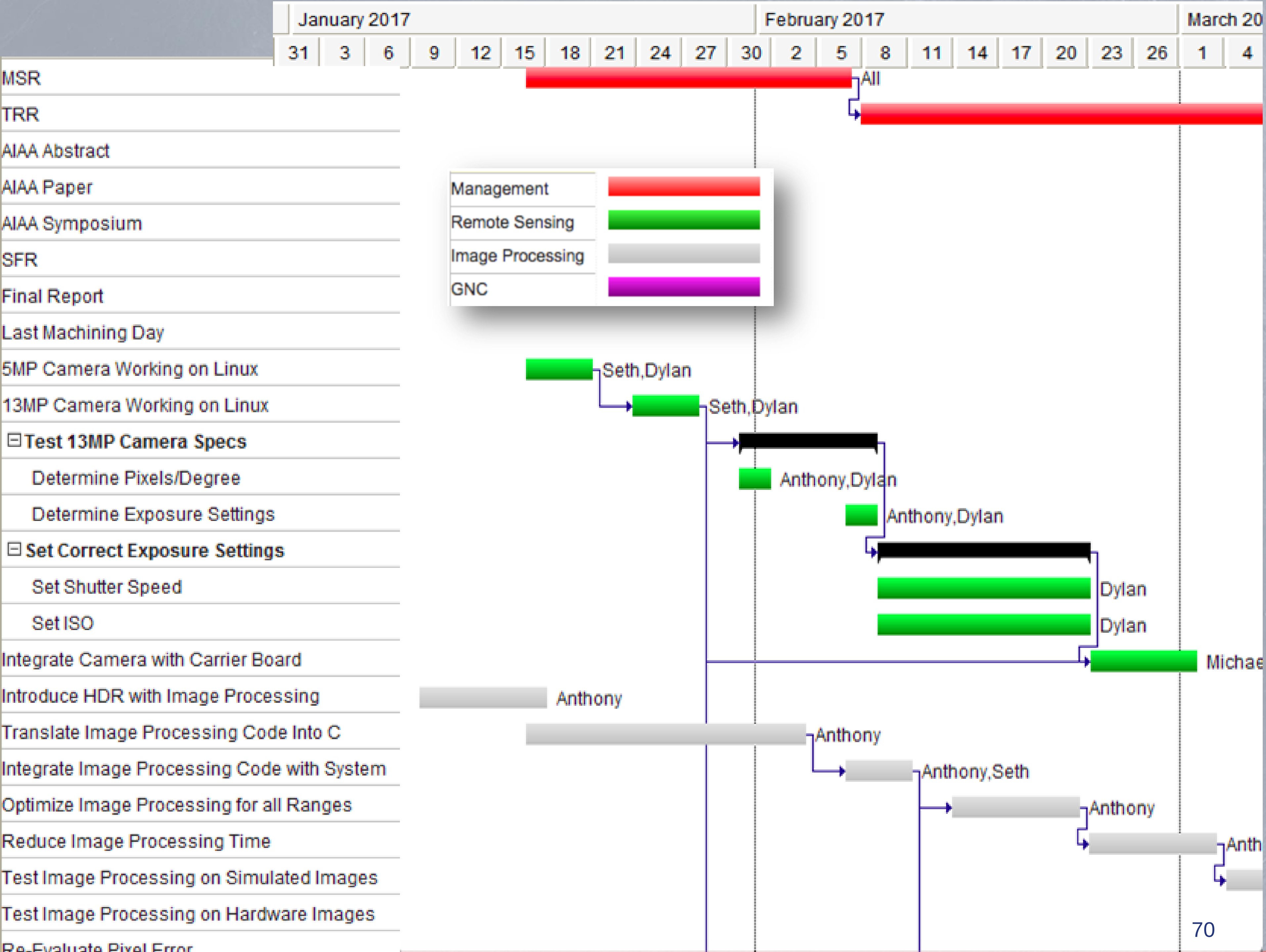


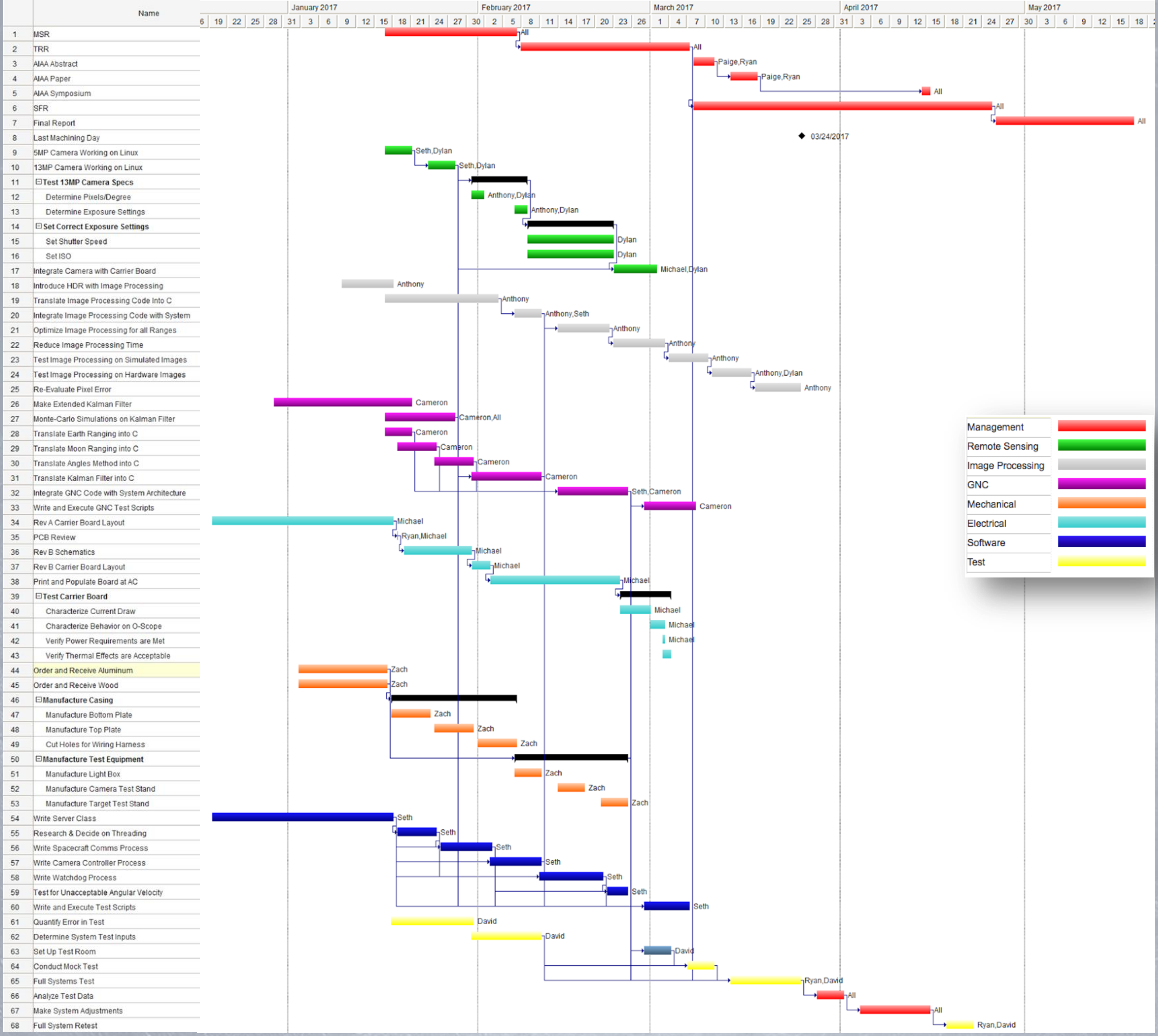
	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest



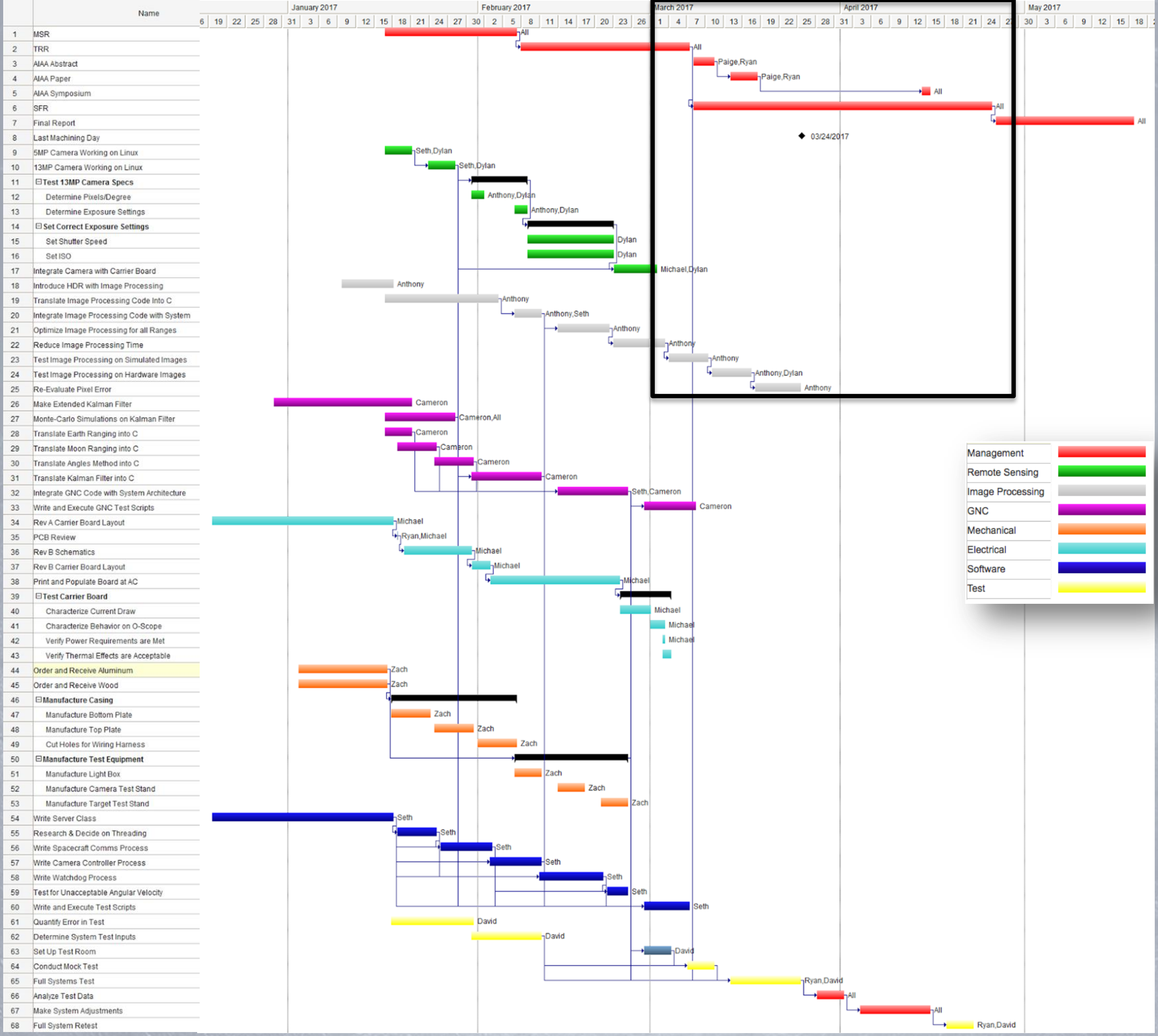
	Name	
1	MSR	All
2	TRR	All
3	AIAA Abstract	Paige,Ryan
4	AIAA Paper	Paige,Ryan
5	AIAA Symposium	All
6	SFR	All
7	Final Report	All
8	Last Machining Day	03/24/2017
9	5MP Camera Working on Linux	Seth,Dylan
10	13MP Camera Working on Linux	Seth,Dylan
11	⊟ **Test 13MP Camera Specs**	
12	Determine Pixels/Degree	Anthony,Dylan
13	Determine Exposure Settings	Anthony,Dylan
14	⊟ **Set Correct Exposure Settings**	
15	Set Shutter Speed	Dylan
16	Set ISO	Dylan
17	Integrate Camera with Carrier Board	Michael,Dylan
18	Introduce HDR with Image Processing	Anthony
19	Translate Image Processing Code Into C	Anthony
20	Integrate Image Processing Code with System	Anthony,Seth
21	Optimize Image Processing for all Ranges	Anthony
22	Reduce Image Processing Time	Antho
23	Test Image Processing on Simulated Images	Anthony
24	Test Image Processing on Hardware Images	Anthony,Dylan
25	Re-Evaluate Pixel Error	Anthony
26	Make Extended Kalman Filter	Cameron
27	Monte-Carlo Simulations on Kalman Filter	Cameron,All
28	Translate Earth Ranging into C	Cameron
29	Translate Moon Ranging into C	Cameron
30	Translate Angles Method into C	Cameron
31	Translate Kalman Filter into C	Cameron
32	Integrate GNC Code with System Architecture	Seth,Cameron
33	Write and Execute GNC Test Scripts	Cameron
34	Rev A Carrier Board Layout	Michael
35	PCB Review	Ryan,Michael
36	Rev B Schematics	Michael
37	Rev B Carrier Board Layout	Michael
38	Print and Populate Board at AC	Michael
39	⊟ **Test Carrier Board**	
40	Characterize Current Draw	Michael
41	Characterize Behavior on O-Scope	Michael
42	Verify Power Requirements are Met	Michael
43	Verify Thermal Effects are Acceptable	
44	Order and Receive Aluminum	Zach
45	Order and Receive Wood	Zach
46	⊟ **Manufacture Casing**	
47	Manufacture Bottom Plate	Zach
48	Manufacture Top Plate	Zach
49	Cut Holes for Wiring Harness	Zach
50	⊟ **Manufacture Test Equipment**	
51	Manufacture Light Box	Zach
52	Manufacture Camera Test Stand	Zach
53	Manufacture Target Test Stand	Zach
54	Write Server Class	Seth
55	Research & Decide on Threading	Seth
56	Write Spacecraft Comms Process	Seth
57	Write Camera Controller Process	Seth
58	Write Watchdog Process	Seth
59	Test for Unacceptable Angular Velocity	Seth
60	Write and Execute Test Scripts	Seth
61	Quantify Error in Test	David
62	Determine System Test Inputs	David
63	Set Up Test Room	David
64	Conduct Mock Test	
65	Full Systems Test	Ryan,David
66	Analyze Test Data	All
67	Make System Adjustments	All
68	Full System Retest	Ryan,David

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨


January 2017
February 2017
March 20
31
3
6
9
12
15
18
21
24
27
30
2
5
8
11
14
17
20
23
26
1
4
MSR
TRR
AIAA Abstract
AIAA Paper
AIAA Symposium
SFR
Final Report
Last Machining Day
5MP Camera Working on Linux
13MP Camera Working on Linux
⊟ Test 13MP Camera Specs
Determine Pixels/Degree
Determine Exposure Settings
⊟ Set Correct Exposure Settings
Set Shutter Speed
Set ISO
Integrate Camera with Carrier Board
Introduce HDR with Image Processing
Translate Image Processing Code Into C
Integrate Image Processing Code with System
Optimize Image Processing for all Ranges
Reduce Image Processing Time
Test Image Processing on Simulated Images
Test Image Processing on Hardware Images
Re-Evaluate Pixel Error
All
Management 🟥
Remote Sensing 🟩
Image Processing ⬜
GNC 🟪
Seth,Dylan
Seth,Dylan
Anthony,Dylan
Anthony,Dylan
Dylan
Dylan
Michae
Anthony
Anthony
Anthony,Seth
Anthony
Anth

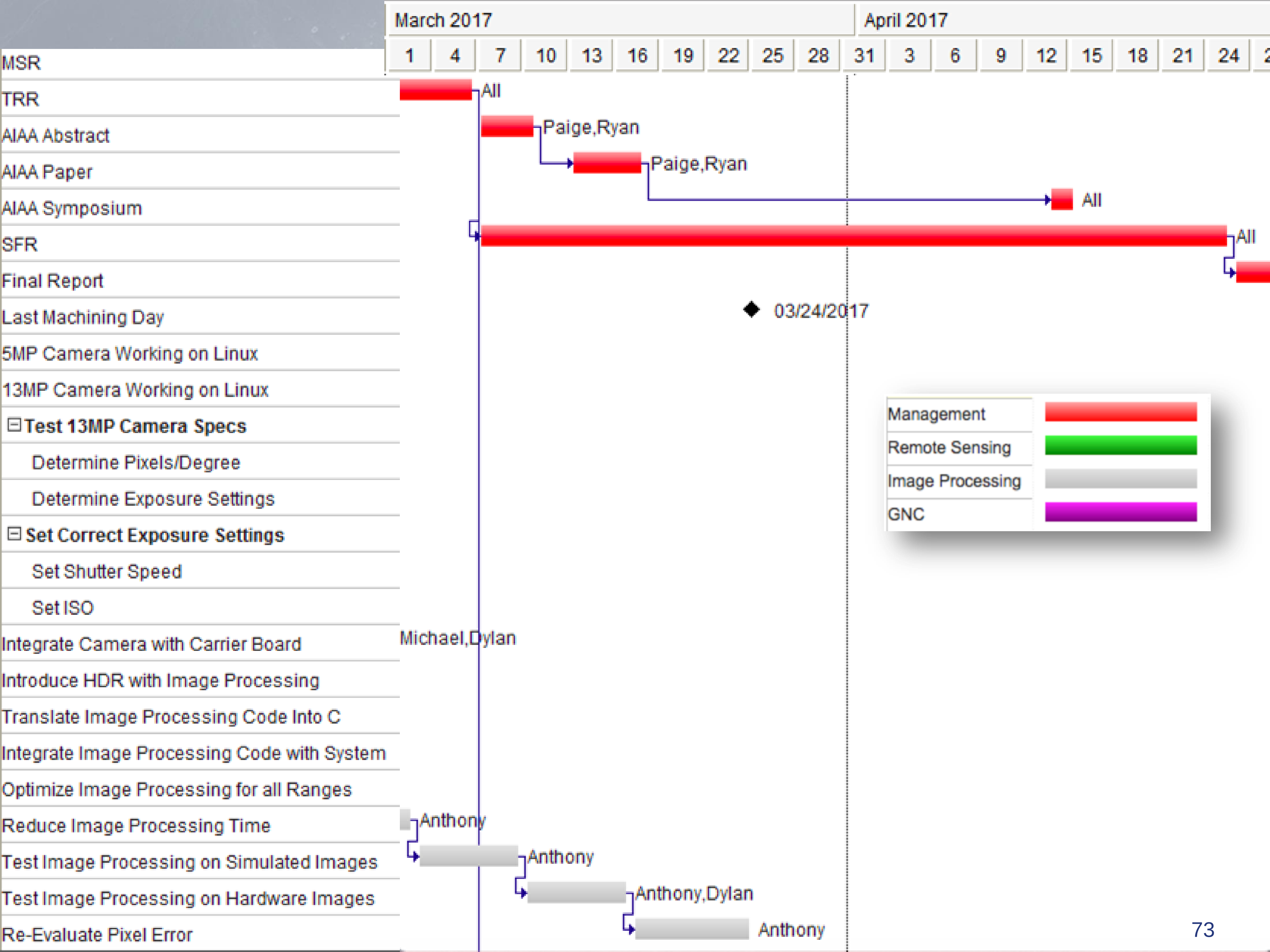


	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟Test 13MP Camera Specs
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟Set Correct Exposure Settings
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟Test Carrier Board
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟Manufacture Casing
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟Manufacture Test Equipment
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest



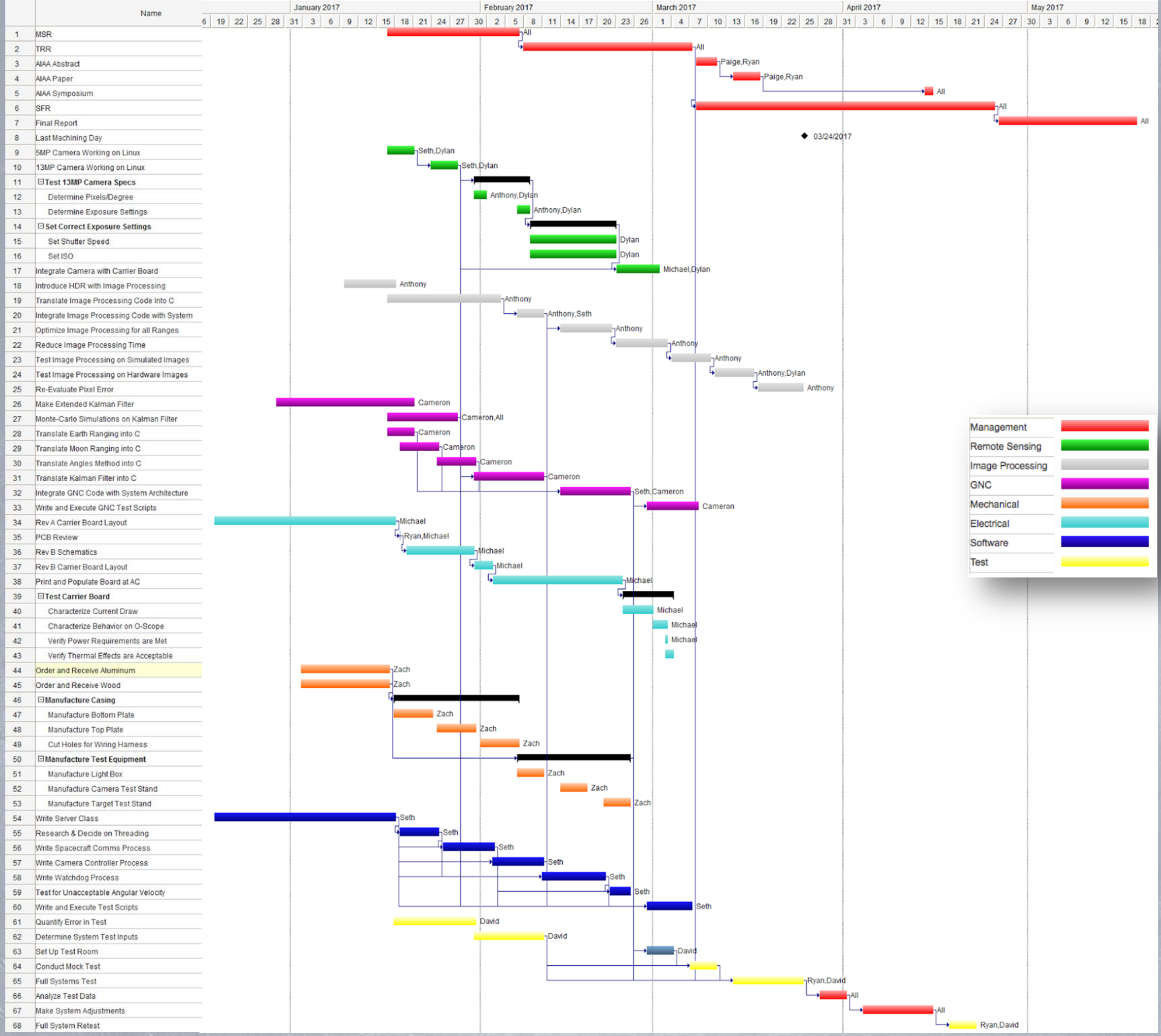
	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨



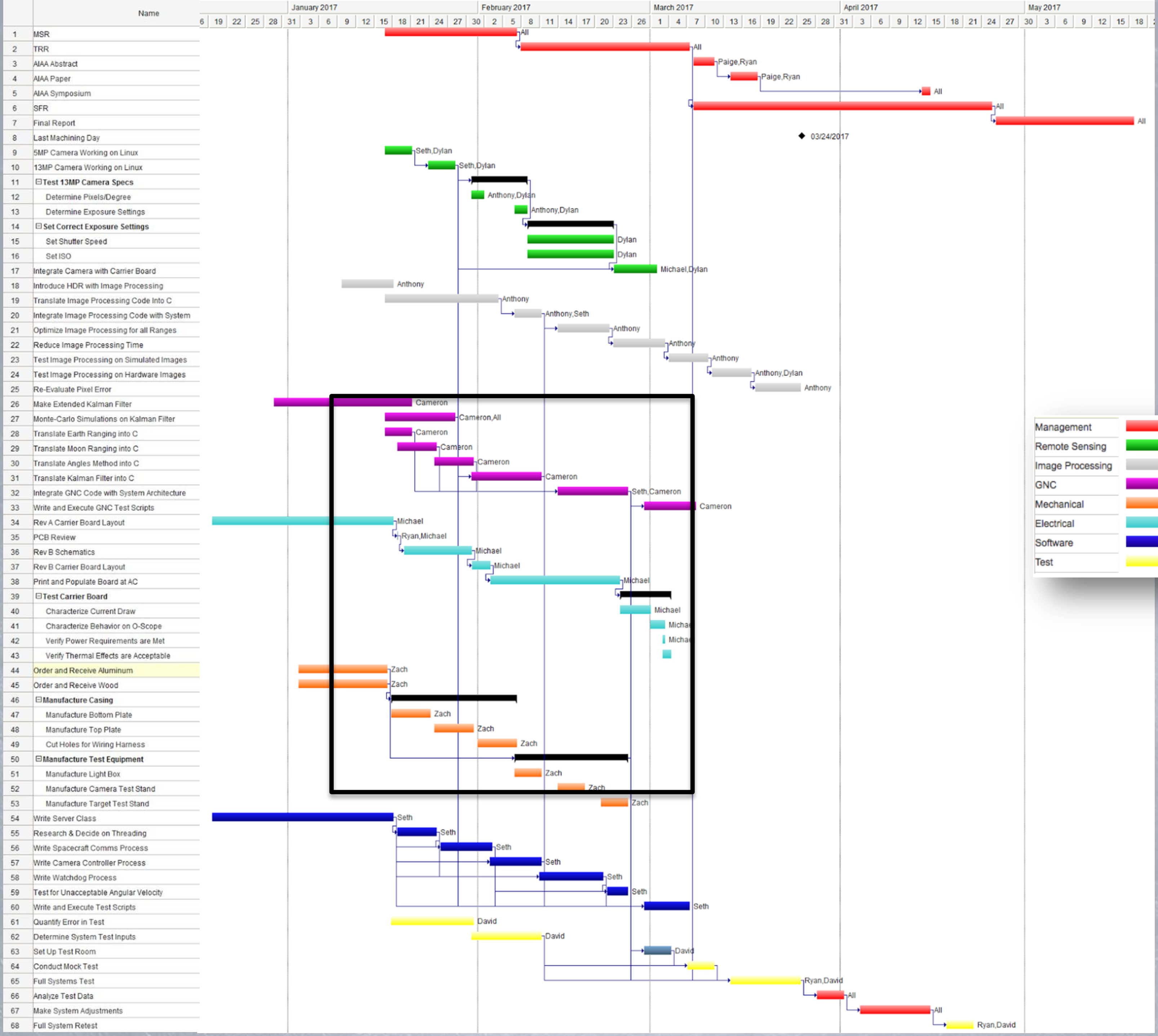
MSR
TRR
AIAA Abstract
AIAA Paper
AIAA Symposium
SFR
Final Report
Last Machining Day
5MP Camera Working on Linux
13MP Camera Working on Linux
⊟ **Test 13MP Camera Specs**
  Determine Pixels/Degree
  Determine Exposure Settings
⊟ **Set Correct Exposure Settings**
  Set Shutter Speed
  Set ISO
Integrate Camera with Carrier Board
Introduce HDR with Image Processing
Translate Image Processing Code Into C
Integrate Image Processing Code with System
Optimize Image Processing for all Ranges
Reduce Image Processing Time
Test Image Processing on Simulated Images
Test Image Processing on Hardware Images
Re-Evaluate Pixel Error

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪



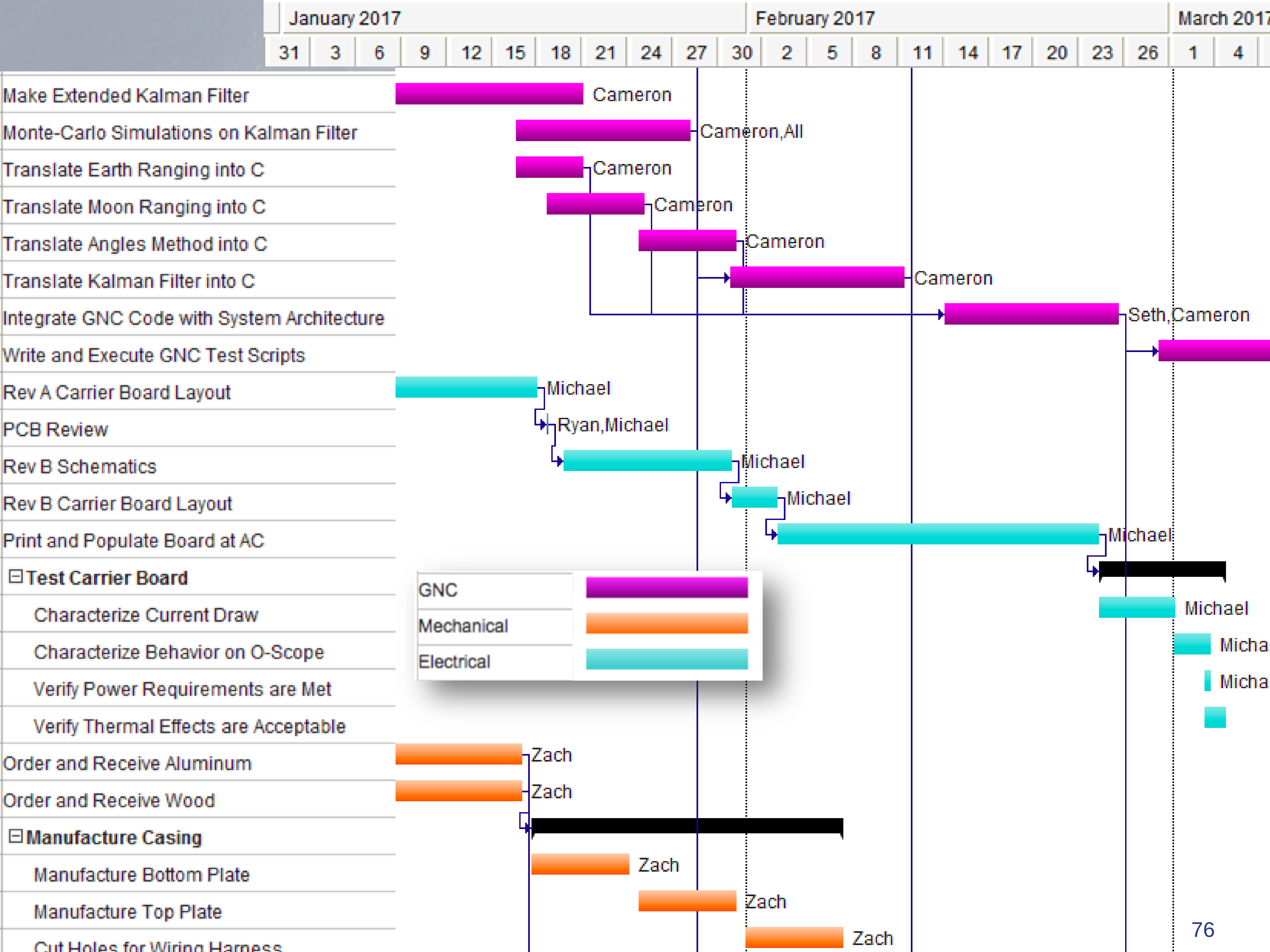
	Name	
1	MSR	All
2	TRR	All
3	AIAA Abstract	Paige,Ryan
4	AIAA Paper	Paige,Ryan
5	AIAA Symposium	All
6	SFR	All
7	Final Report	All
8	Last Machining Day	03/24/2017
9	5MP Camera Working on Linux	Seth,Dylan
10	13MP Camera Working on Linux	Seth,Dylan
11	⊟ **Test 13MP Camera Specs**	
12	Determine Pixels/Degree	Anthony,Dylan
13	Determine Exposure Settings	Anthony,Dylan
14	⊟ **Set Correct Exposure Settings**	
15	Set Shutter Speed	Dylan
16	Set ISO	Dylan
17	Integrate Camera with Carrier Board	Michael,Dylan
18	Introduce HDR with Image Processing	Anthony
19	Translate Image Processing Code Into C	Anthony
20	Integrate Image Processing Code with System	Anthony,Seth
21	Optimize Image Processing for all Ranges	Anthony
22	Reduce Image Processing Time	Anthony
23	Test Image Processing on Simulated Images	Anthony
24	Test Image Processing on Hardware Images	Anthony,Dylan
25	Re-Evaluate Pixel Error	Anthony
26	Make Extended Kalman Filter	Cameron
27	Monte-Carlo Simulations on Kalman Filter	Cameron,All
28	Translate Earth Ranging into C	Cameron
29	Translate Moon Ranging into C	Cameron
30	Translate Angles Method into C	Cameron
31	Translate Kalman Filter into C	Cameron
32	Integrate GNC Code with System Architecture	Seth,Cameron
33	Write and Execute GNC Test Scripts	Cameron
34	Rev A Carrier Board Layout	Michael
35	PCB Review	Ryan,Michael
36	Rev B Schematics	Michael
37	Rev B Carrier Board Layout	Michael
38	Print and Populate Board at AC	Michael
39	⊟ **Test Carrier Board**	
40	Characterize Current Draw	Michael
41	Characterize Behavior on O-Scope	Michael
42	Verify Power Requirements are Met	Michael
43	Verify Thermal Effects are Acceptable	
44	Order and Receive Aluminum	Zach
45	Order and Receive Wood	Zach
46	⊟ **Manufacture Casing**	
47	Manufacture Bottom Plate	Zach
48	Manufacture Top Plate	Zach
49	Cut Holes for Wiring Harness	Zach
50	⊟ **Manufacture Test Equipment**	
51	Manufacture Light Box	Zach
52	Manufacture Camera Test Stand	Zach
53	Manufacture Target Test Stand	Zach
54	Write Server Class	Seth
55	Research & Decide on Threading	Seth
56	Write Spacecraft Comms Process	Seth
57	Write Camera Controller Process	Seth
58	Write Watchdog Process	Seth
59	Test for Unacceptable Angular Velocity	Seth
60	Write and Execute Test Scripts	Seth
61	Quantify Error in Test	David
62	Determine System Test Inputs	David
63	Set Up Test Room	David
64	Conduct Mock Test	
65	Full Systems Test	Ryan,David
66	Analyze Test Data	All
67	Make System Adjustments	All
68	Full System Retest	Ryan,David

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨

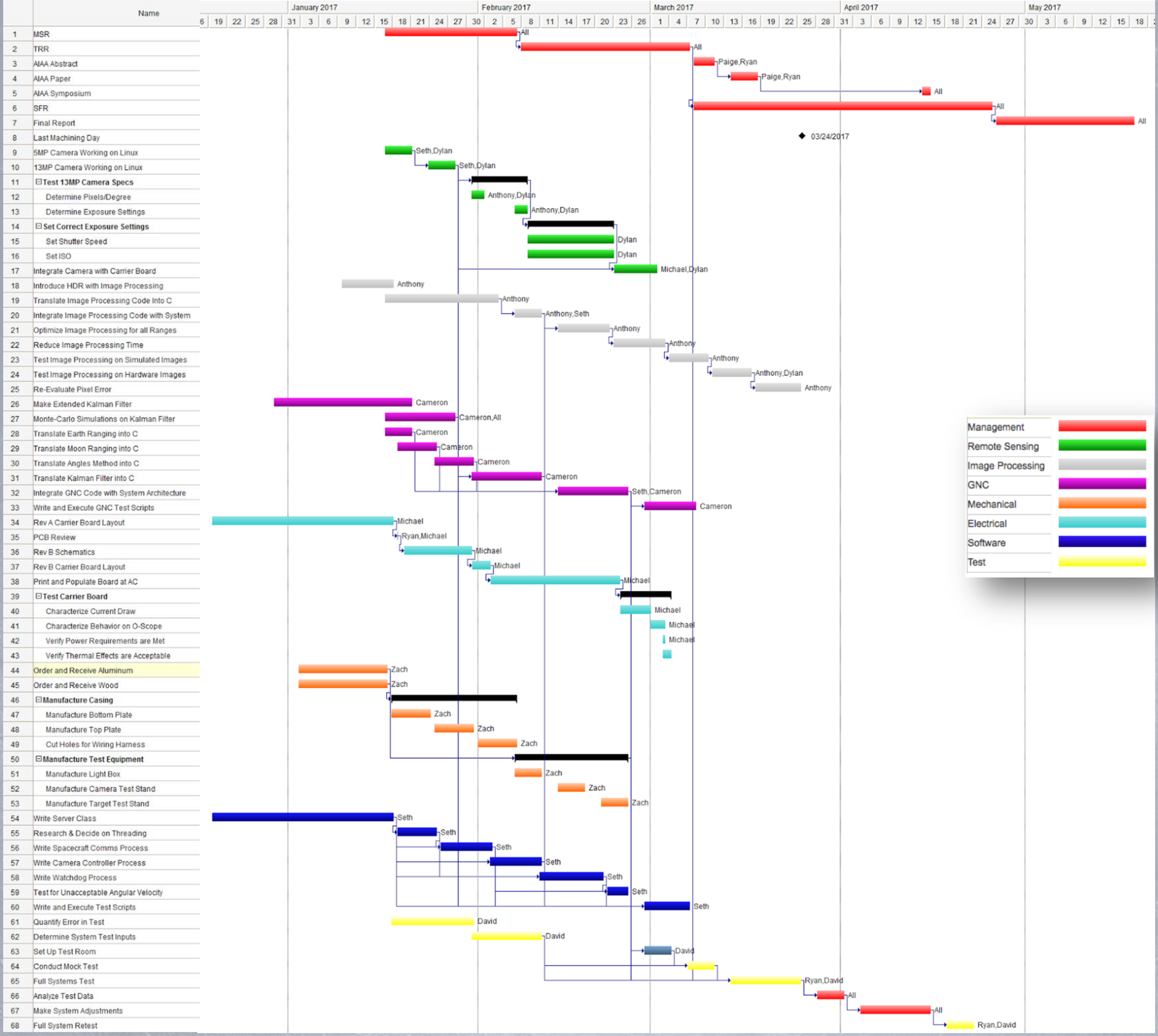


	Name	
1	MSR	All
2	TRR	All
3	AIAA Abstract	Paige,Ryan
4	AIAA Paper	Paige,Ryan
5	AIAA Symposium	All
6	SFR	All
7	Final Report	All
8	Last Machining Day	03/24/2017
9	5MP Camera Working on Linux	Seth,Dylan
10	13MP Camera Working on Linux	Seth,Dylan
11	**⊟Test 13MP Camera Specs**	
12	Determine Pixels/Degree	Anthony,Dylan
13	Determine Exposure Settings	Anthony,Dylan
14	**⊟Set Correct Exposure Settings**	
15	Set Shutter Speed	Dylan
16	Set ISO	Dylan
17	Integrate Camera with Carrier Board	Michael,Dylan
18	Introduce HDR with Image Processing	Anthony
19	Translate Image Processing Code Into C	Anthony
20	Integrate Image Processing Code with System	Anthony,Seth
21	Optimize Image Processing for all Ranges	Anthony
22	Reduce Image Processing Time	Anthony
23	Test Image Processing on Simulated Images	Anthony
24	Test Image Processing on Hardware Images	Anthony,Dylan
25	Re-Evaluate Pixel Error	Anthony
26	Make Extended Kalman Filter	Cameron
27	Monte-Carlo Simulations on Kalman Filter	Cameron,All
28	Translate Earth Ranging into C	Cameron
29	Translate Moon Ranging into C	Cameron
30	Translate Angles Method into C	Cameron
31	Translate Kalman Filter into C	Cameron
32	Integrate GNC Code with System Architecture	Seth,Cameron
33	Write and Execute GNC Test Scripts	Cameron
34	Rev A Carrier Board Layout	Michael
35	PCB Review	Ryan,Michael
36	Rev B Schematics	Michael
37	Rev B Carrier Board Layout	Michael
38	Print and Populate Board at AC	Michael
39	**⊟Test Carrier Board**	
40	Characterize Current Draw	Michael
41	Characterize Behavior on O-Scope	Micha
42	Verify Power Requirements are Met	Micha
43	Verify Thermal Effects are Acceptable	
44	Order and Receive Aluminum	Zach
45	Order and Receive Wood	Zach
46	**⊟Manufacture Casing**	
47	Manufacture Bottom Plate	Zach
48	Manufacture Top Plate	Zach
49	Cut Holes for Wiring Harness	Zach
50	**⊟Manufacture Test Equipment**	
51	Manufacture Light Box	Zach
52	Manufacture Camera Test Stand	Zach
53	Manufacture Target Test Stand	Zach
54	Write Server Class	Seth
55	Research & Decide on Threading	Seth
56	Write Spacecraft Comms Process	Seth
57	Write Camera Controller Process	Seth
58	Write Watchdog Process	Seth
59	Test for Unacceptable Angular Velocity	Seth
60	Write and Execute Test Scripts	Seth
61	Quantify Error in Test	David
62	Determine System Test Inputs	David
63	Set Up Test Room	David
64	Conduct Mock Test	
65	Full Systems Test	Ryan,David
66	Analyze Test Data	All
67	Make System Adjustments	All
68	Full System Retest	Ryan,David

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨

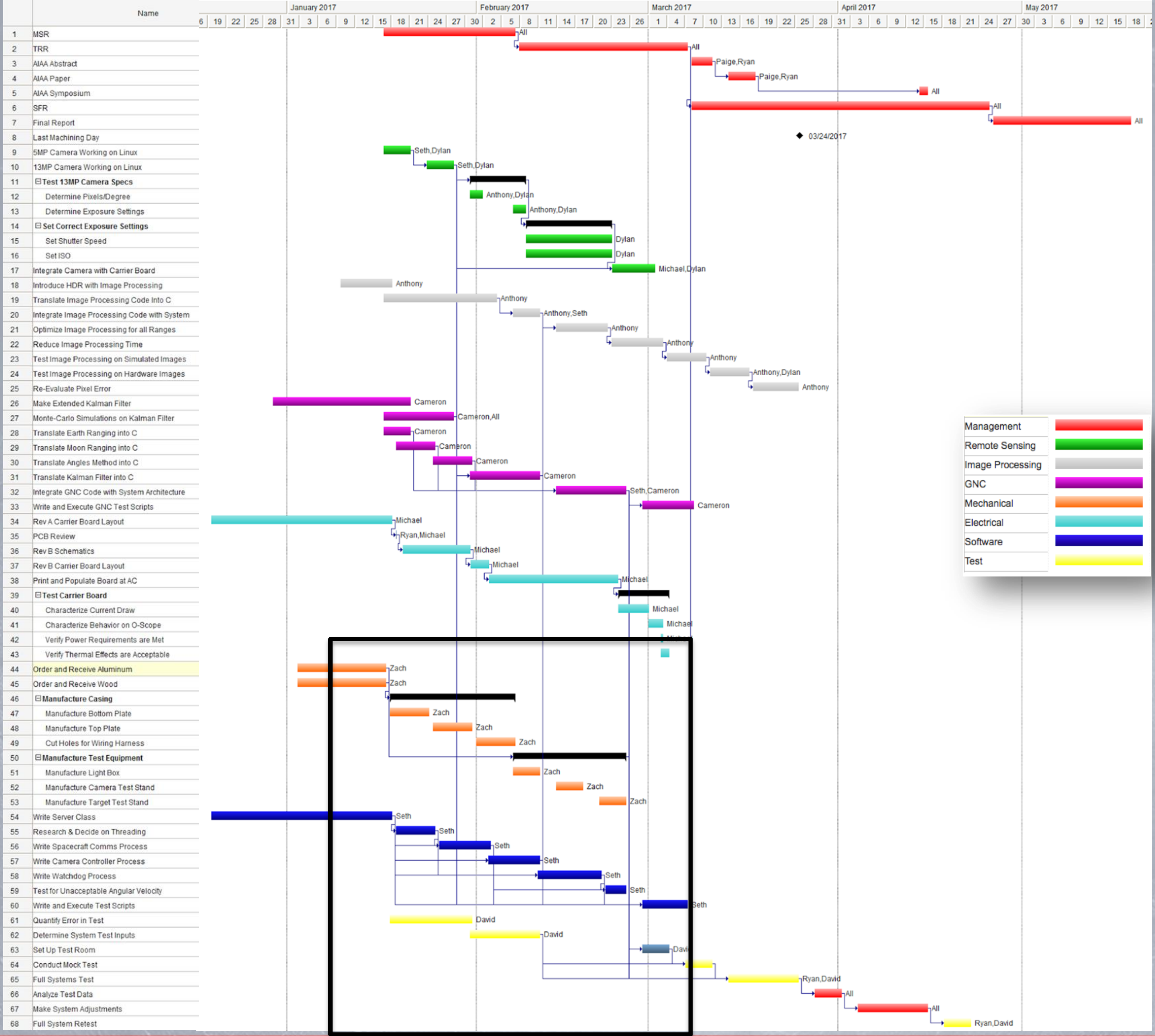


Make Extended Kalman Filter
Monte-Carlo Simulations on Kalman Filter
Translate Earth Ranging into C
Translate Moon Ranging into C
Translate Angles Method into C
Translate Kalman Filter into C
Integrate GNC Code with System Architecture
Write and Execute GNC Test Scripts
Rev A Carrier Board Layout
PCB Review
Rev B Schematics
Rev B Carrier Board Layout
Print and Populate Board at AC
⊟ **Test Carrier Board**
  Characterize Current Draw
  Characterize Behavior on O-Scope
  Verify Power Requirements are Met
  Verify Thermal Effects are Acceptable
Order and Receive Aluminum
Order and Receive Wood
⊟ **Manufacture Casing**
  Manufacture Bottom Plate
  Manufacture Top Plate
  Cut Holes for Wiring Harness



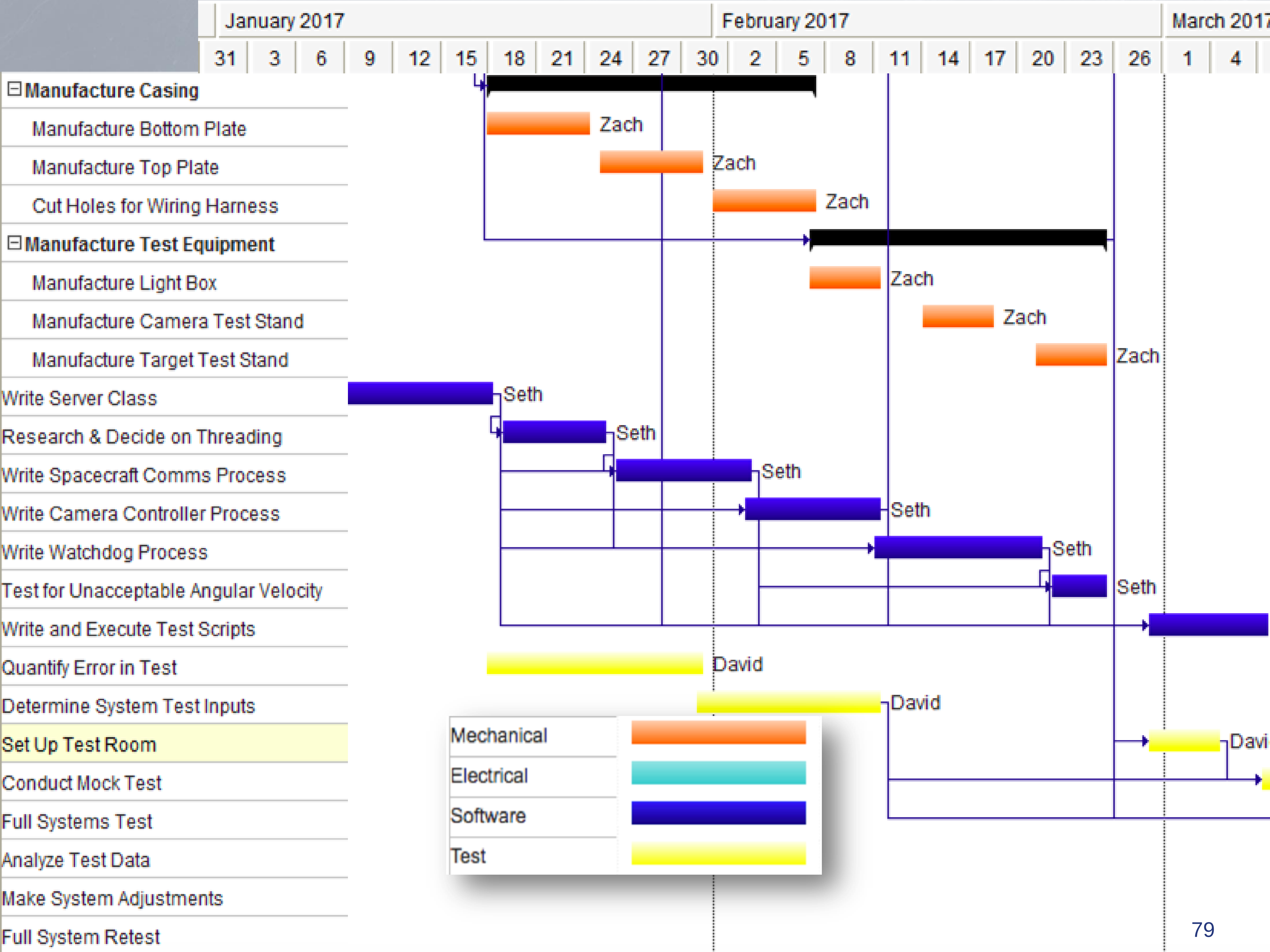
	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨

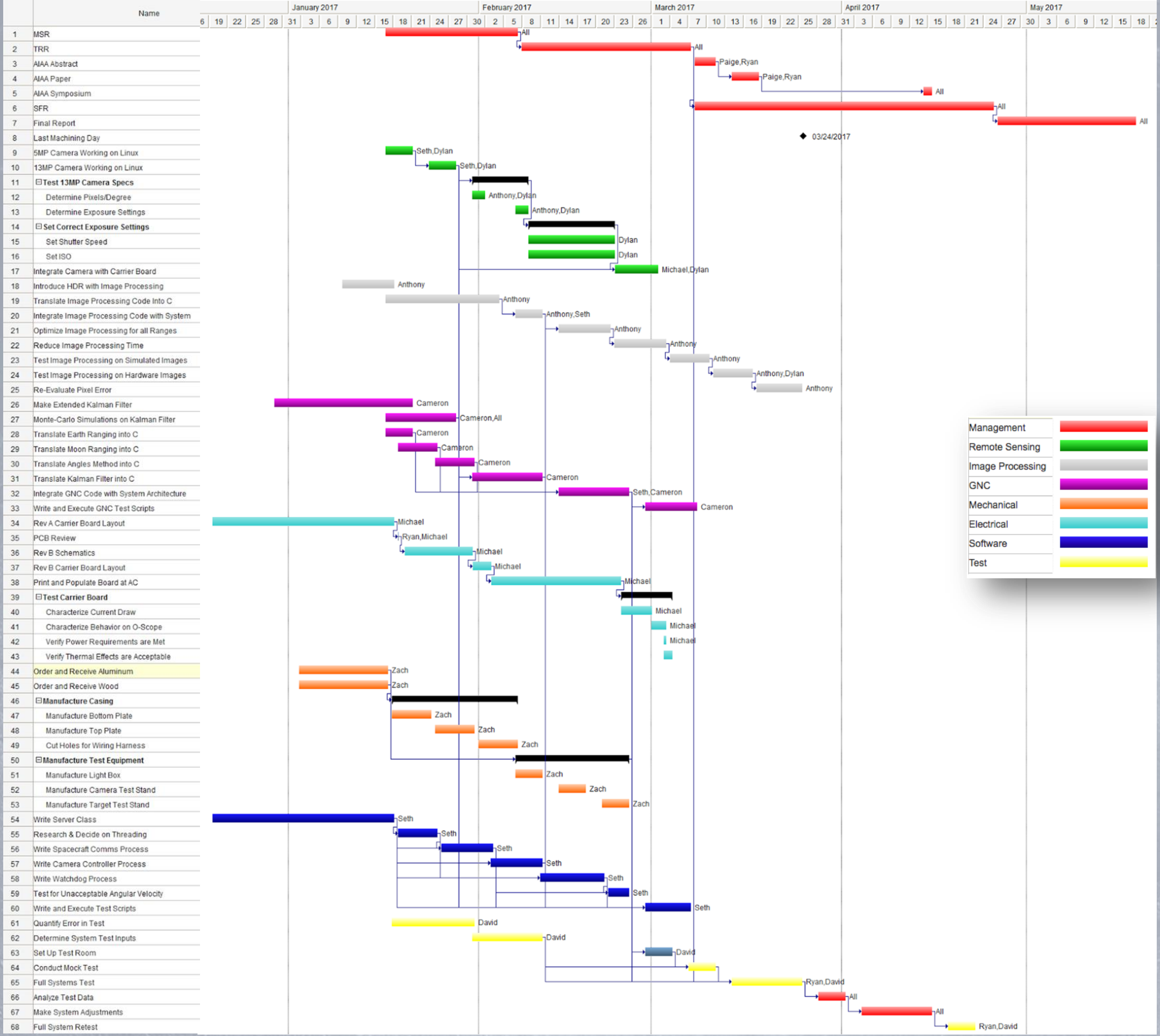


	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨

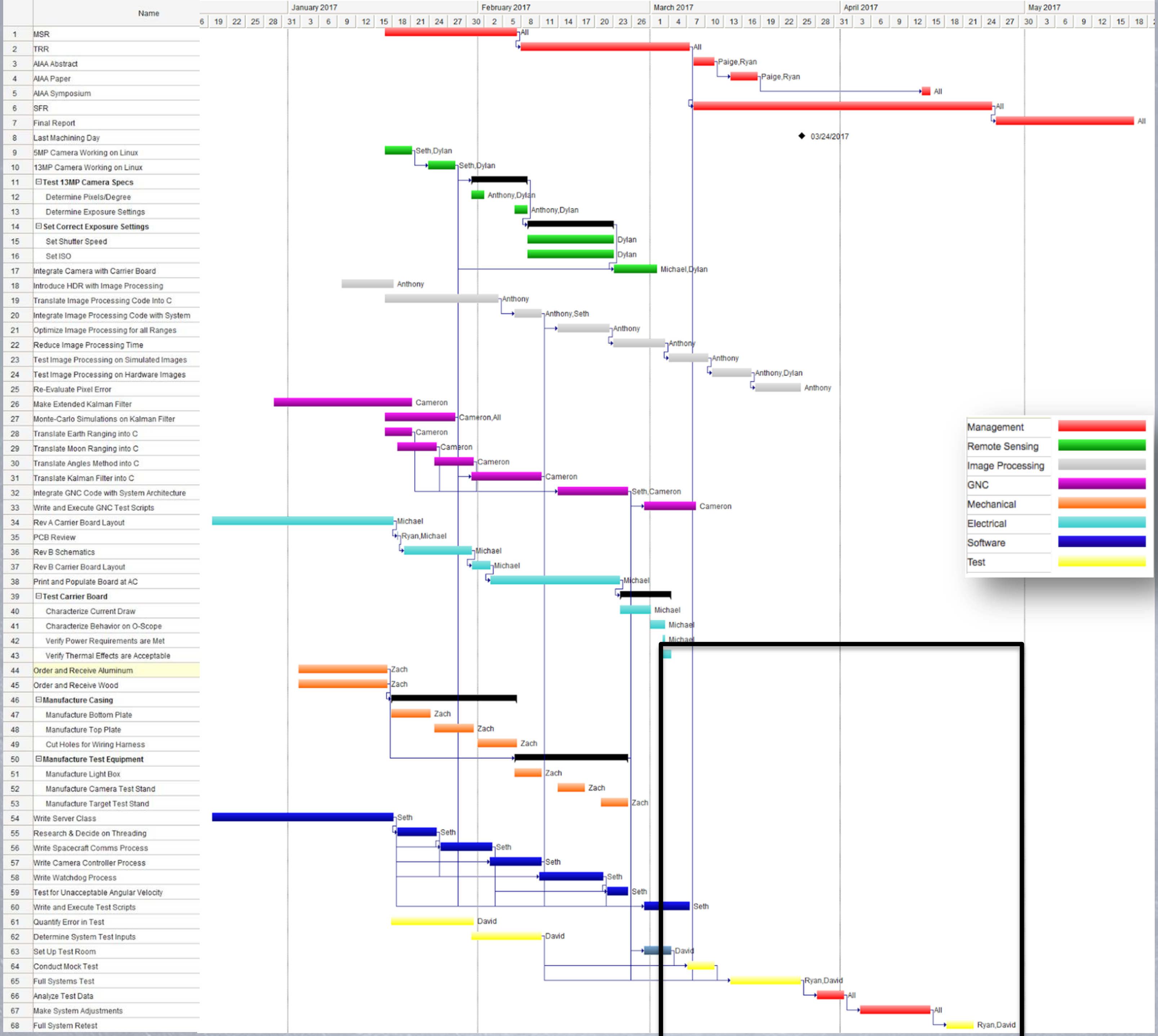


⊟ **Manufacture Casing**	
Manufacture Bottom Plate	Zach
Manufacture Top Plate	Zach
Cut Holes for Wiring Harness	Zach
⊟ **Manufacture Test Equipment**	
Manufacture Light Box	Zach
Manufacture Camera Test Stand	Zach
Manufacture Target Test Stand	Zach
Write Server Class	Seth
Research & Decide on Threading	Seth
Write Spacecraft Comms Process	Seth
Write Camera Controller Process	Seth
Write Watchdog Process	Seth
Test for Unacceptable Angular Velocity	Seth
Write and Execute Test Scripts	
Quantify Error in Test	David
Determine System Test Inputs	David
Set Up Test Room	Davi
Conduct Mock Test	
Full Systems Test	
Analyze Test Data	
Make System Adjustments	
Full System Retest	



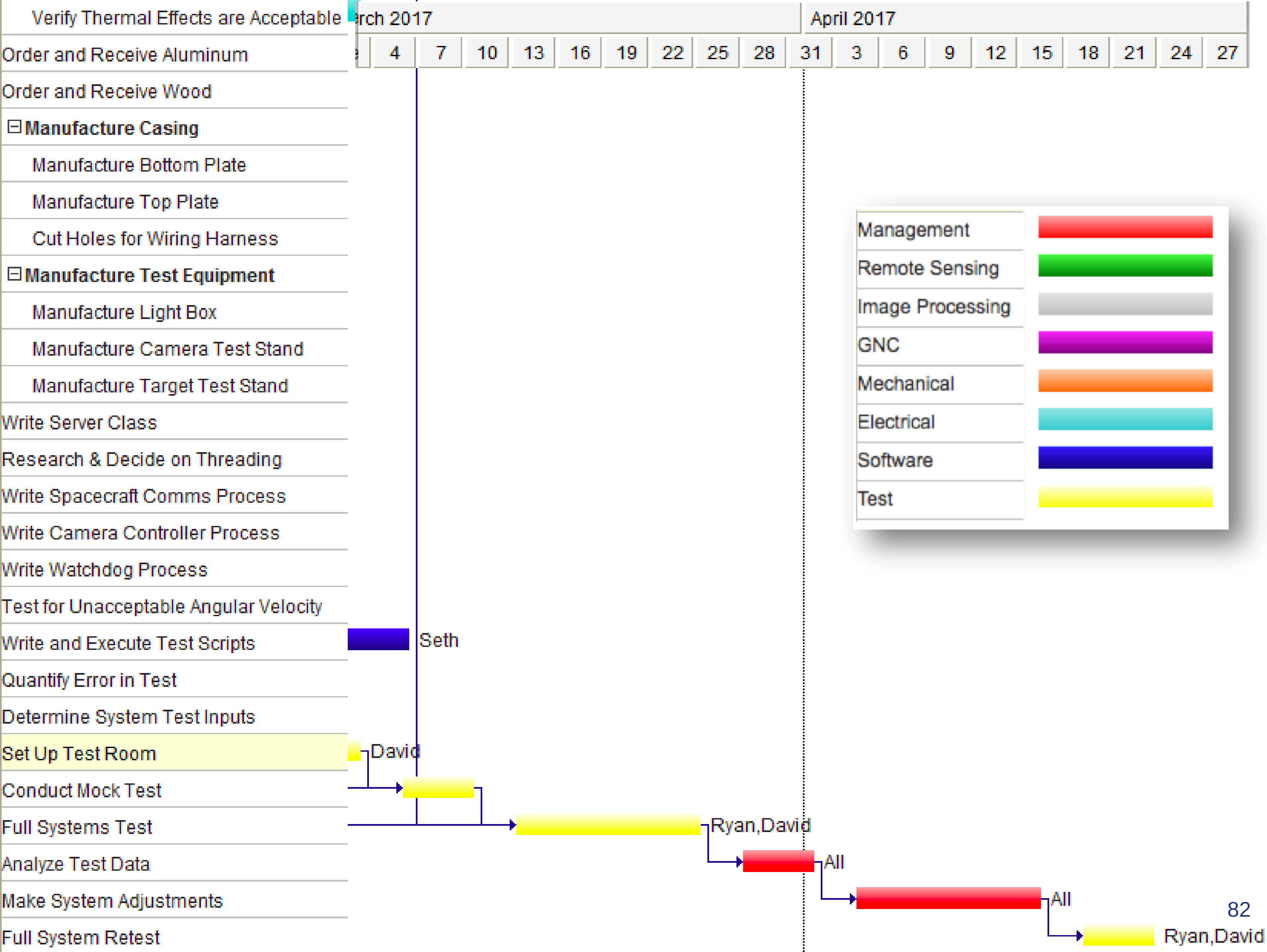
	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest

Management	🟥
Remote Sensing	🟩
Image Processing	▭
GNC	🟪
Mechanical	🟧
Electrical	▭
Software	🟦
Test	🟨

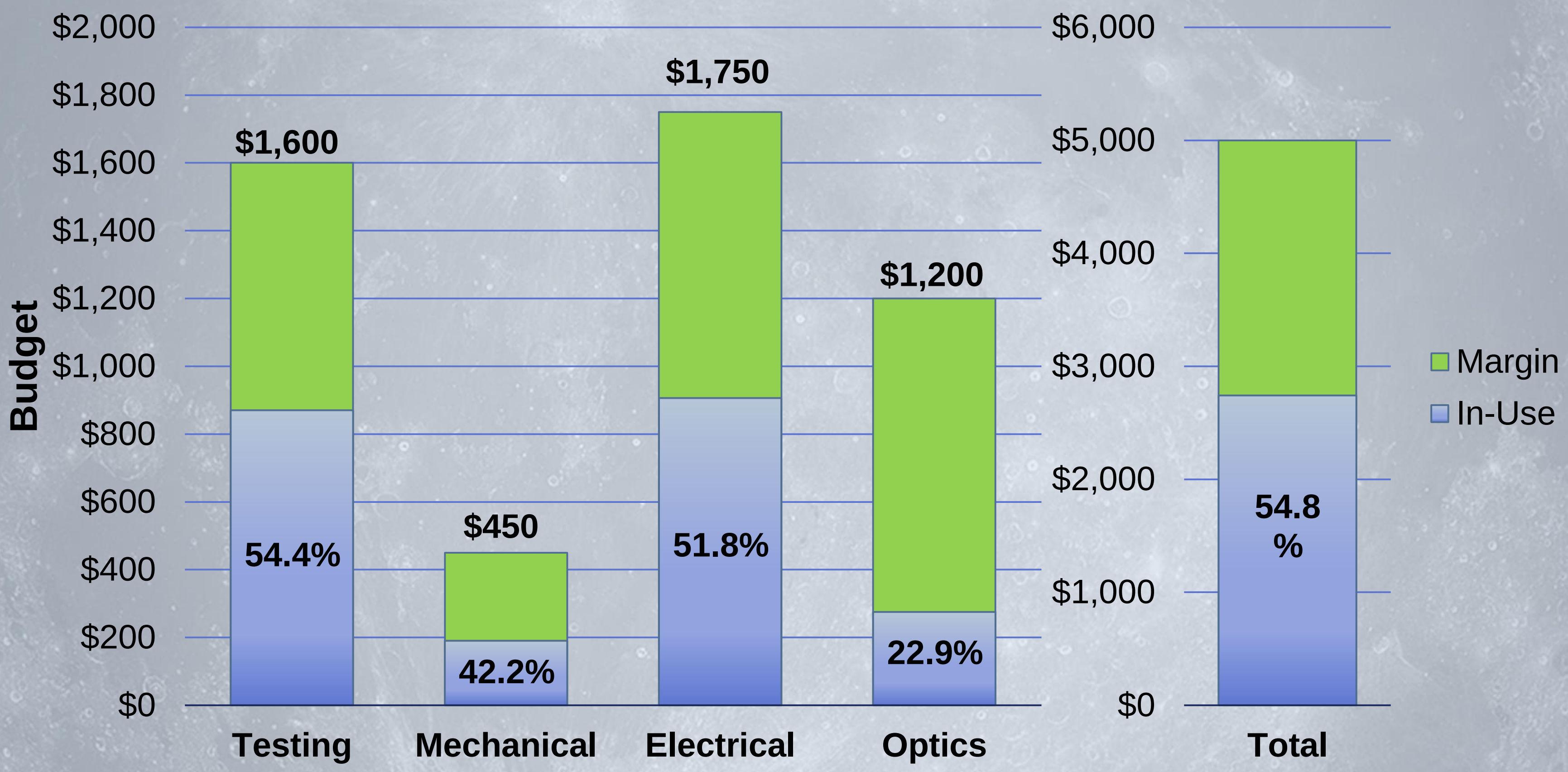


	Name
1	MSR
2	TRR
3	AIAA Abstract
4	AIAA Paper
5	AIAA Symposium
6	SFR
7	Final Report
8	Last Machining Day
9	5MP Camera Working on Linux
10	13MP Camera Working on Linux
11	⊟ **Test 13MP Camera Specs**
12	Determine Pixels/Degree
13	Determine Exposure Settings
14	⊟ **Set Correct Exposure Settings**
15	Set Shutter Speed
16	Set ISO
17	Integrate Camera with Carrier Board
18	Introduce HDR with Image Processing
19	Translate Image Processing Code Into C
20	Integrate Image Processing Code with System
21	Optimize Image Processing for all Ranges
22	Reduce Image Processing Time
23	Test Image Processing on Simulated Images
24	Test Image Processing on Hardware Images
25	Re-Evaluate Pixel Error
26	Make Extended Kalman Filter
27	Monte-Carlo Simulations on Kalman Filter
28	Translate Earth Ranging into C
29	Translate Moon Ranging into C
30	Translate Angles Method into C
31	Translate Kalman Filter into C
32	Integrate GNC Code with System Architecture
33	Write and Execute GNC Test Scripts
34	Rev A Carrier Board Layout
35	PCB Review
36	Rev B Schematics
37	Rev B Carrier Board Layout
38	Print and Populate Board at AC
39	⊟ **Test Carrier Board**
40	Characterize Current Draw
41	Characterize Behavior on O-Scope
42	Verify Power Requirements are Met
43	Verify Thermal Effects are Acceptable
44	Order and Receive Aluminum
45	Order and Receive Wood
46	⊟ **Manufacture Casing**
47	Manufacture Bottom Plate
48	Manufacture Top Plate
49	Cut Holes for Wiring Harness
50	⊟ **Manufacture Test Equipment**
51	Manufacture Light Box
52	Manufacture Camera Test Stand
53	Manufacture Target Test Stand
54	Write Server Class
55	Research & Decide on Threading
56	Write Spacecraft Comms Process
57	Write Camera Controller Process
58	Write Watchdog Process
59	Test for Unacceptable Angular Velocity
60	Write and Execute Test Scripts
61	Quantify Error in Test
62	Determine System Test Inputs
63	Set Up Test Room
64	Conduct Mock Test
65	Full Systems Test
66	Analyze Test Data
67	Make System Adjustments
68	Full System Retest

Management	🟥
Remote Sensing	🟩
Image Processing	⬜
GNC	🟪
Mechanical	🟧
Electrical	🟦
Software	🟦
Test	🟨


Verify Thermal Effects are Acceptable
Order and Receive Aluminum
Order and Receive Wood
⊟ Manufacture Casing
Manufacture Bottom Plate
Manufacture Top Plate
Cut Holes for Wiring Harness
⊟ Manufacture Test Equipment
Manufacture Light Box
Manufacture Camera Test Stand
Manufacture Target Test Stand
Write Server Class
Research & Decide on Threading
Write Spacecraft Comms Process
Write Camera Controller Process
Write Watchdog Process
Test for Unacceptable Angular Velocity
Write and Execute Test Scripts
Quantify Error in Test
Determine System Test Inputs
Set Up Test Room
Conduct Mock Test
Full Systems Test
Analyze Test Data
Make System Adjustments
Full System Retest
rch 2017
April 2017
4
7
10
13
16
19
22
25
28
31
3
6
9
12
15
18
21
24
27
Seth
David
Ryan,David
All
All
Ryan,David
Management
Remote Sensing
Image Processing
GNC
Mechanical
Electrical
Software
Test

Budget


Budget
$2,000
$1,800
$1,600
$1,400
$1,200
$1,000
$800
$600
$400
$200
$0
$1,600
54.4%
$450
42.2%
$1,750
51.8%
$1,200
22.9%
Testing
Mechanical
Electrical
Optics
$6,000
$5,000
$4,000
$3,000
$2,000
$1,000
$0
54.8
%
Total
Margin
In-Use

Test Plan

Test	Equipment & Facilities	Personnel	Date
Software Test: Does the data filter, pre-processing, and image processing work within the system software architecture as expected?	None	S. Zegelstein, A. Torres, C. Maywood, D. Walden	2/27/16 - 3/6/16
Hardware Test: Does the system turn on and complete all nominal functions upon initialization while remaining within power requirements?	Oscilloscope, ESD workstation, power supply	M. Ricciardi, D. Richards, D. Walden	2/27/16 - 3/6/16
Problem Simulation Test: Can the system deal with issues with temperature, spacecraft inputs, bad photos, glare, etc?	None	D. Walden, R. Cutter	3/6/16 - 3/13/16
Operational Systems Test: Does the system produce a solution with the required accuracy?	Dark room, light box, test stand	D. Walden, R. Cutter, Z. Folger	3/13/16 - 3/24/16

Test Equipment & Facilities

Facility / Equipment	**Resource**
Geared Tripod Head	Purchased
Tripods	Borrowed from team members & and test facility
Simulated Celestial Body	Machined in Aerospace Machine Shop
Light Panel	Purchased
Measurement Tools	Purchased and borrowed from test facility/ITLL/aerospace department
Simulated Spacecraft Computer	Purchased a ZedBoard
Test Stands	Machined in Aerospace Machine Shop
Dark Room	ECEE 2B49A headed by Dr. Cash, who has granted permission for OSPRE to use the facility next semester

References

1. Samtec, "0.40mm Razor Beam LP Ultra Fine Pitch Socket Strip," 1 October 2016. [Online]. Available: https://www.samtec.com/products/ss4.
2. Analog Devices, "ADT7311: Automotive, ±0.5°C Accurate, 16-Bit," Analog Devices, Norwood, MA, 2011.
3. Future Technology Devices Intl., "FT234XD USB TO BASIC UART IC V1.2," FTDI Ltd., Glasgow, UK, 2015.
4. Richtek, "RT7258 8A, 24V, 600kHz Step-Down Converter," Richtek Technology Corp., San Jose, 2013.
5. Intrinsyc Technologies Corp., "Open-Q 410 Development Kit BSP Programming Guide V1.0," Intrinsyc Technologies Corp., Vancouver, BC, 2016.
6. Intrinsyc Technologies Corp., "Intrinsyc Open-QTM 410 (APQ8016) Carrier Board Design Guide R1.3," Intrinsyc Technologies Corp., Vancouver, BC, 2016.
7. Intrinsyc Technologies Corp., "Open-QTM 410 Development Kit based on the Qualcomm® SnapdragonTM 410 processor (APQ8016) User Guide R1.3," Intrinsyc Technologies Corp., Vancouver, BC, 2016.
8. Intrinsyc Technologies Corp., "Intrinsyc Open-QTM 410 (APQ8016) Development Kit Technical Note 18: Camera Board Design Guide R1.0," Intrinsyc Technologies Corp., Vancouver, BC, 2016.
9. Intrinsyc Technologies Corp., "Open-Q 410 Development Carrier Board Electrical Schematics 8May2015," Intrinsyc Technologies Corp., Vancouver, BC, 2015.
10. Intrynsic Technologies Corp., "APQ-8016 Open-Q SOM Electrical Schematics 26Mar2015," Intrynsic Technologies Corp., Vancouver, BC, 2016.

Thank You

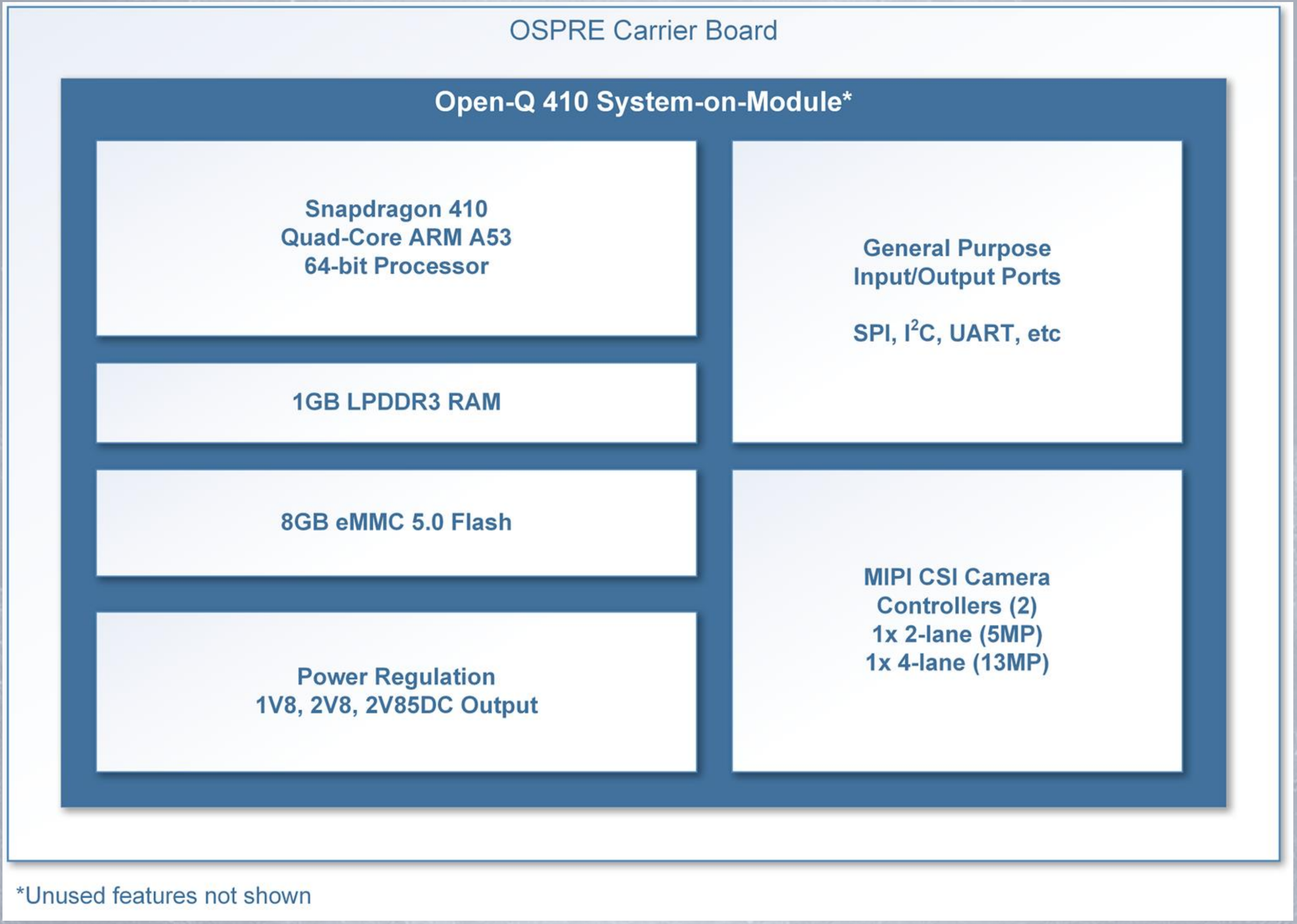
Paige Arthur	PM
Ryan Cutter	Systems
Seth Zegelstein	Software
Michael Ricciardi	Electrical
Anthony Torres	Image Processing
Cameron Maywood	Navigation
Dylan Richards	Remote Sensing
Zach Folger	Mechanical
David Walden	Testing

BACKUP SLIDES

Electrical/Software Backup Slides

OSPRE Electrical Architecture

System-on-Module Functional Capabilities



OSPRE Carrier Board

Open-Q 410 System-on-Module*

Snapdragon 410
Quad-Core ARM A53
64-bit Processor

1GB LPDDR3 RAM

8GB eMMC 5.0 Flash

Power Regulation
1V8, 2V8, 2V85DC Output

General Purpose
Input/Output Ports

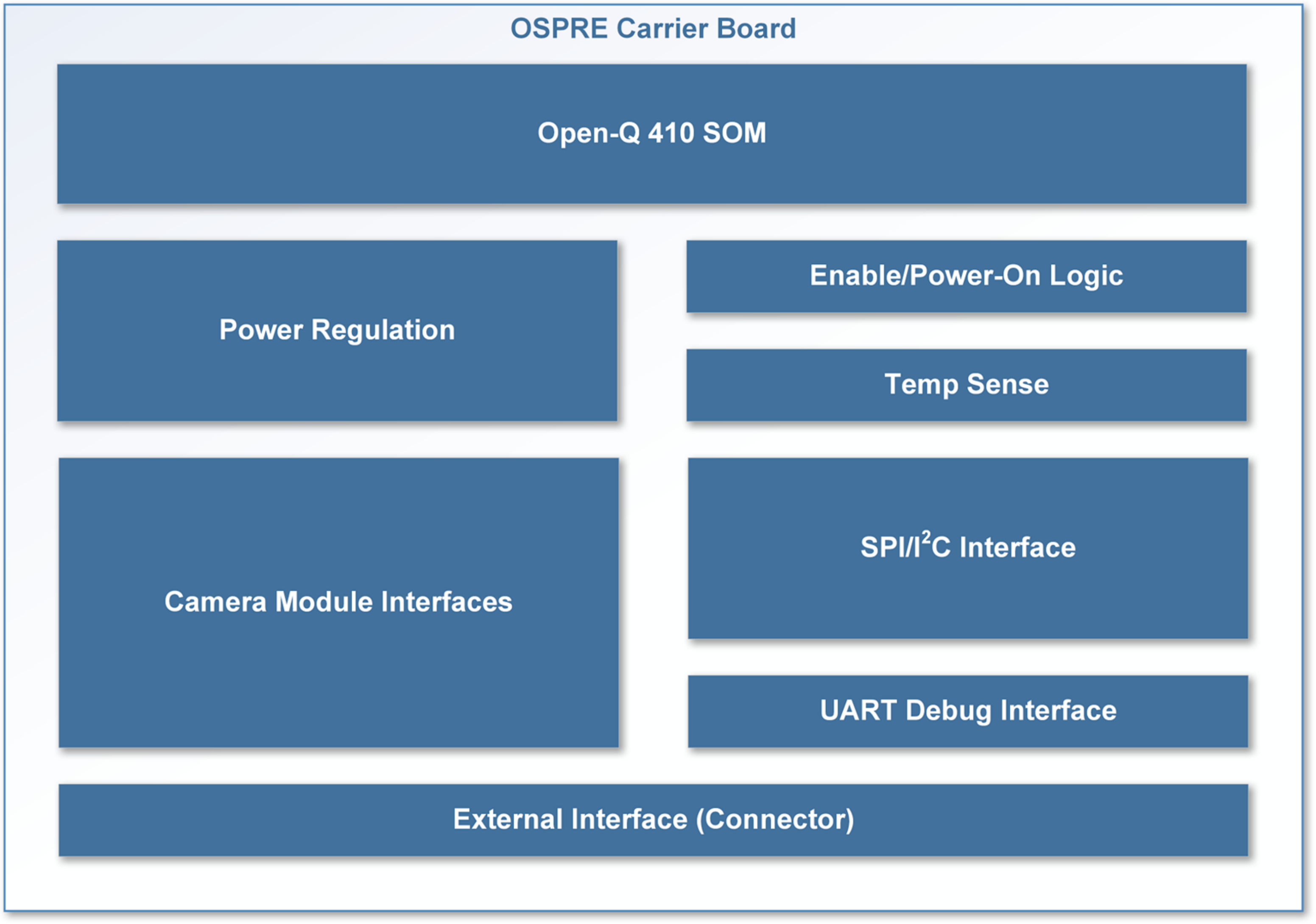
SPI, I^2C, UART, etc

MIPI CSI Camera
Controllers (2)
1x 2-lane (5MP)
1x 4-lane (13MP)

*Unused features not shown

OSPRE Electrical Architecture

Carrier Board Functional Block



OSPRE Carrier Board

Open-Q 410 SOM

Power Regulation

Enable/Power-On Logic

Temp Sense

Camera Module Interfaces

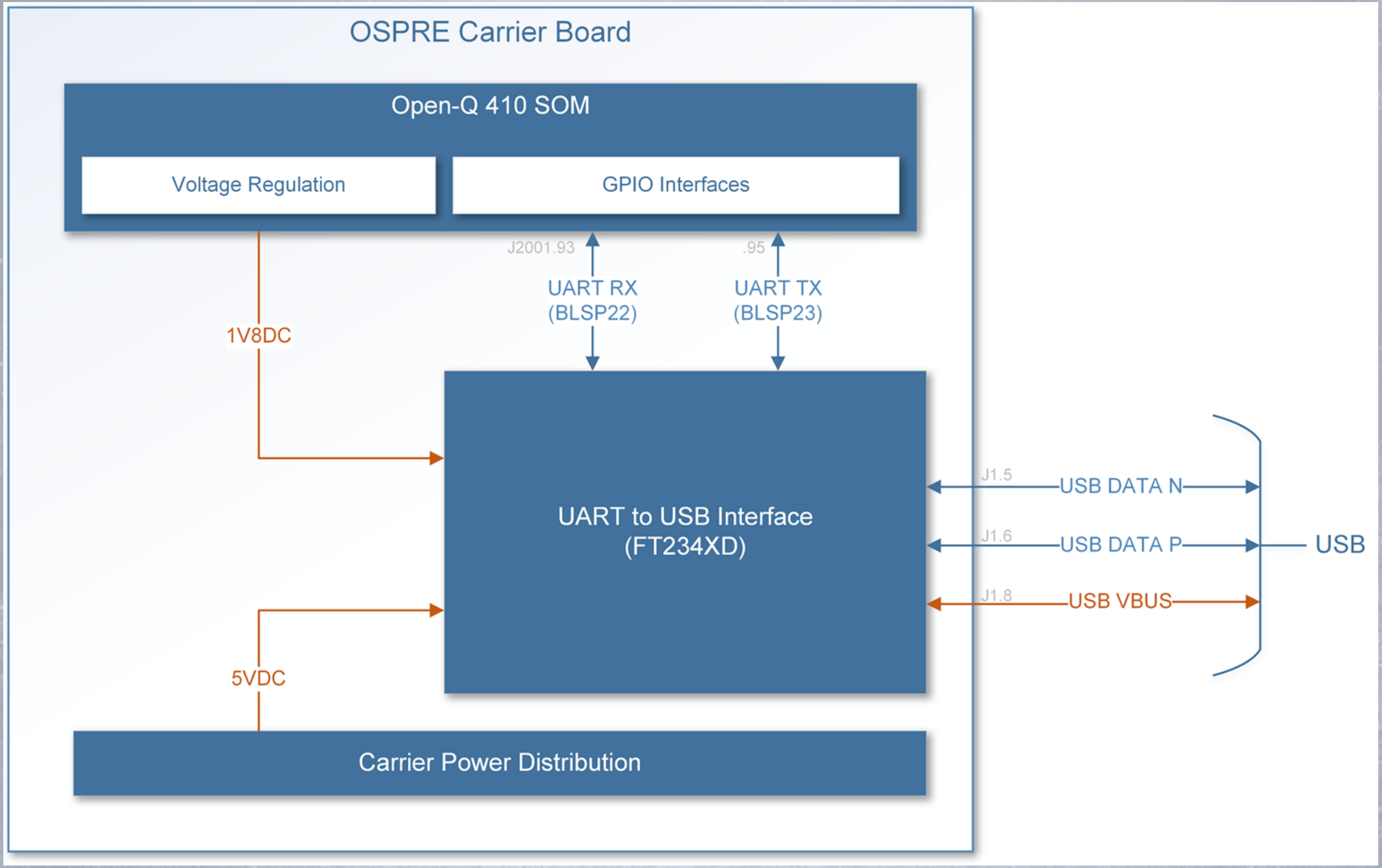
SPI/I^{2}C Interface

UART Debug Interface

External Interface (Connector)

OSPRE Electrical Architecture

Serial Debug Interface (UART)



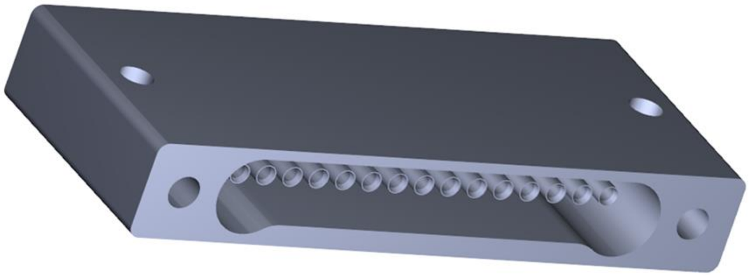
OSPRE Electrical Architecture

External Interface & Pin-Out

TE Connectivity 1589462, 15-Position Connector

Physical Description

- Right-angle, surface-mount
- Nickel-plated Aluminum 6061-T6 (MIL-C-26074)
- 14.2 x 5.7 x 2.2 mm



CONNECTOR PINOUT		
PARNO		**TE 1589426 15Pos**
REFDES	**CONTACT**	**SIGNAL NAME**
J1	1	12VDC
J1	2	12VDC
J1	3	12VDC_RTN
J1	4	12VDC_RTN
J1	5	UART_USB_DATA_N
J1	6	UART_USB_DATA_P
J1	7	UART_USB_GND
J1	8	UART_USB_VBUS
J1	9	CHASSIS_GND
J1	10	SPI_CLK
J1	11	SPI_SS
J1	12	SPI_DATA_MOSI
J1	13	SPI_DATA_MISO
J1	14	I2C_SCL
J1	15	I2C_SDA

OSPRE Electrical Architecture

Electrical Cost and Power Budget

Components	**Power and Volume Budget**						
Description	**Dollar, Approx. Max.**	**Mass, Approx. Max.**	**Real Estate, Approx.**	**Height, Approx.**	**Voltage Supply**	**Current (External), Approx. Max**	**Power, Approx. Max.**
	USD	**g**	**mm**	**mm**	**VDC**	**mA**	**mW**
PCB (4L) Manufacture and Populate	300.00	6	44x44	4	NA	NA	NA
External Connector, 15-Pos (TE 1589462)	200.00	2	14.2x5.7	2.18	NA	NA	NA
B2B Connectors (SS4)	8.20	0.5	23.54 x4.35				
Open-Q 410 SOM	80.00	5	44x27	4	3.7	100	1100
DC-DC Step-Down Converter (RT7258)	1.00	1	4x3	0.75	12.0	2.5	30
DC-DC Step-Down Converter (RT7258)	1.00	1	4x3	0.75	12.0	2.5	30
Temp Sensor (ADT7311)	2.00	1	5x4	1.75	3.7	0.1	1
Temp Sensor (ADT7311)	2.00	1	5x4	1.75	3.7	0.1	1
Image Sensor Module 5MP (OV5640)	70.00	1.5	21x8.5	6	3.7	150	700
Image Sensor Module 13MP (IMX214)	125.00	1.5	21x8.5	6	3.7	150	700
USB Serial UART Controller (FT234XD)	2.30	1	3x3		5.0	NONFLT	NONFLT
TOTAL	791.50	21.5	720			405.2	2562
BUDGET	2,000.00	800	2100	8		500	3000
MARGIN	1,208.50	778.5	1380	2		94.8	438
ASSOCIATED REQUIREMENT	**BUDGET**	**DR2.6**	**DR2.7**	**DR2.7**	**DR2.1.1**	**DR2.1.2**	**DR2.1.3**

OSPRE Electrical Architecture

Imaging Sensor

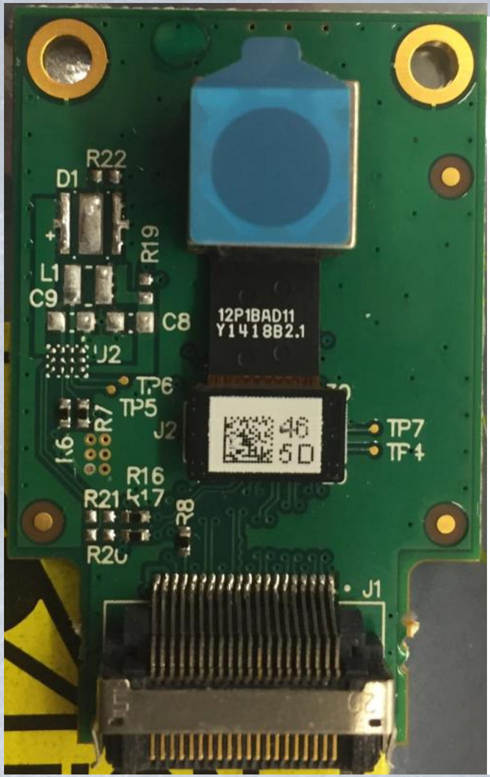
Design Drivers:

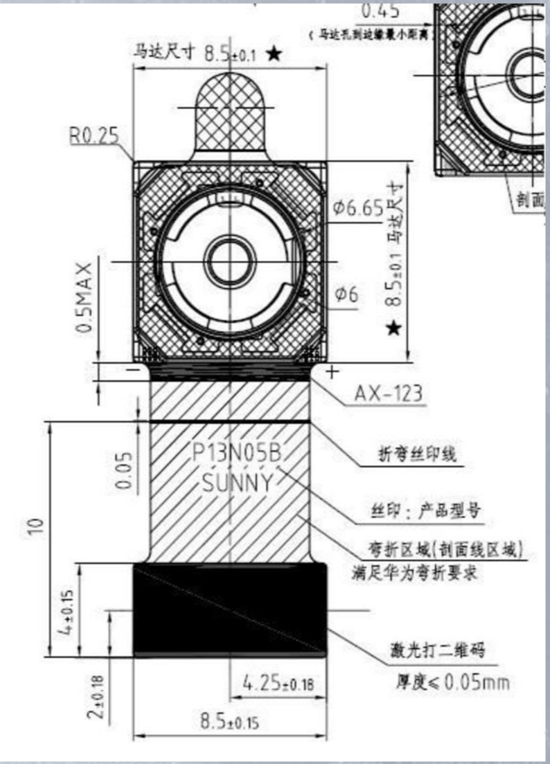
FR 1.0 - Accuracy

FR 2.0 - Volume

Camera Specs:

- 8.5 x 8.5 x 5.3 mm lens package
- 13MP sensor (IMX214)
- FOV = 60° x 45°
- ~70 pixels/°





Further Software Architecture Detail

- OSPRE needs multiple processes in order to take advantage of two separate cpu’s
- The OSPRE system was designed to be modular
  - If the camera or S/C change only the Camera Controller or SCComms process need to change
- All processes will be derived from an virtual server class containing the common elements of all OSPRE’s server processes

OSPRE Software Architecture

Overview

OSPRE will run on 2 Linux OS CPUs. Processes will be split between:

- ZedBoard (Simulated Spacecraft) [2 core]
- Onboard SOM processor [4 core]

OSPRE consists of 5 major processes:

- Camera Controller
- Image Processing
- Navigation
- S/C Comms
- WatchDog

Zedboard will simulate the spacecraft by:

- Providing all necessary inputs (Time, Quaternion, Angular Velocity, etc.)
- Reading in OSPRE results and updates. (State Vector + Error, Status, Pointing Adjustments)

Software Design - Startup

- More Complicated due to using two different CPUs
- When the OSPRE sensor package receives power, the microprocessor onboard will turn on and boot up Linux OS. On startup, a script will be run to start our OSPRE system processes residing on SOM.
- OSPRE will have another startup script for the Zed Board to start the OSPRE server processes on the Zed Board

Software Design - Server Process 1

- All processes will listen at a defined port number for new connection requests
- Each process will have a defined number of connections
  - All processes will be accepting connections
  - Processes will connect to each other in a predefined manner
  - Only WatchDog will be written to support N connections

Software Design - Server Process 2

- Servers will be using a call to select and registering for read and write events when interested in them
- The select call will be using timeouts to allow for the handling of error cases
- SIGPIPES are ignored so that the Server doesn’t quit when the client disconnects
- In the event that two processes lose a connection, the process acting as the client will attempt to reconnect to the process acting as the Server

Software Design - Server Process 3

-OSPRE Servers will not follow the classic Server to Client relationship. OSPRE Servers will act as both a Server and Client.

–All Processes will be listening on a dedicated socket for connection requests (Typical Server). Specific processes will connect to others processes because they need information from said process (Typical Client).

-An abstract OSPRE server class will be written in C++ from which all process classes will be derived from.

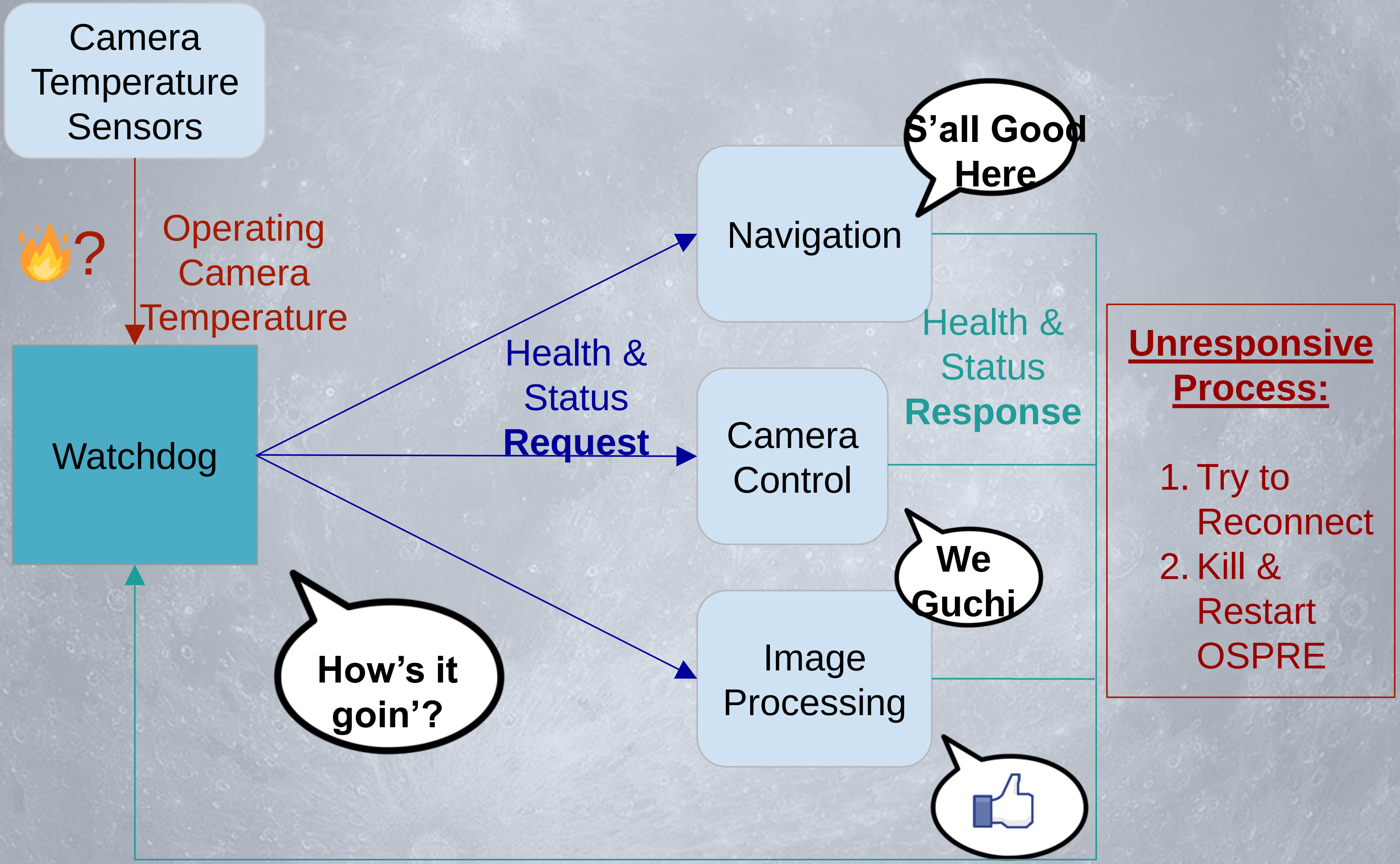
–The server class will contain a virtual run method, which each individual process class will implement.

–The Server class will implement generic methods that all derived classes need such as init(), openServerSocket(), and connectToServer()

–The Server class will be built by composition and have instances of other classes to deal with intra-process communication (details below)

OSPRE Software Architecture

Watchdog



Camera Temperature Sensors

🔥?

Operating Camera Temperature

Watchdog

Health & Status **Request**

Navigation

Camera Control

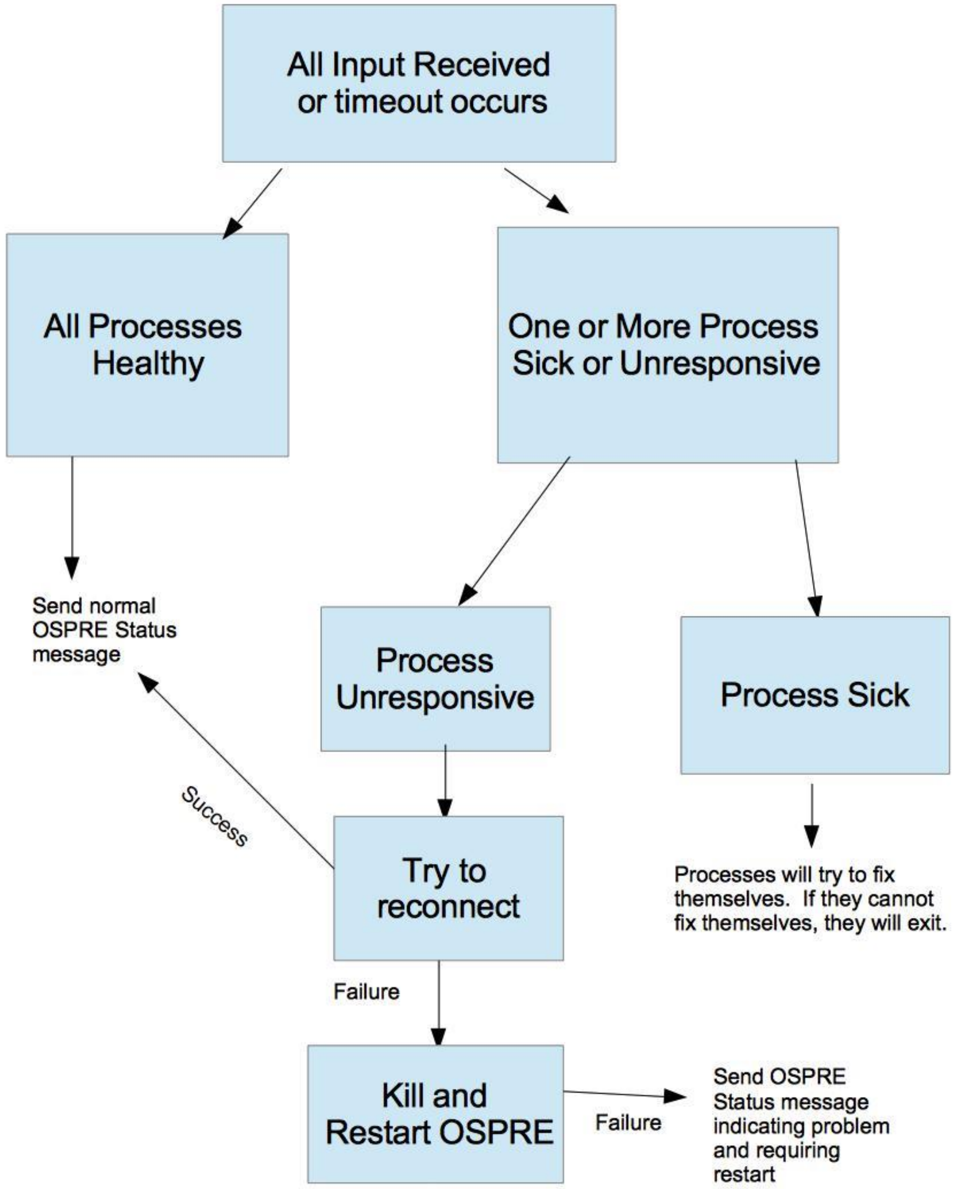
Image Processing

Health & Status **Response**

Unresponsive Process:

1. Try to Reconnect
2. Kill & Restart OSPRE

Watch Dog / Controller Logic Flow



All Input Received or timeout occurs

All Processes Healthy

One or More Process Sick or Unresponsive

Send normal OSPRE Status message

Process Unresponsive

Process Sick

Success

Try to reconnect

Processes will try to fix themselves. If they cannot fix themselves, they will exit.

Failure

Kill and Restart OSPRE

Failure

Send OSPRE Status message indicating problem and requiring restart

Software Design - Error Cases

Error: Process Failure

Action: Restart OSPRE system (Power on/ Off), (S/C responsibility)

Error: GNC cannot achieve accurate solution to within error bounds

Action: Report solution to current error

Error: Image Processing cannot find body (Earth / Moon) in the picture

Action: Report status to with error message S/C, attempt to change Camera setting, keep trying

Error: Not pointing at the right targets (Earth / Moon)

Action: Send additional pointing request to S/C and wait

Software Design - Shutdown

- More Complicated due to using two different CPUs
- Cut power to OSPRE sensor package
- Send SIG_KILL signals to processes on ZedBoard

Image Processing Backup Slides

Image Processing

- Circular Hough Transform Overview

- Celestial body outline of radius R

    - Outline found with gradient of image

    - Edge could also be an arc (for phased bodies)

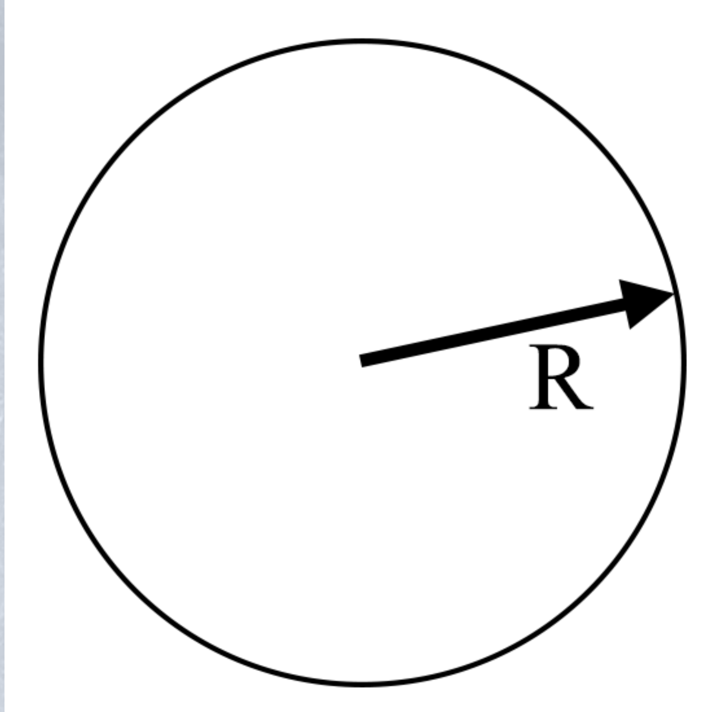


Image Processing

- Move along edge
- Draw circle of votes with a radius R
- Keep these votes in a matrix, called the accumulator matrix

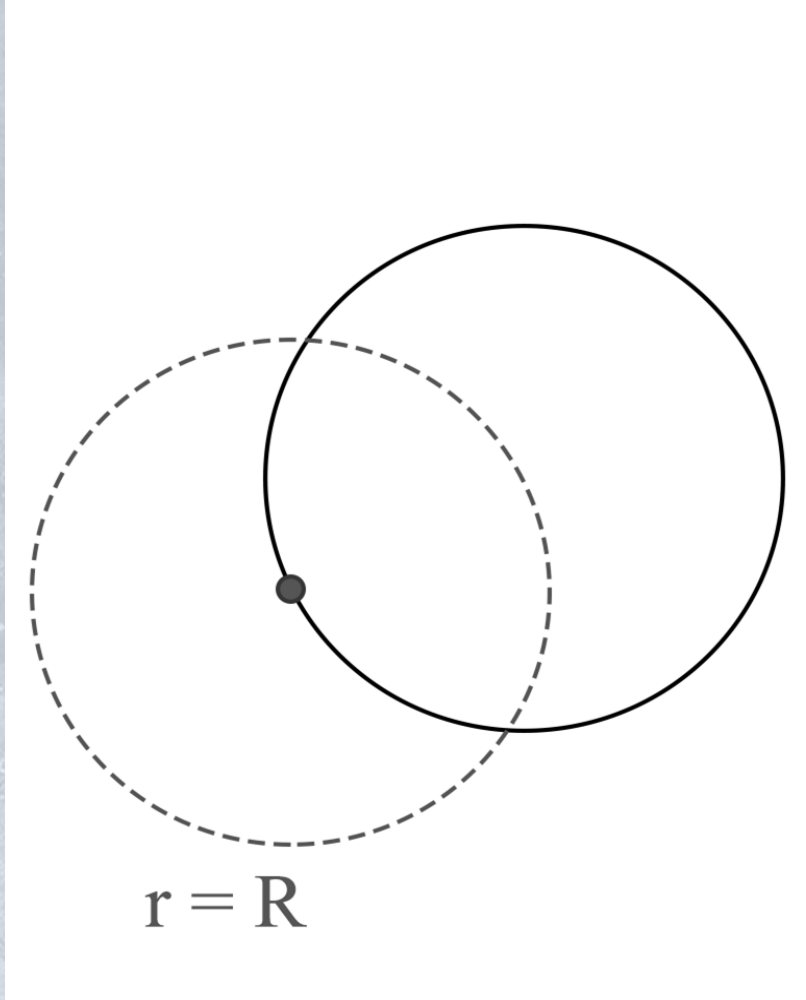


Image Processing

- Keep voting along edge

- These votes will then overlap at the center

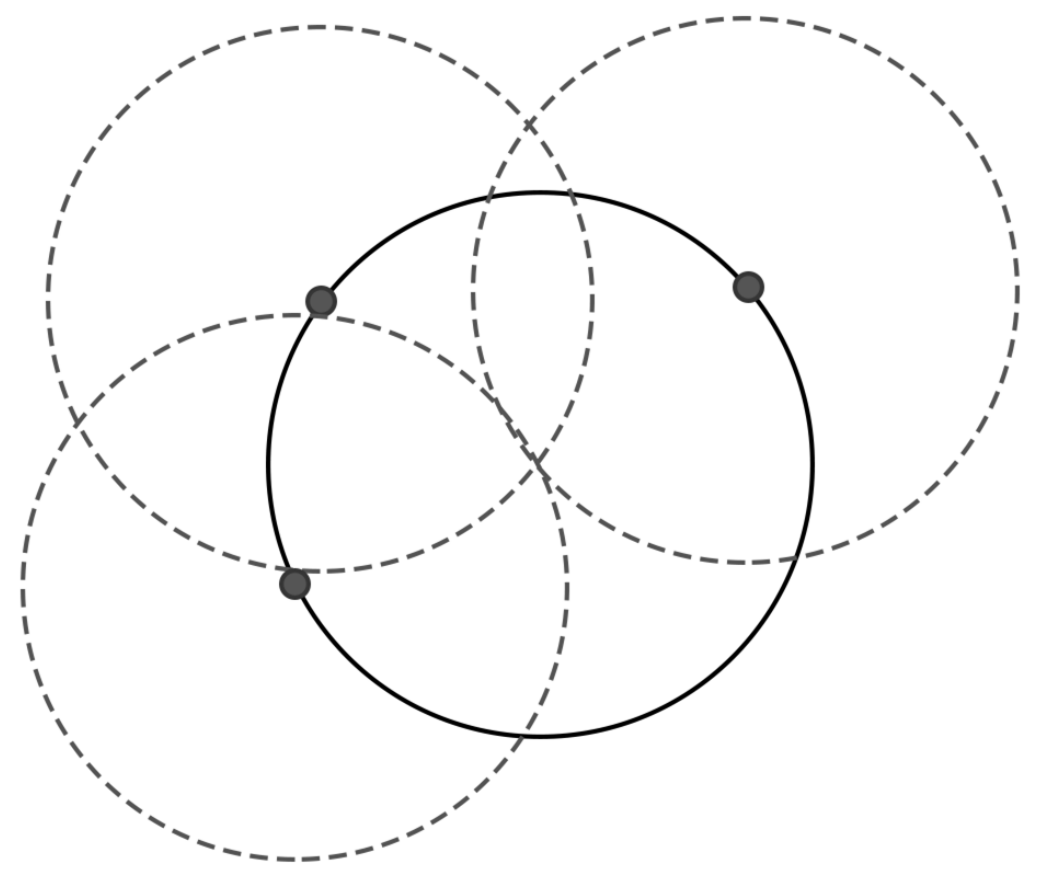


Image Processing

- The point with the most votes, the most overlap, is the calculated center

- This finds both the center and the best radius at the same time

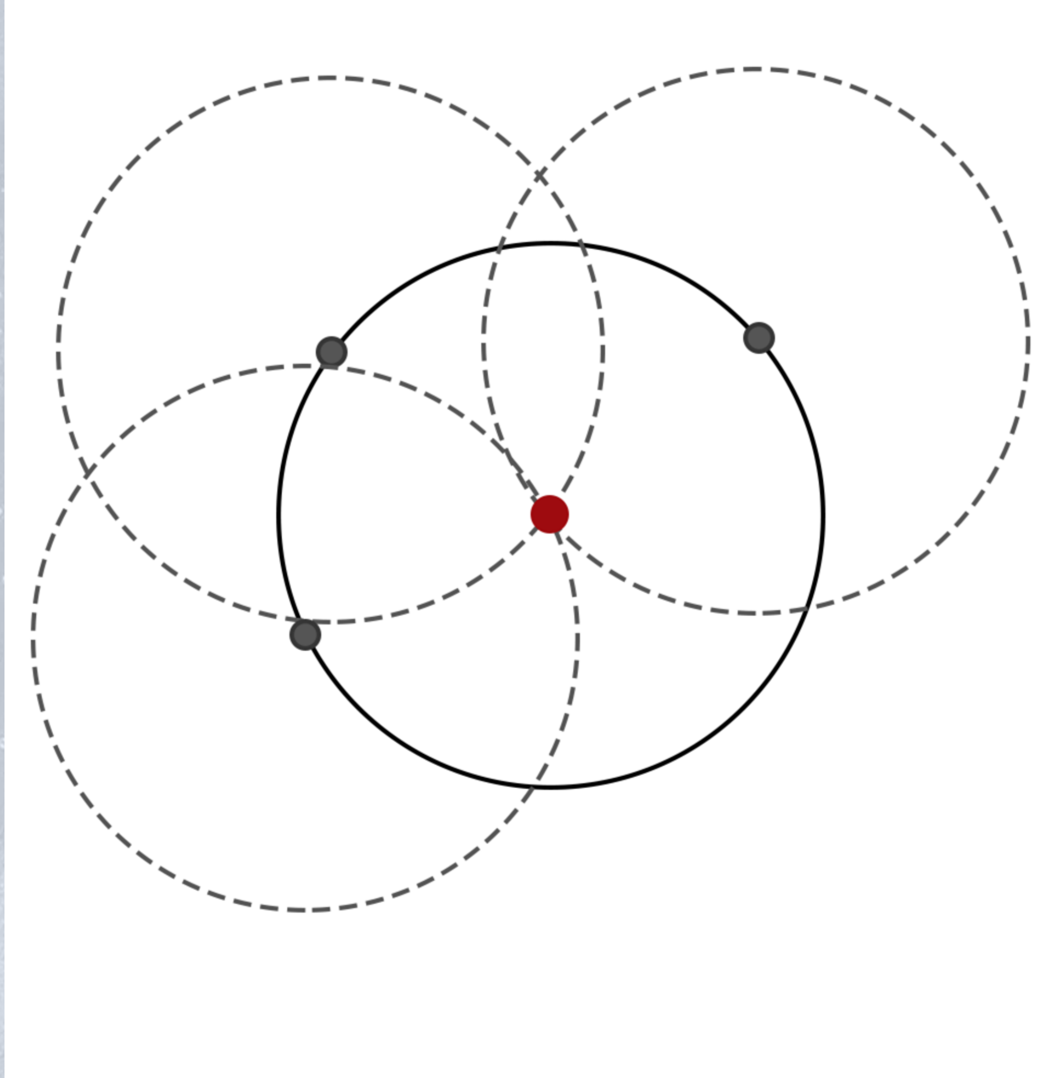


Image Processing

- A more efficient method uses normal vectors
- Along the same celestial body outline, draw a normal vector outwards
  - The Moon or Earth being lighter than space allows for the direction to be determined

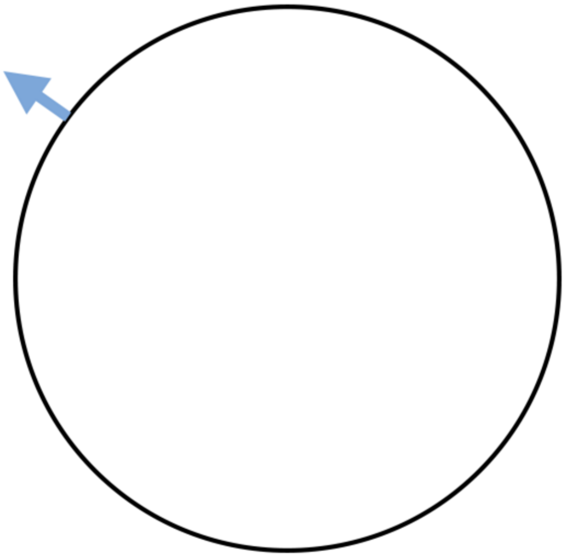


Image Processing

- Draw a line from the normal vector

- Only draw a portion of the line around the guessed radius

- Further, each point on the line has an associated phase
    - More on phase encoding further on

- This method decreases computation time, and increases accuracy

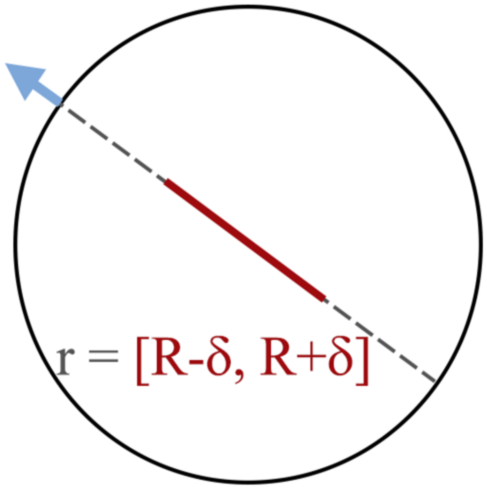


Image Processing

- Keep moving around edge and drawing normal vectors, and the subsequent radius guesses

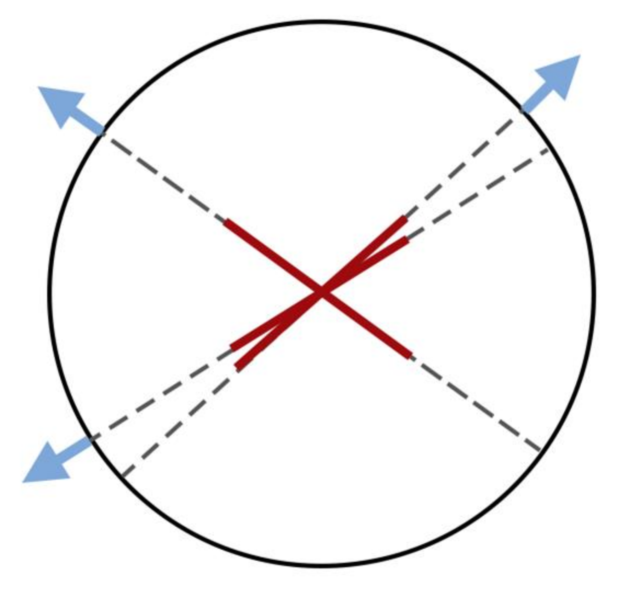


Image Processing

- The overlap of these guesses indicate the center

- The phase constructively interferes at the center, which corresponds to a radius

- This again finds the center and the diameter of the celestial body

- This also works for phases, but with less votes
    - Increasing the sensitivity of the algorithm compensates

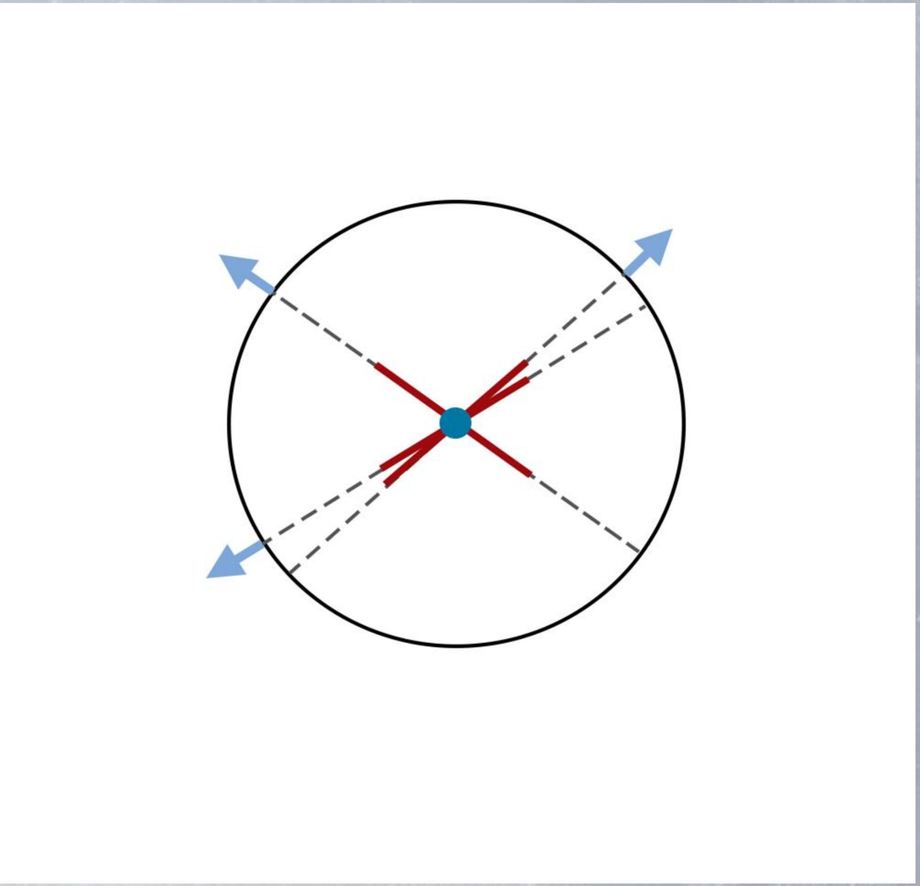
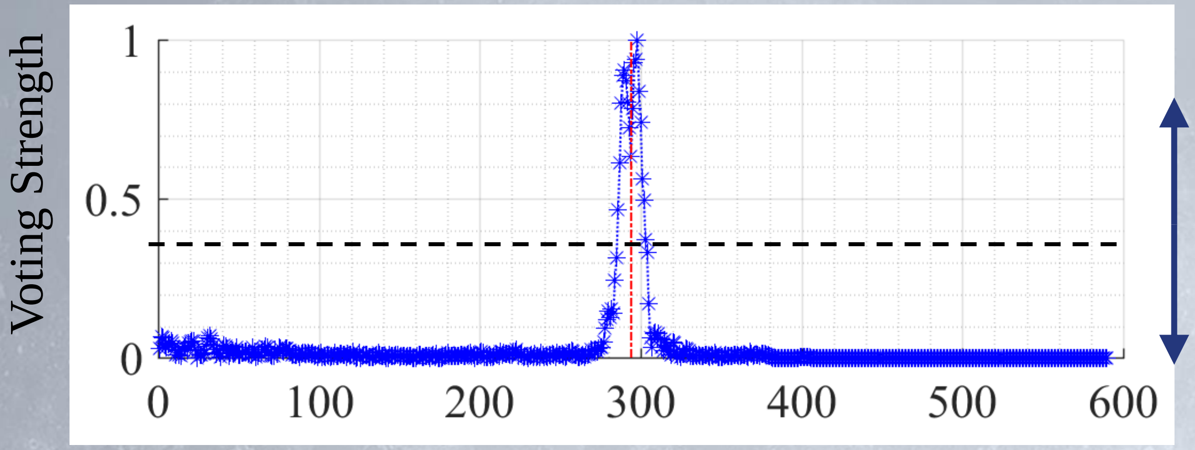


Image Processing



Kept Above Threshold

Discarded Below Threshold

Votes are kept or discarded based on a strength threshold. Decreasing the threshold potentially allows for more noise, but also allows for more sensitive analysis with lower strength.

Image Processing

- Each radius guess, for each point on an edge, has phase encoding associated
  - Using Euler’s Formula, this moves the voting matrix into the complex domain

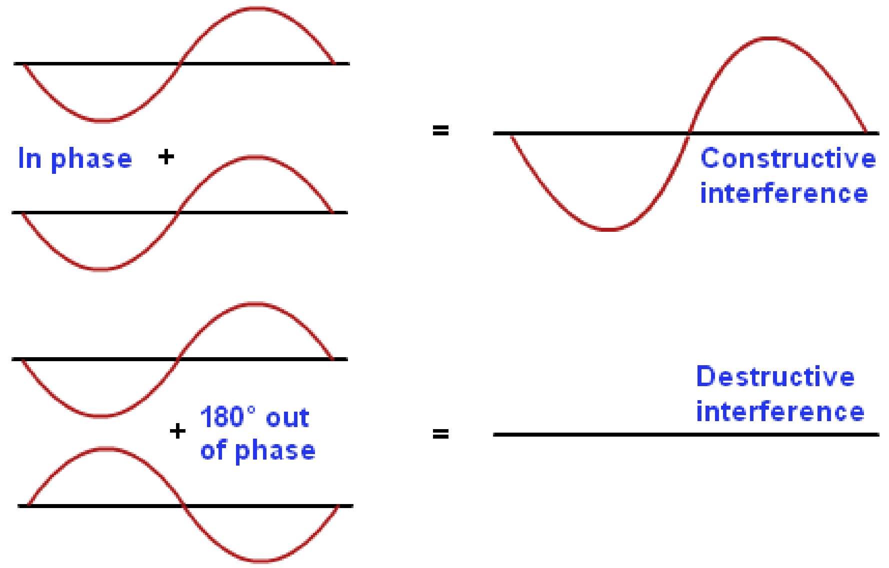
- The different phases will then constructively interfere for common radii, and destructively interfere otherwise

- This encoding provides the **coherence** of the circle finding algorithm

- Phase encoding also makes the algorithm more robust as noise will tend to destructively interfere as there is no order

$$z = e^{\varphi \iota}$$

Phase



Atherton, T.J., Kerbyson, D.J., *The Coherent Circle Hough Transform*, 1993. http://www.bmva.org/bmvc/1993/bmvc-93-027.pdf

http://physics.tutorcircle.com/waves/wave-interference.html

Image Processing

- We use a Sobel Kernel to find the edges of an image
- Performed with Kernel Convolution
  - An image is simply a large matrix
  - Kernel matrix is ‘moved’ through matrix to calculate new values based on neighbor values

Sobel Operator:

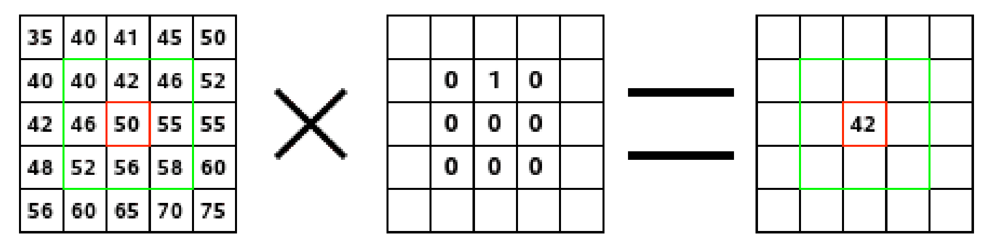
-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Basic Kernel Convolution Idea:



35	40	41	45	50
40	40	42	46	52
42	46	50	55	55
48	52	56	58	60
56	60	65	70	75

×

	0	1	0	
	0	0	0	
	0	0	0	

=

		42		

Result:

https://docs.gimp.org/en/plug-in-convmatrix.html
http://edge.kitiyo.com/2009/intro/relevance-of-the-project.html

Image Processing

- Top-down view of voting accumulator matrix

- Side graphs are 2D views of the matrix

- Majority of the analyzed image can be approximated as zero
    - Allows for precise locations to be found

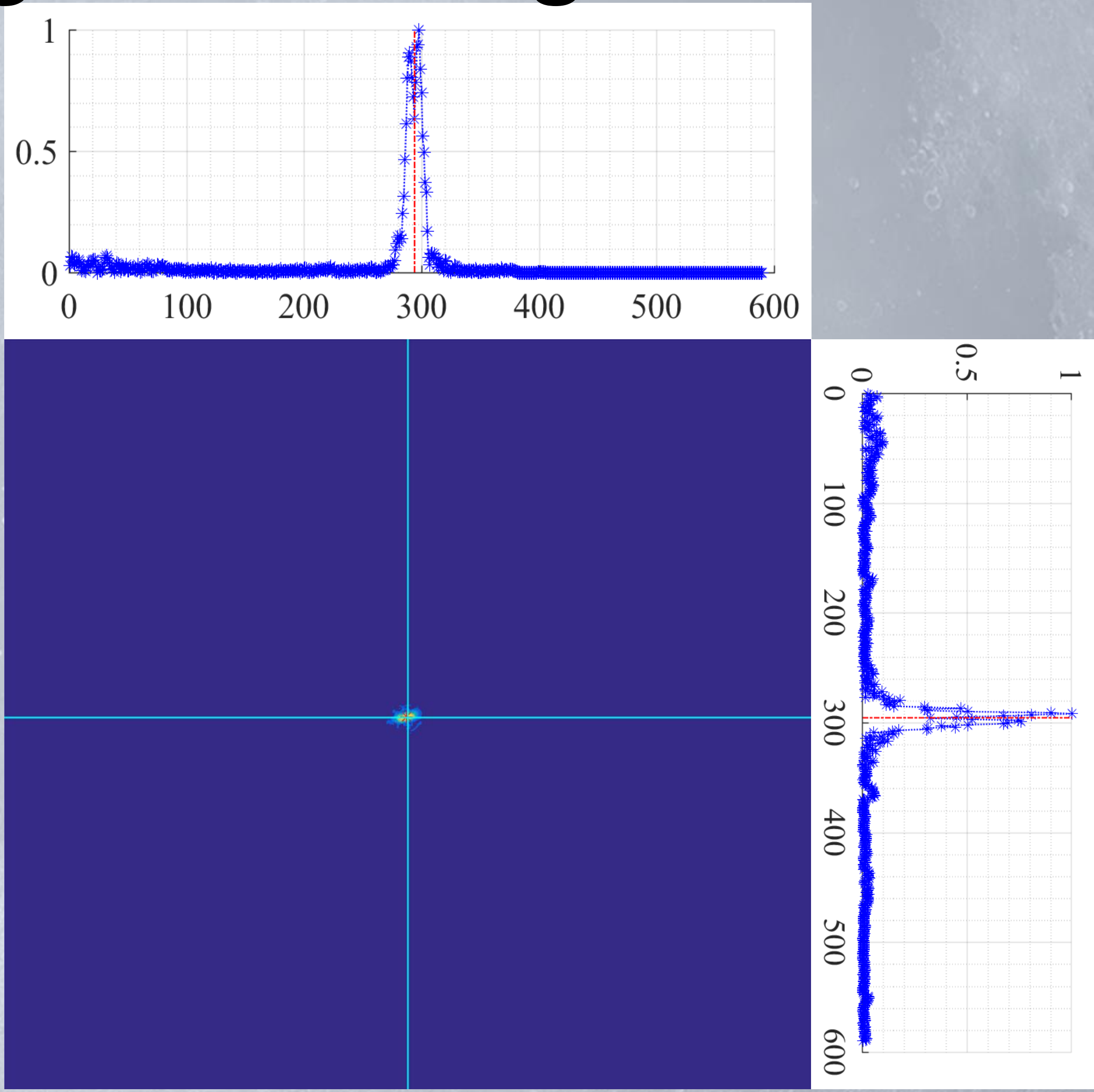


Image Processing

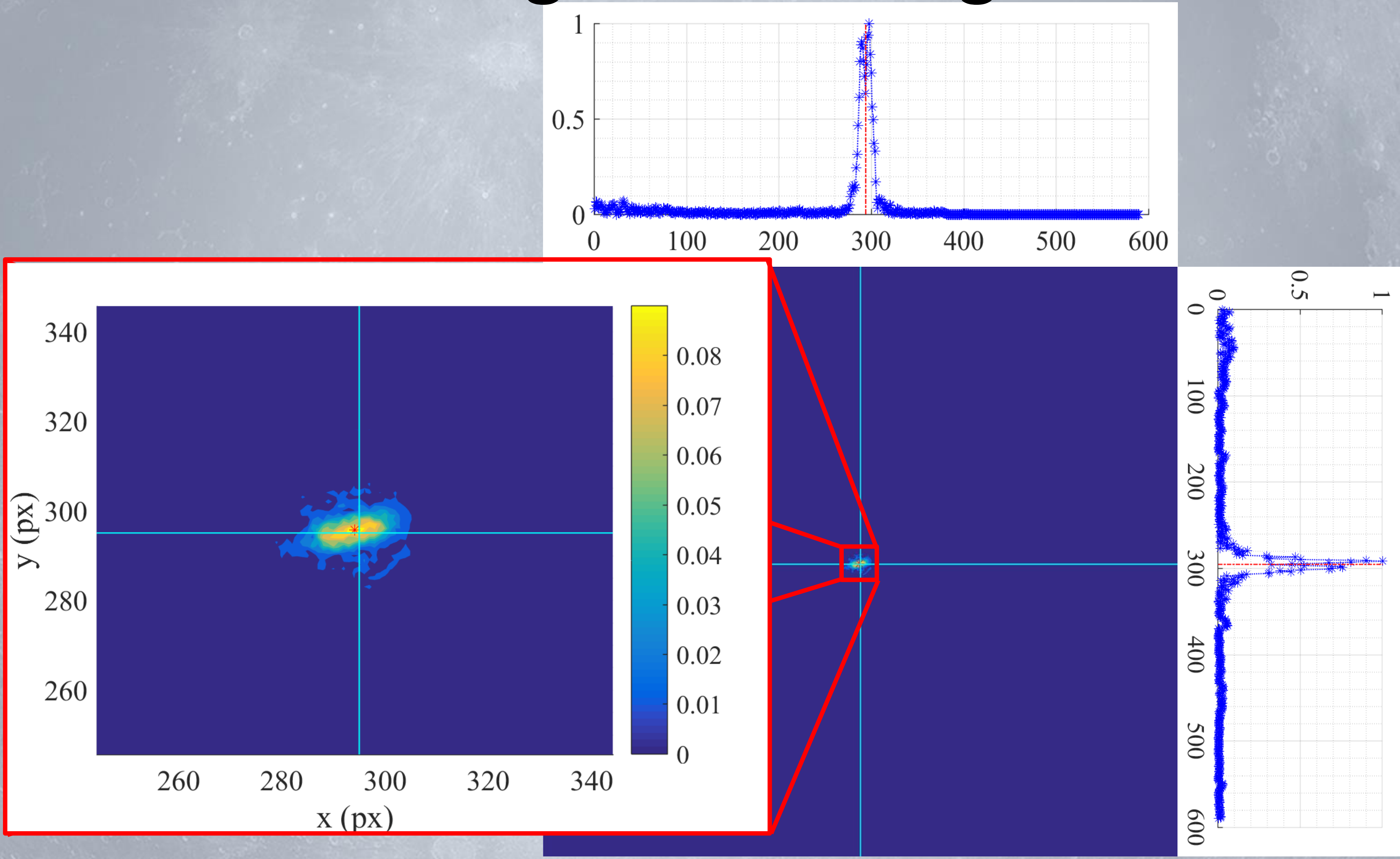
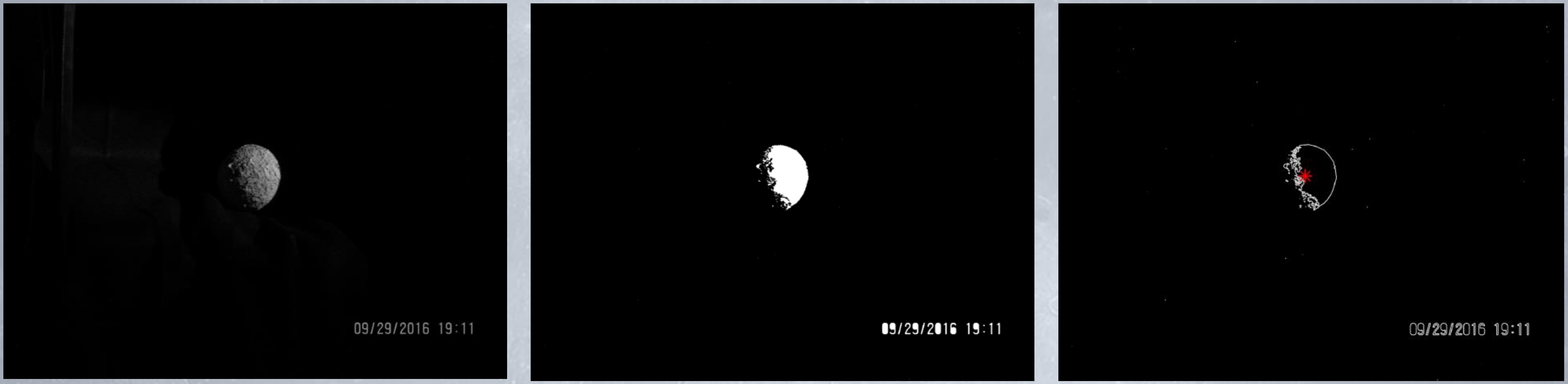


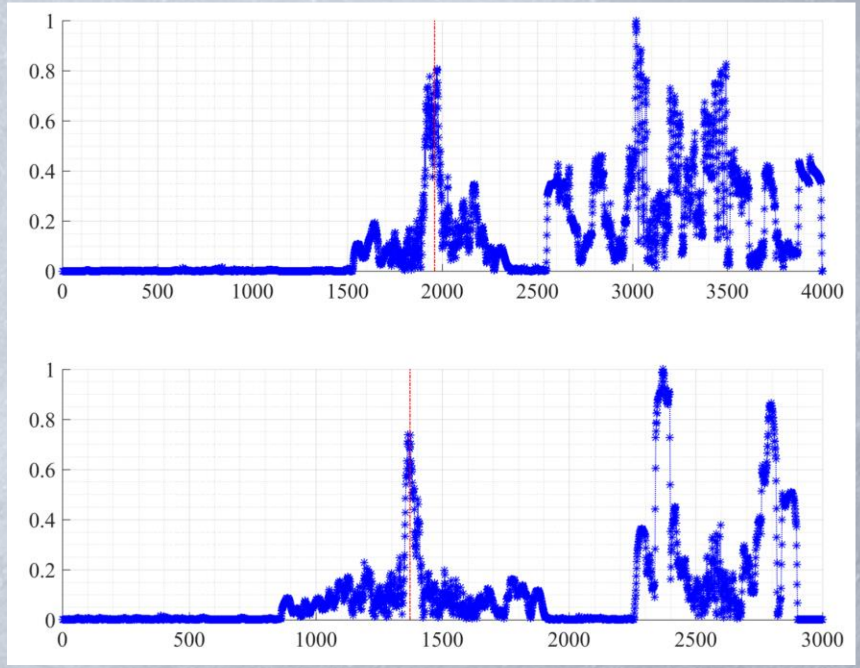
Image Processing



- Rough faux Moon created with plaster

- Coherent Circular Hough Transform robust enough to handle even unrealistic scenarios

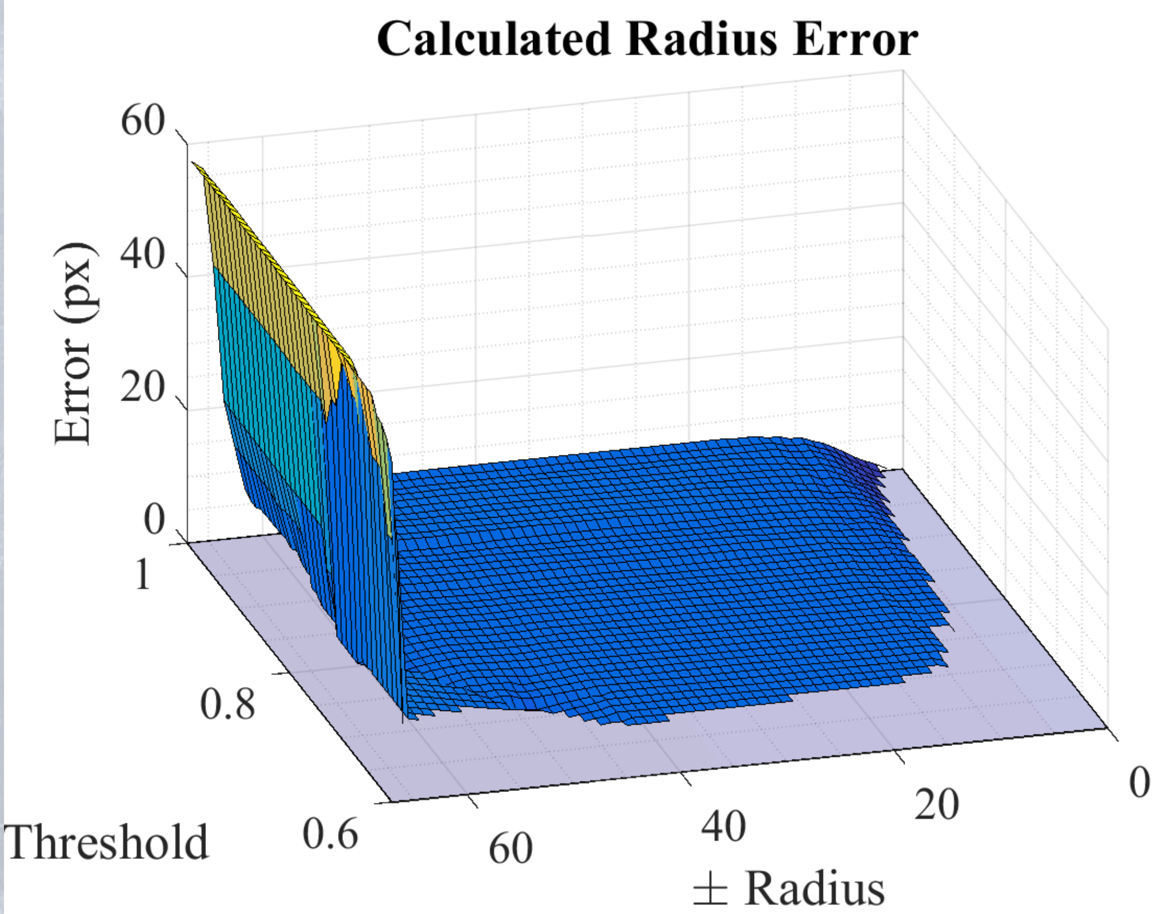
- Timestamp is included for the analysis, with the noise seen in the voting, but the algorithm still finds the circular center



OSPRE Image Processing

Radius Determination Error

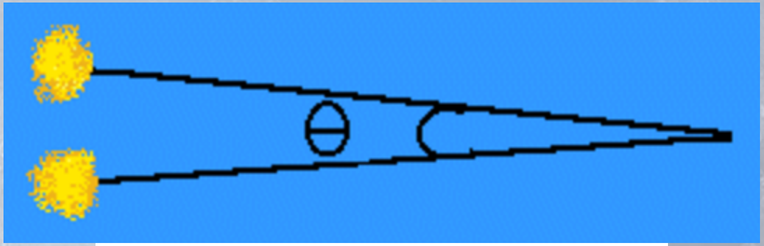
- Majority average of 8 px error in radius
- Spikes as radius estimate range approaches celestial body radius (will not ever be this far off in estimation)
- Decreases to <8 px error, but with increased solution sensitivity

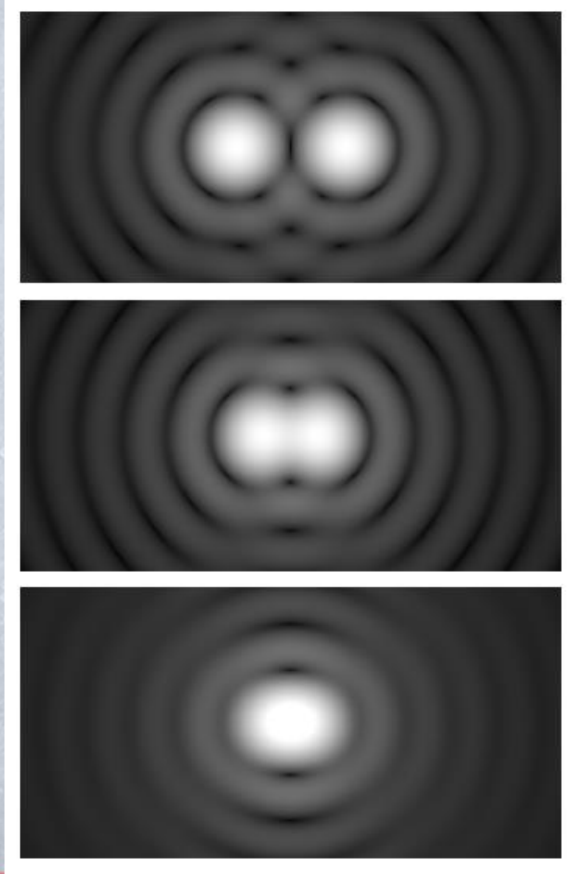


Threshold - voting strength kept, lower = less votes kept, higher = more votes kept
±*Radius* - Plus and minus range of radius to analyze; more information in backup

Diffraction Limit

$$\theta = 1.220\frac{\lambda}{D}$$

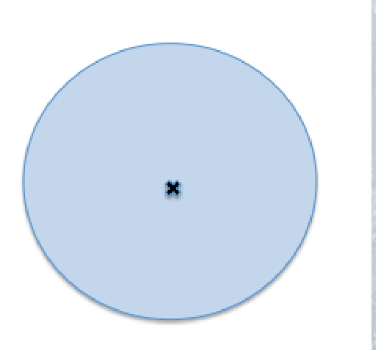




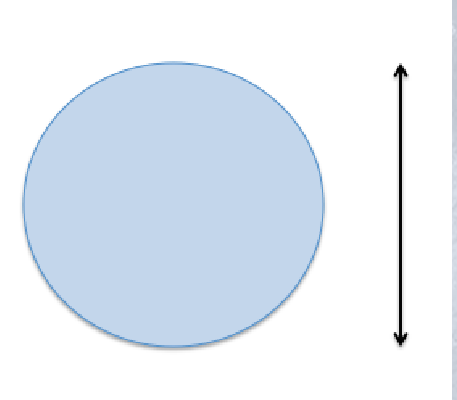
- Degree/Pixel = ~0.0143°
- Diffraction Limit =
  0.01475° (blue)
  0.02766° (red)
- Diffraction limited, but well matched

Image Processing

- Center finding



- Diameter finding

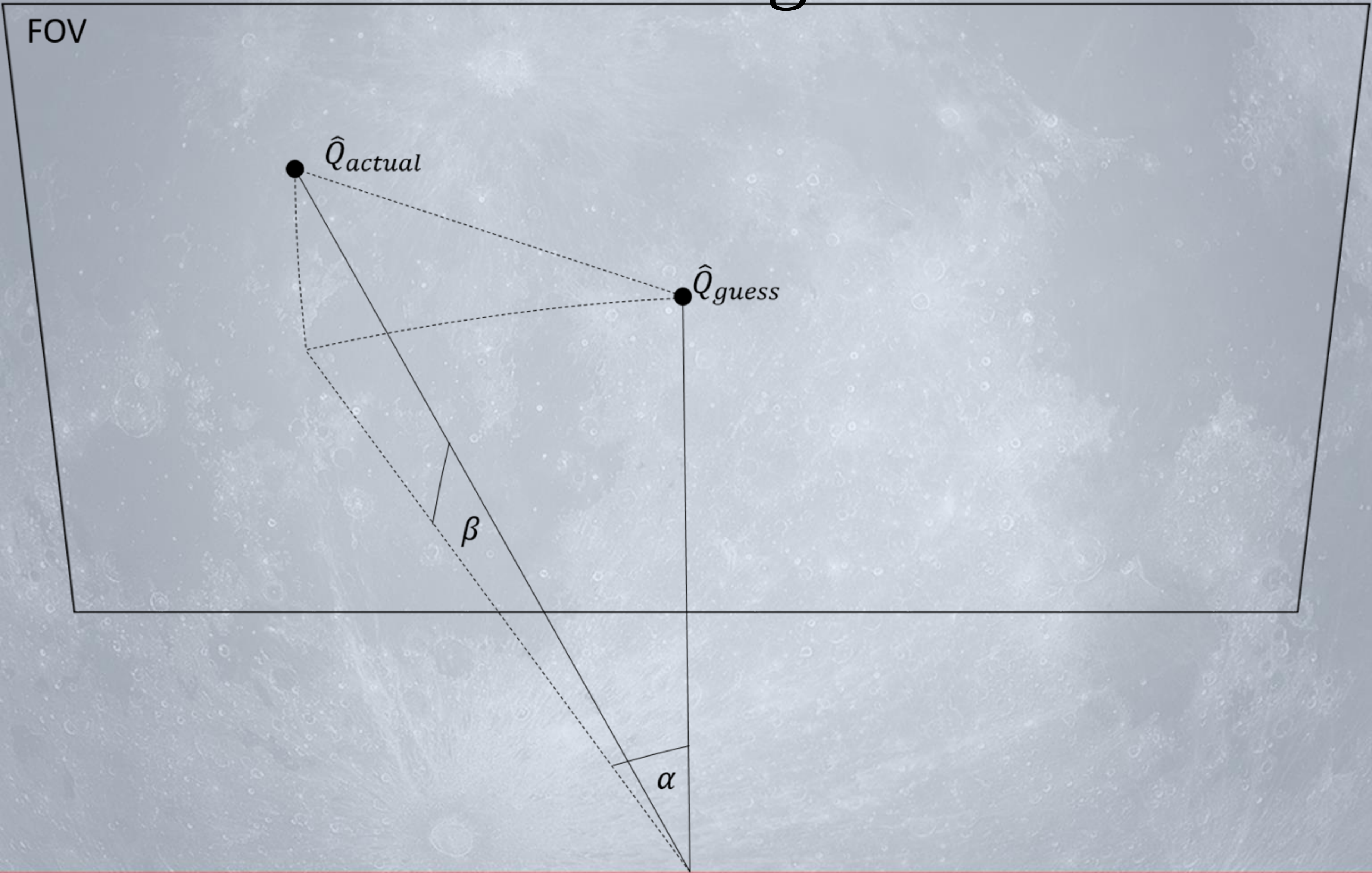


Navigation

- Three position algorithms
  - require images of Earth, Moon, or both
  - most accurate one is chosen at each point in time during flight
- Run through a data filter to increase accuracy and compute velocity

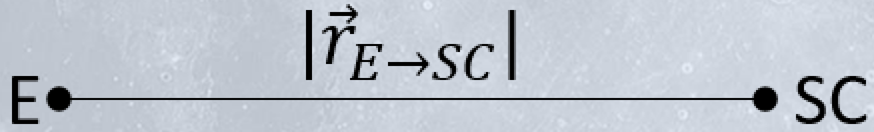
Navigation Backup Slides

Error Angles



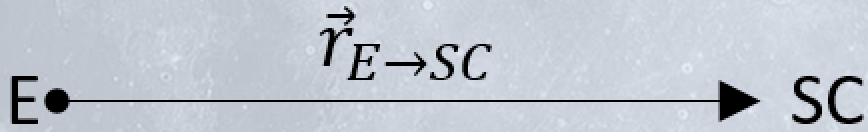
Ranging Method

Ranging Method



Ranging Method

Ranging Method

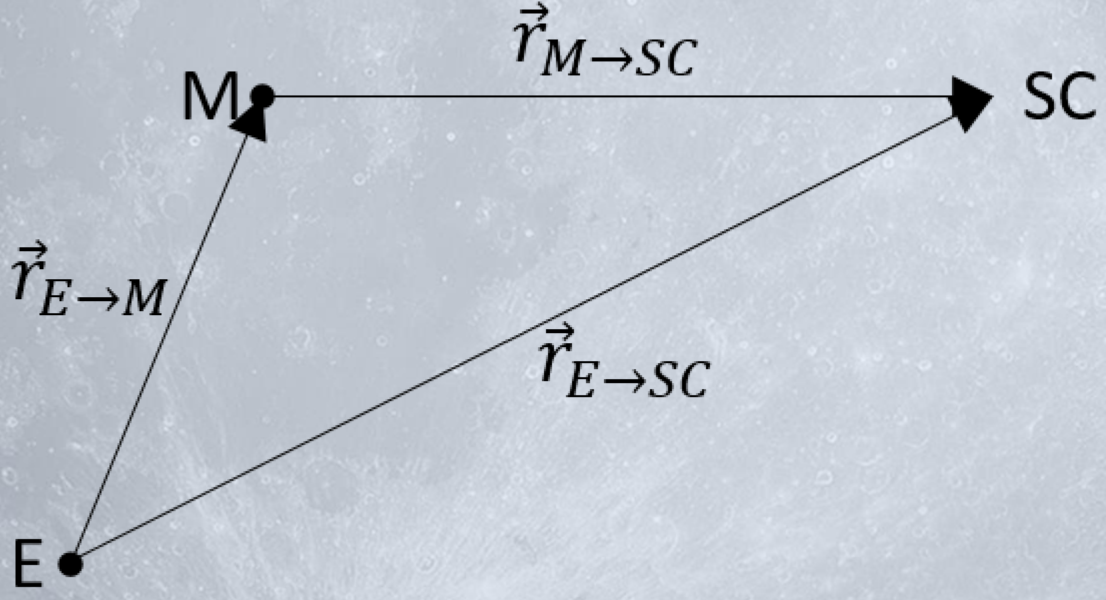


$\vec{r}_{E\rightarrow SC}$

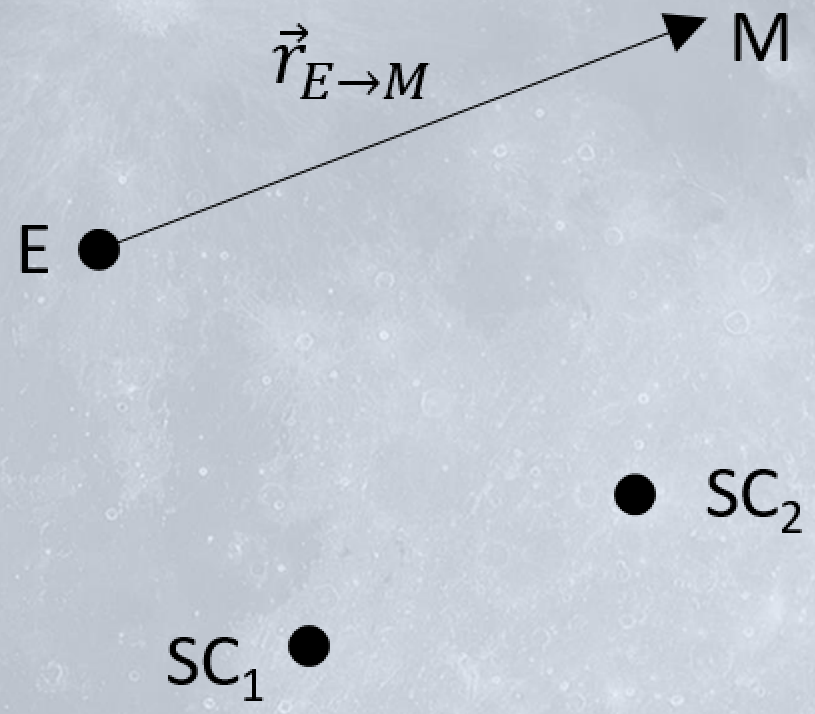
E

SC

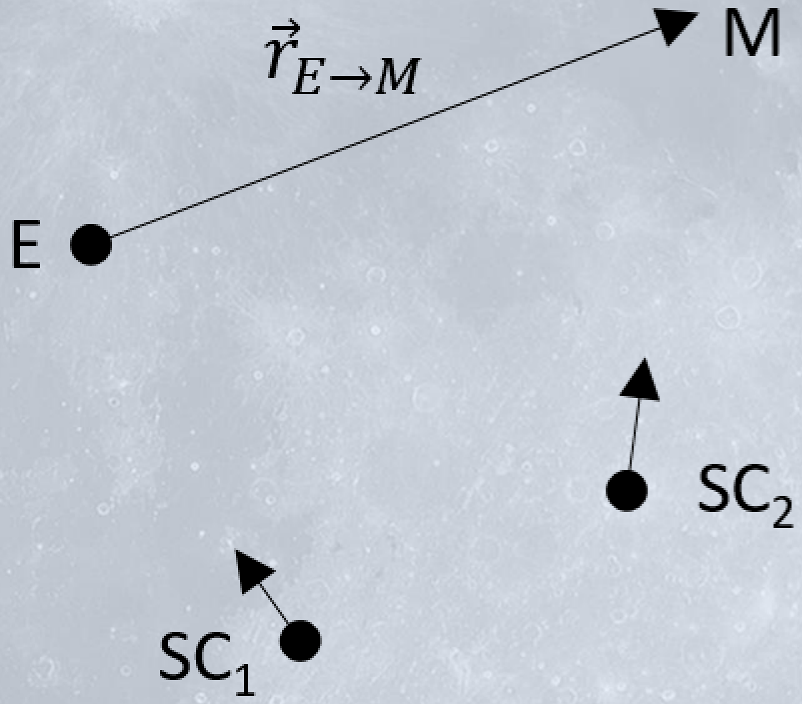
Ranging Method



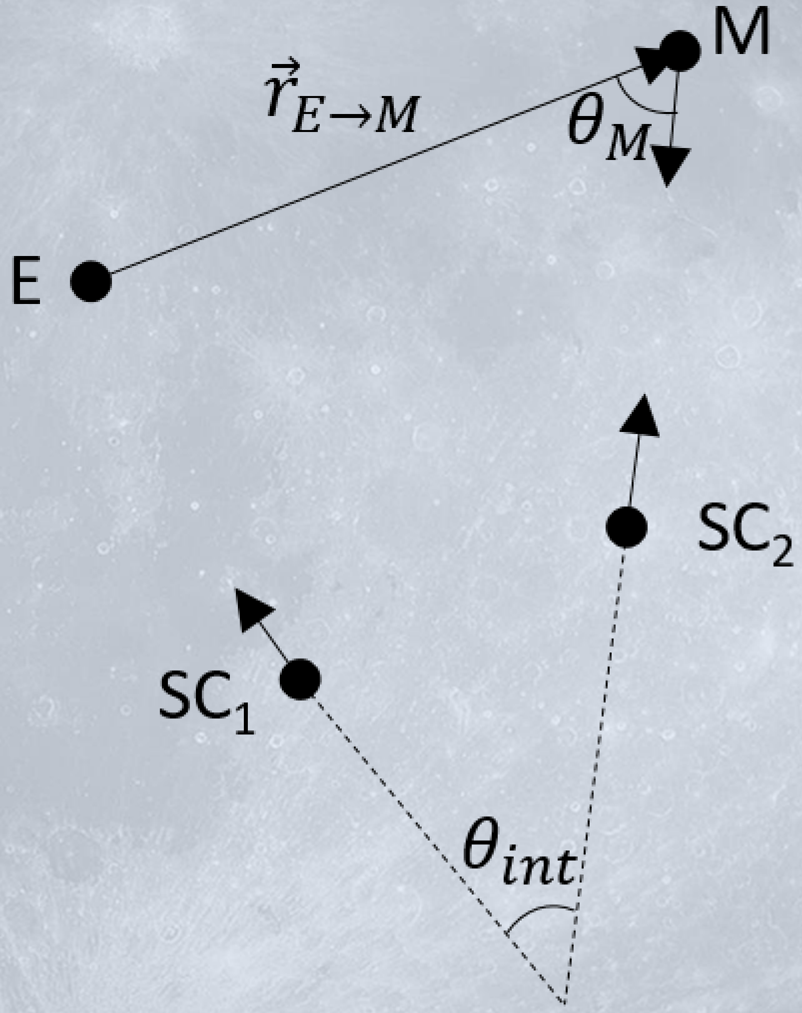
Angles Method



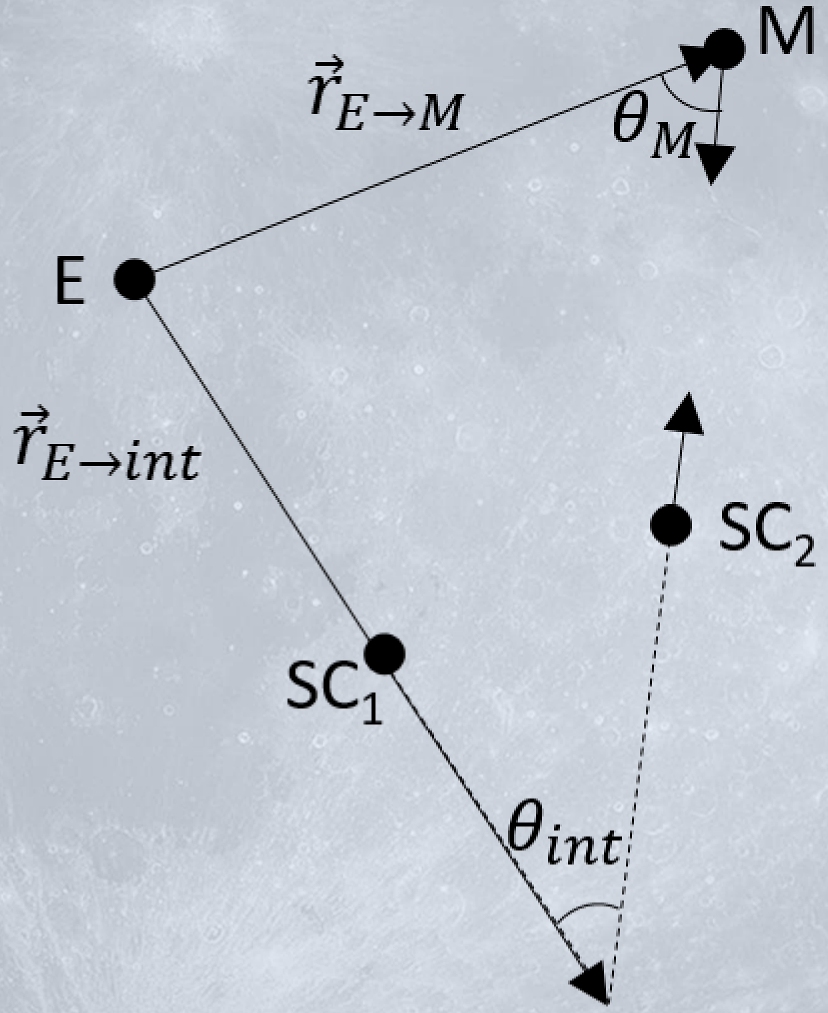
Angles Method



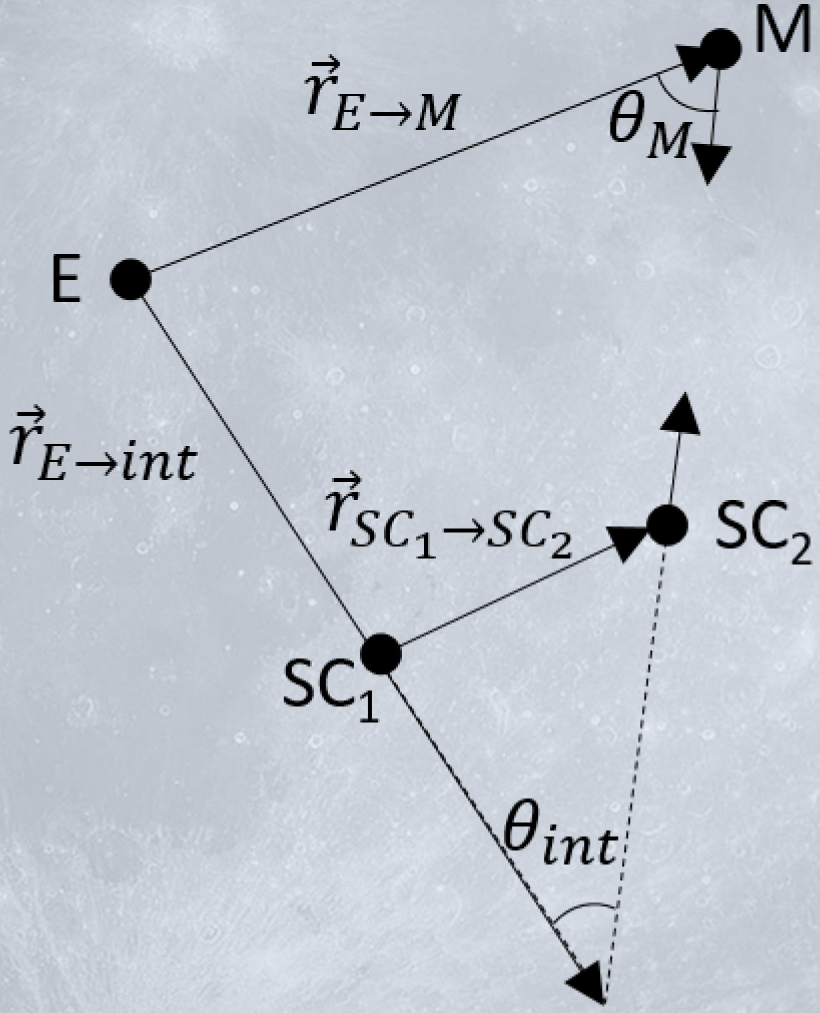
Angles Method



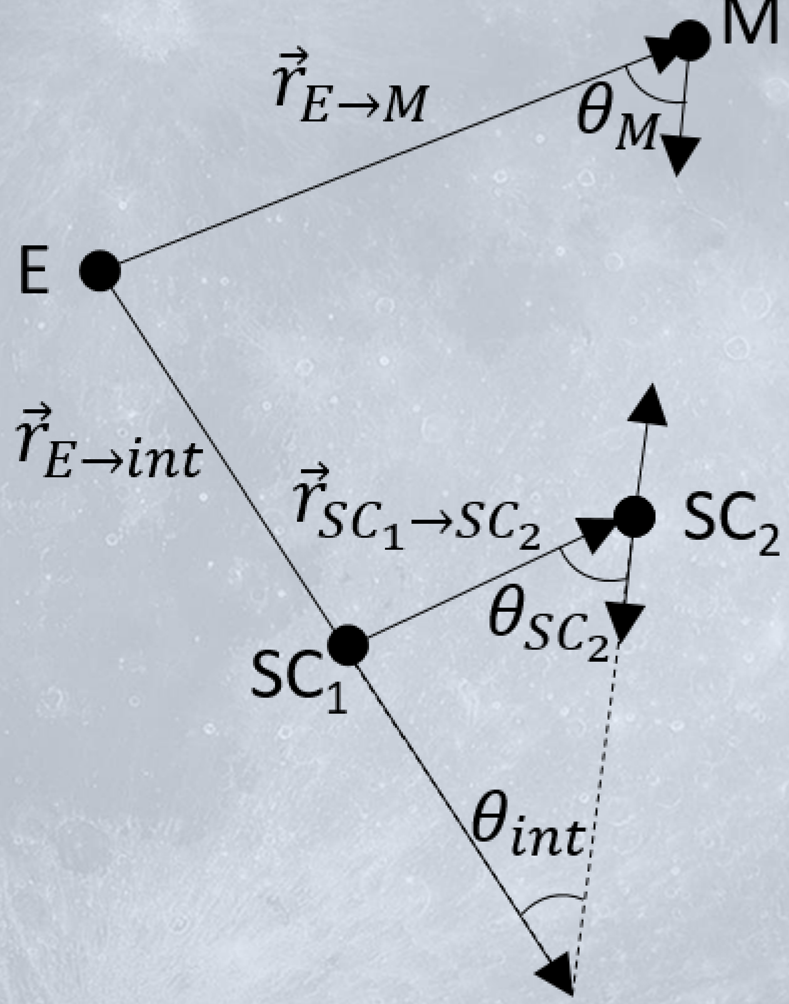
Angles Method



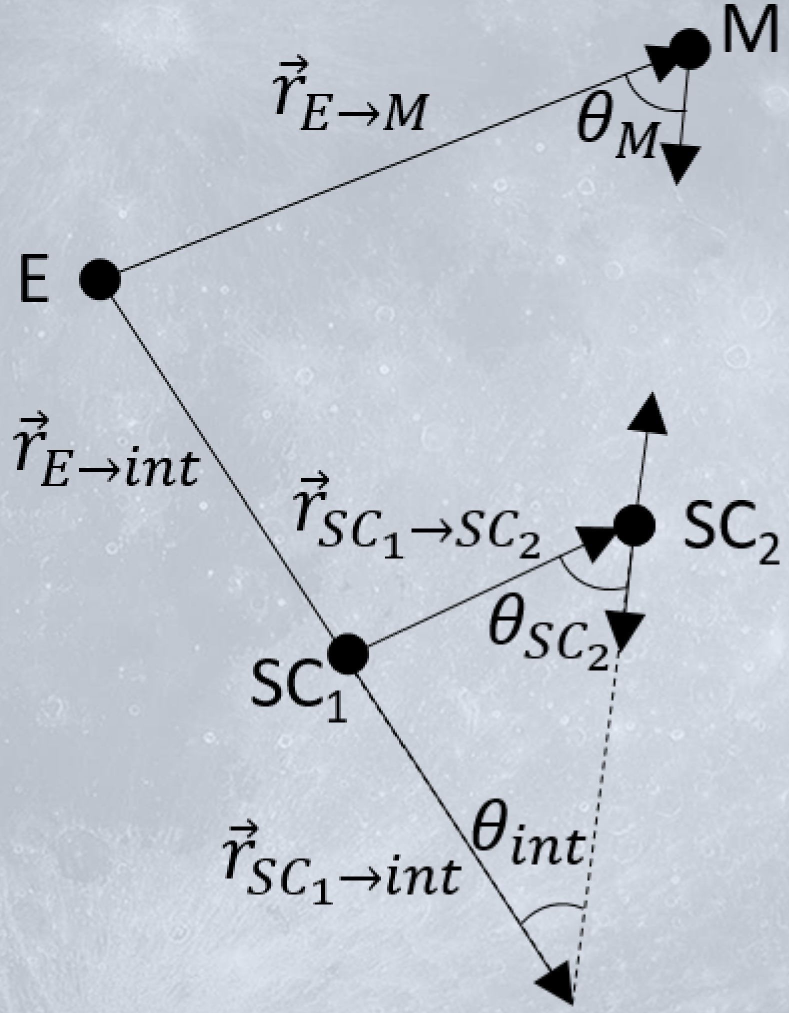
Angles Method



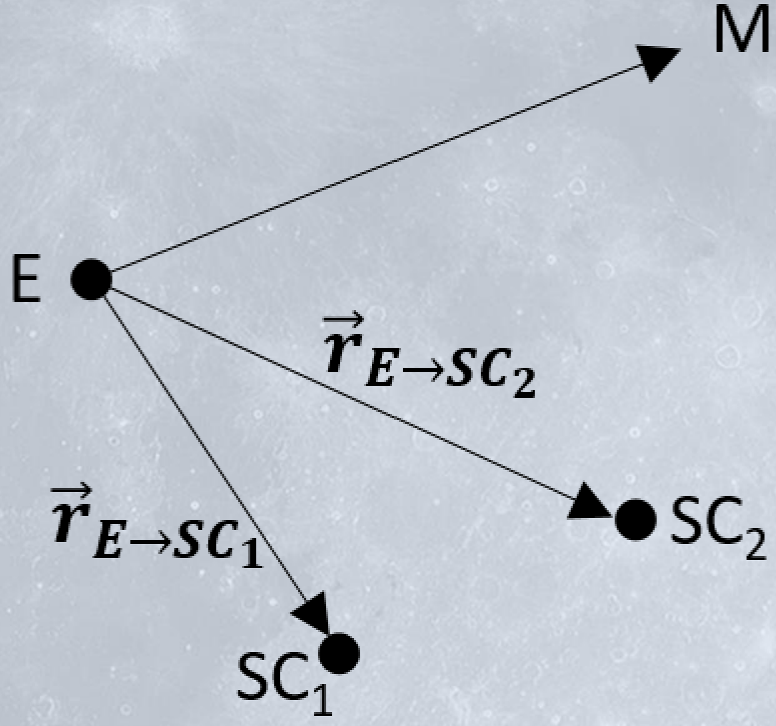
Angles Method



Angles Method



Angles Method



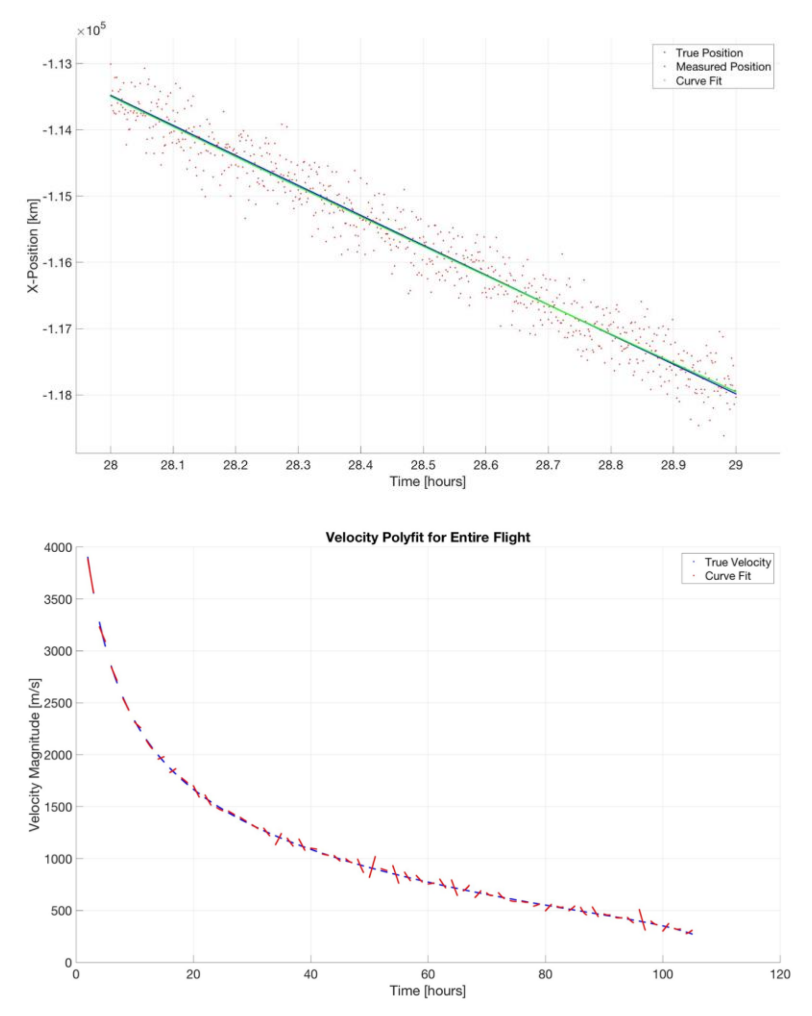
Velocity Algorithm - Curve Fitting Method

Technique:

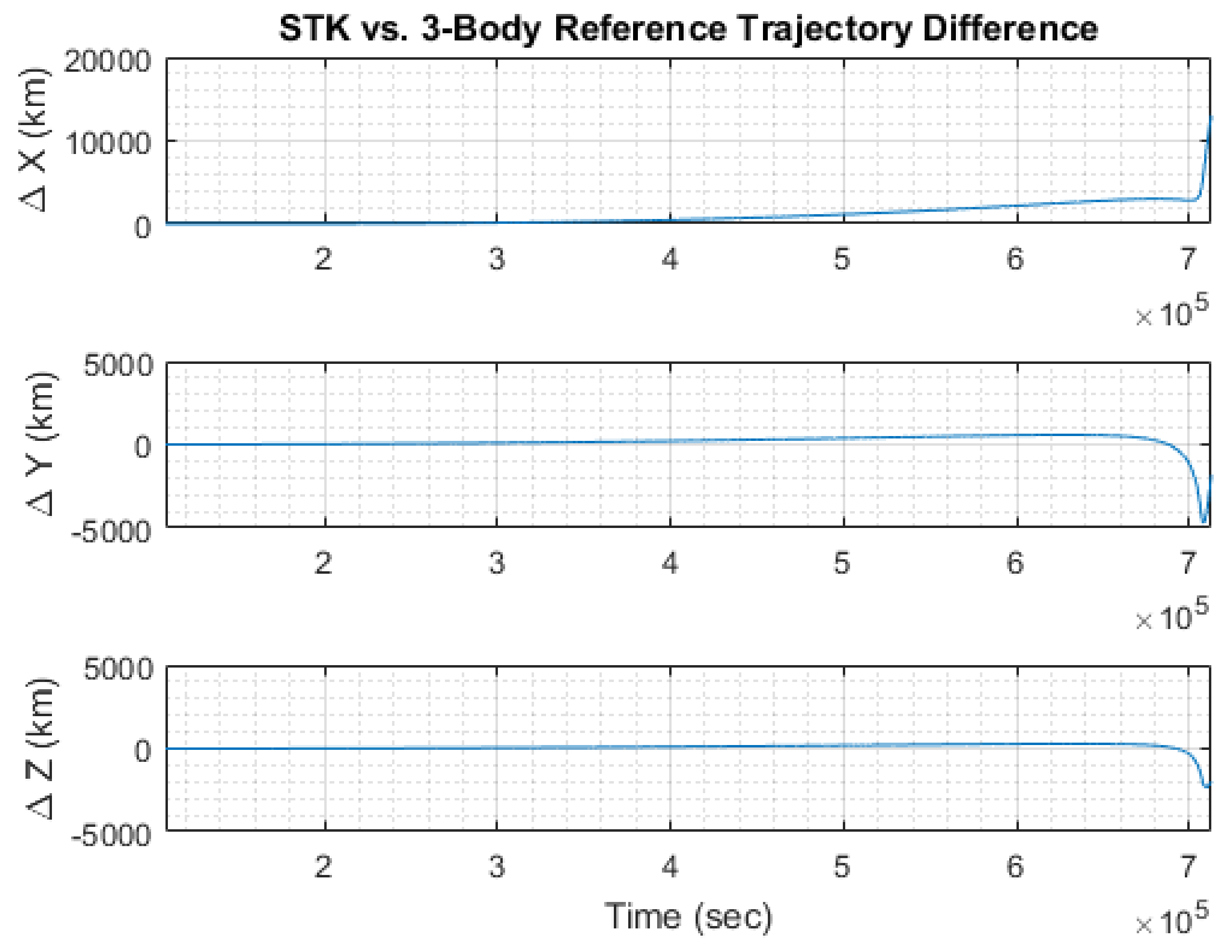
- Curve fit the X, Y, & Z position data individually
- Differentiating each of the position curves individually provides X, Y, & Z velocity data

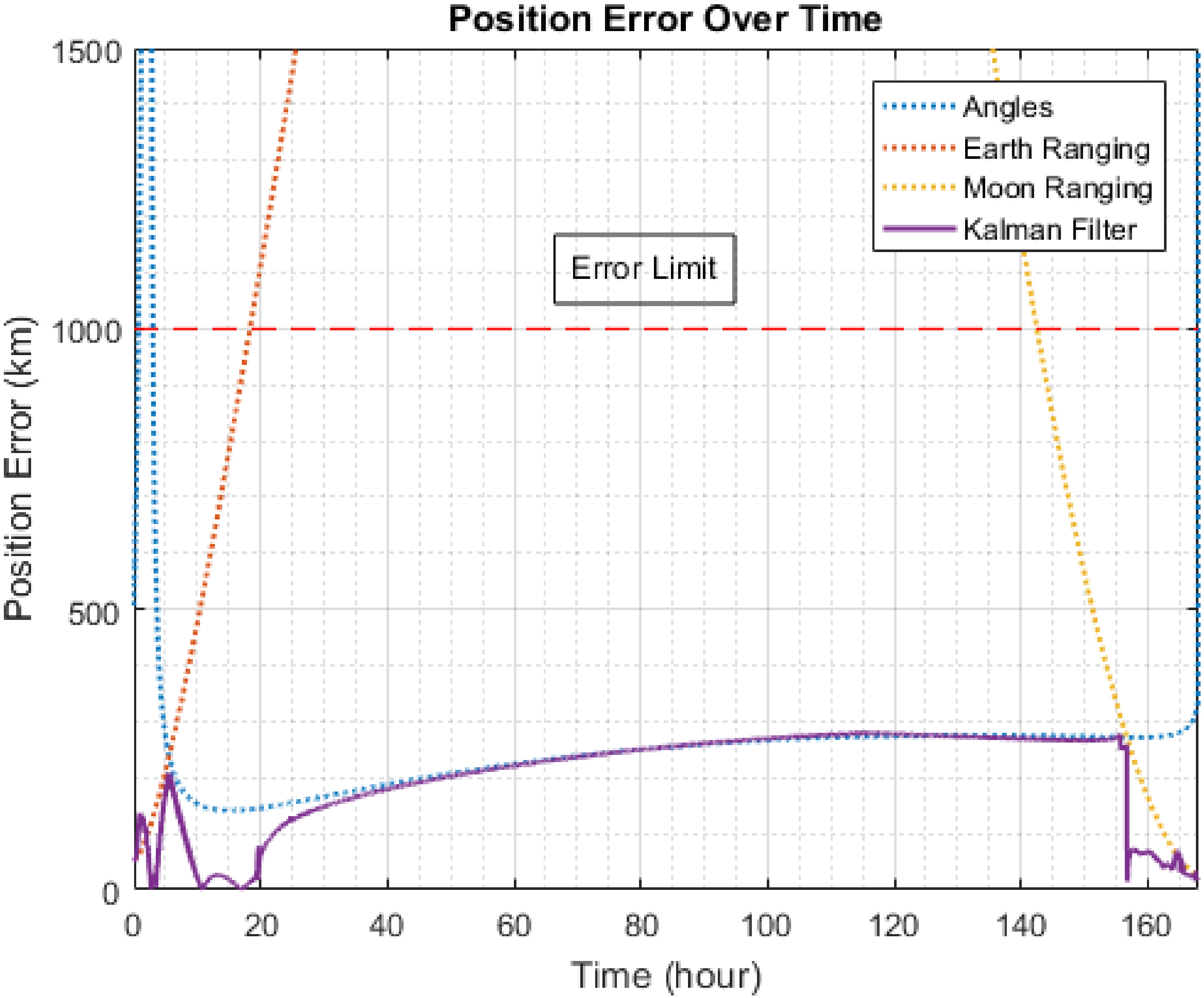
Drawbacks:

- Reports velocity data for the previous hour once at the end of each hour
- Need a lot of data points with small timesteps in between to be accurate



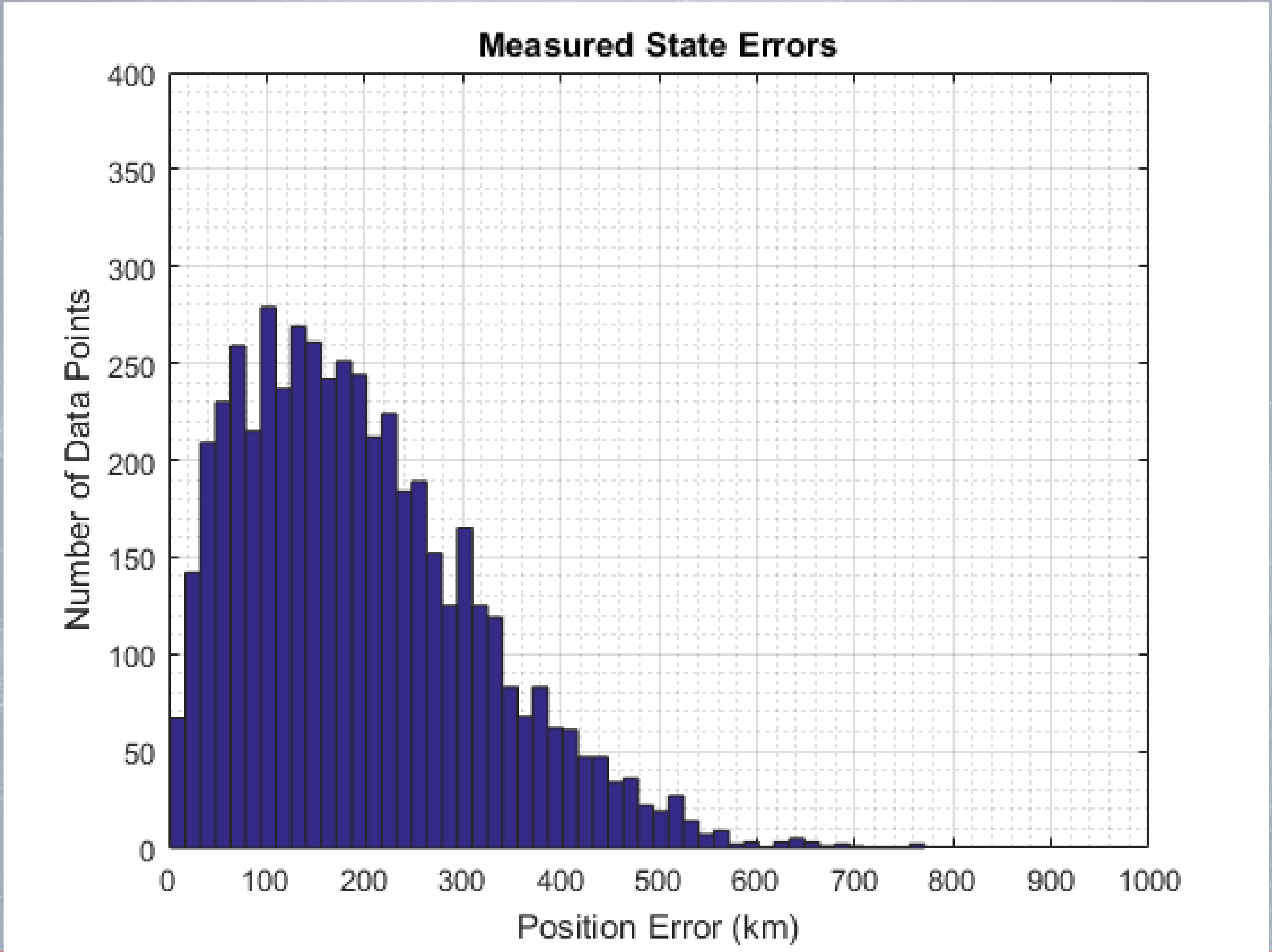
STK vs. 3-Body Trajectory




Position Error Over Time
1500
1000
500
0
Position Error (km)
0
20
40
60
80
100
120
140
160
Time (hour)
Angles
Earth Ranging
Moon Ranging
Kalman Filter
Error Limit

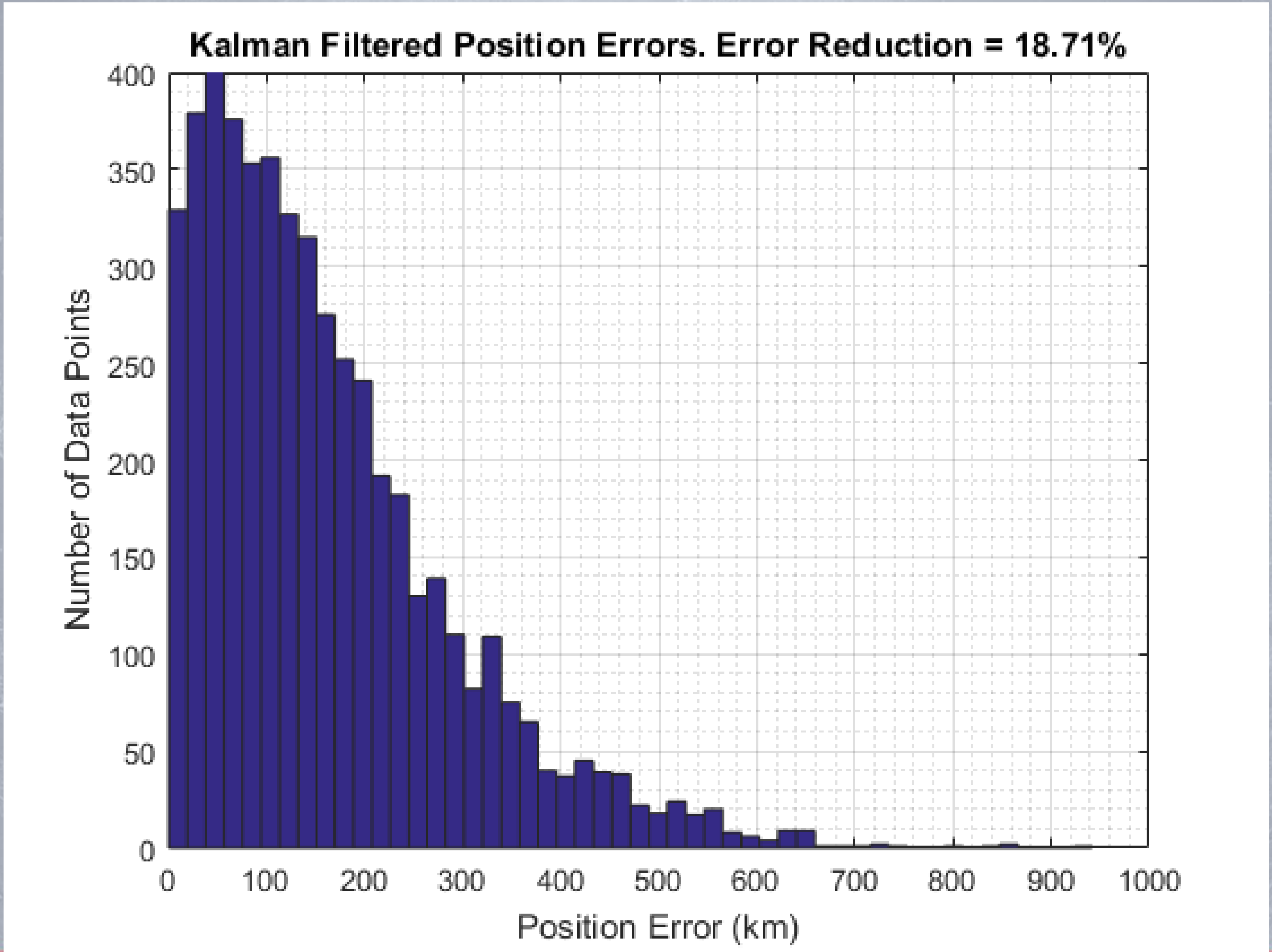
OSPRE Navigation

Position Error Without Kalman Filter



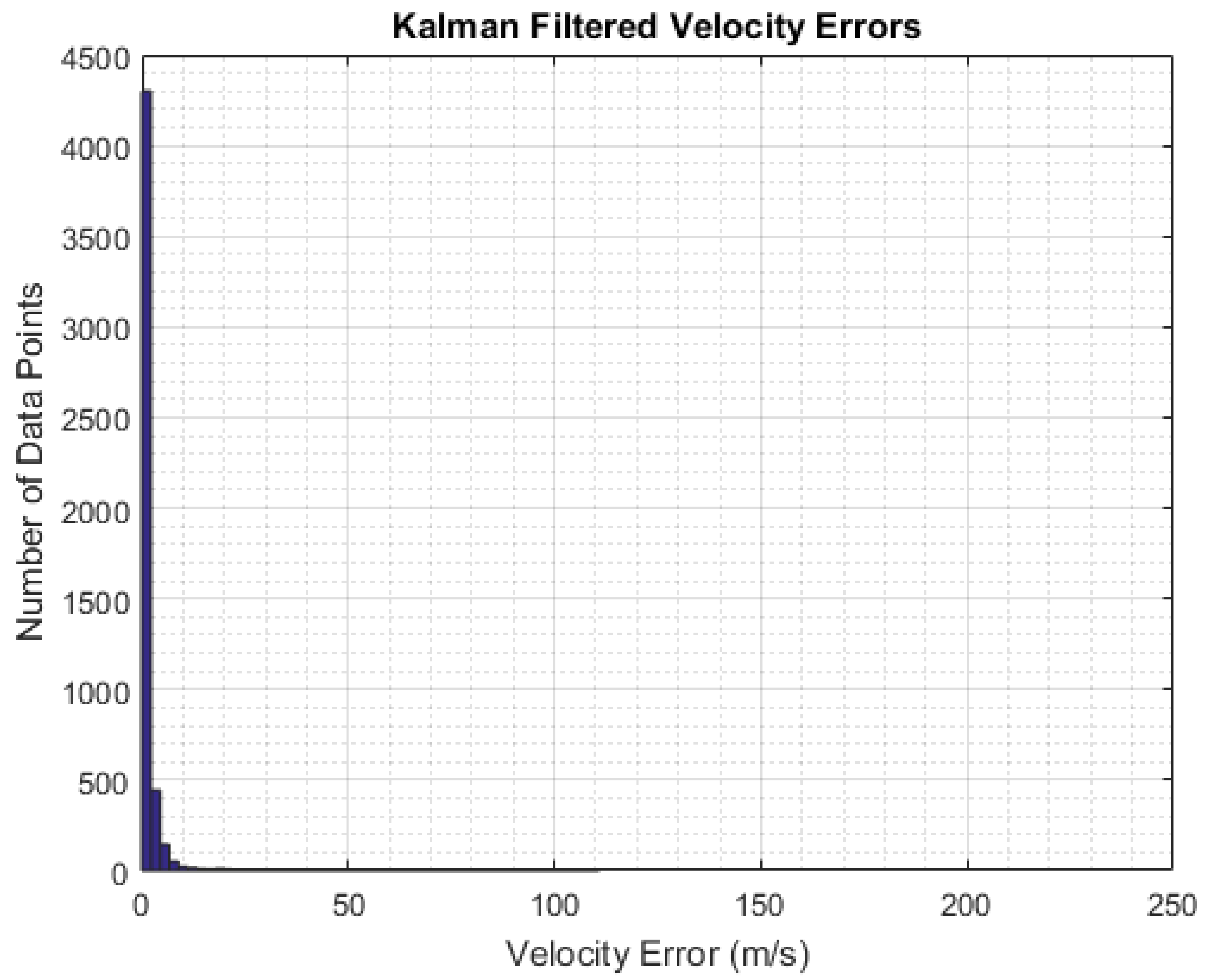
OSPRE Navigation

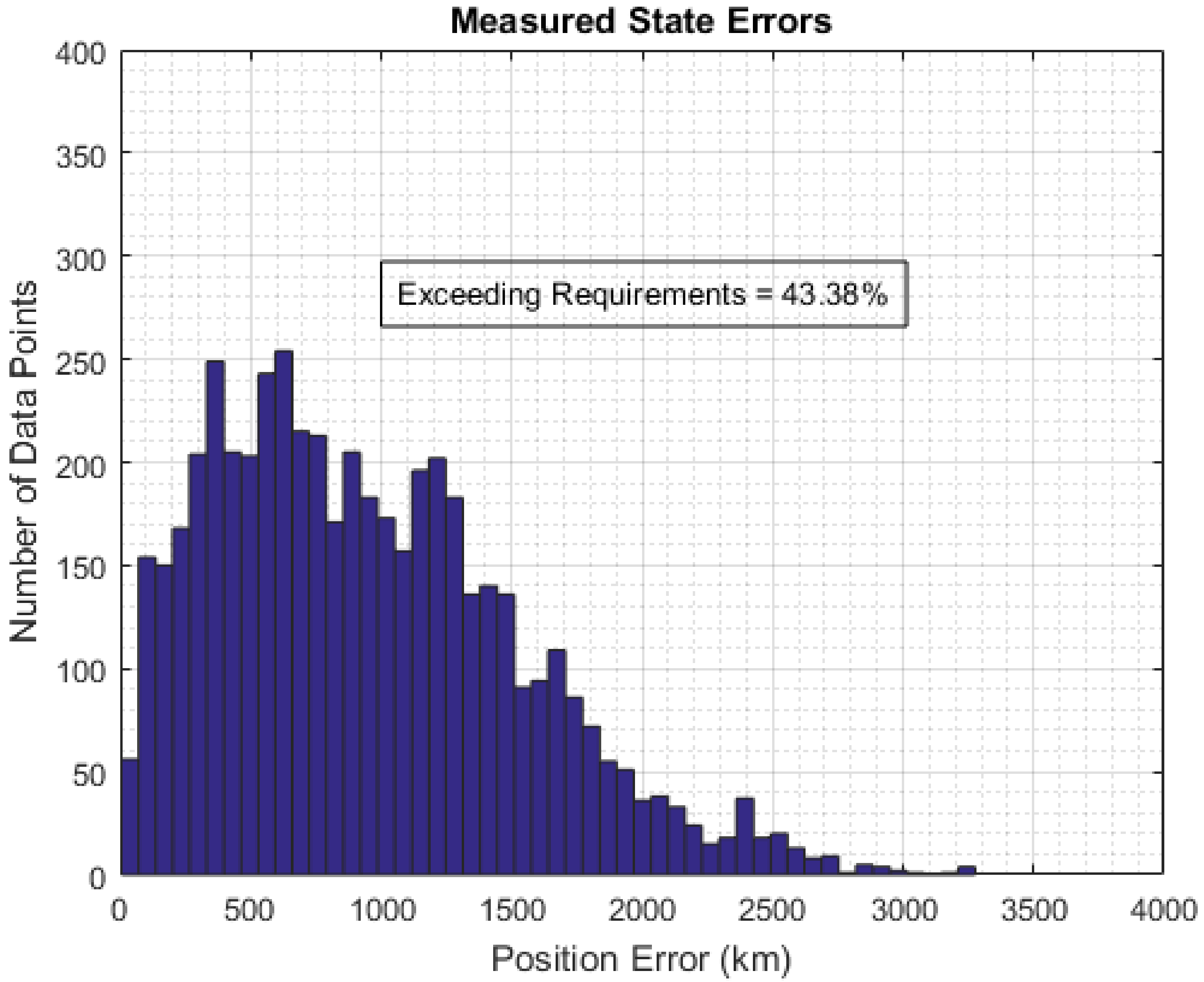
Position Error With Kalman Filter

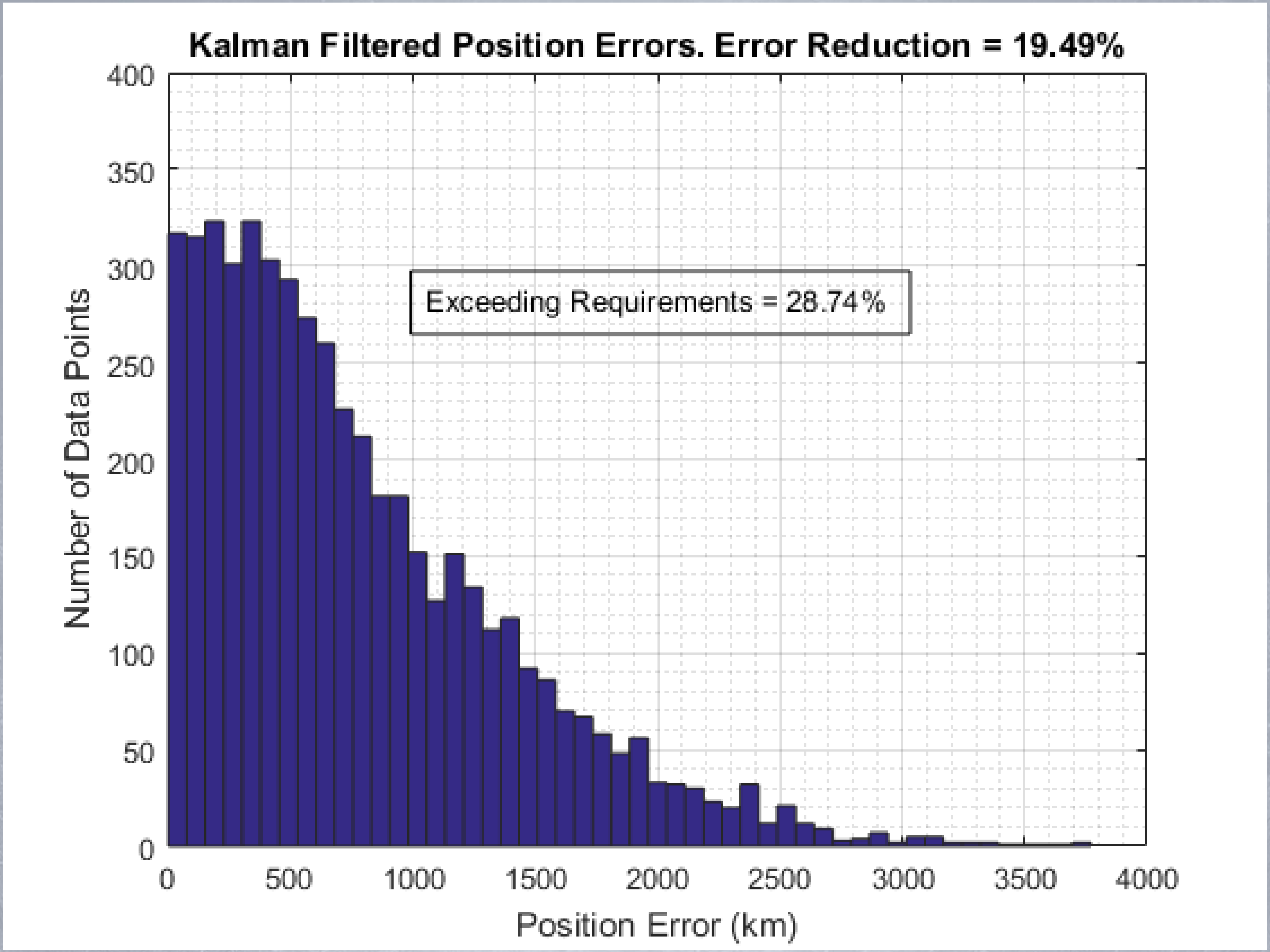


OSPRE Navigation

Kalman Filter Velocity Error

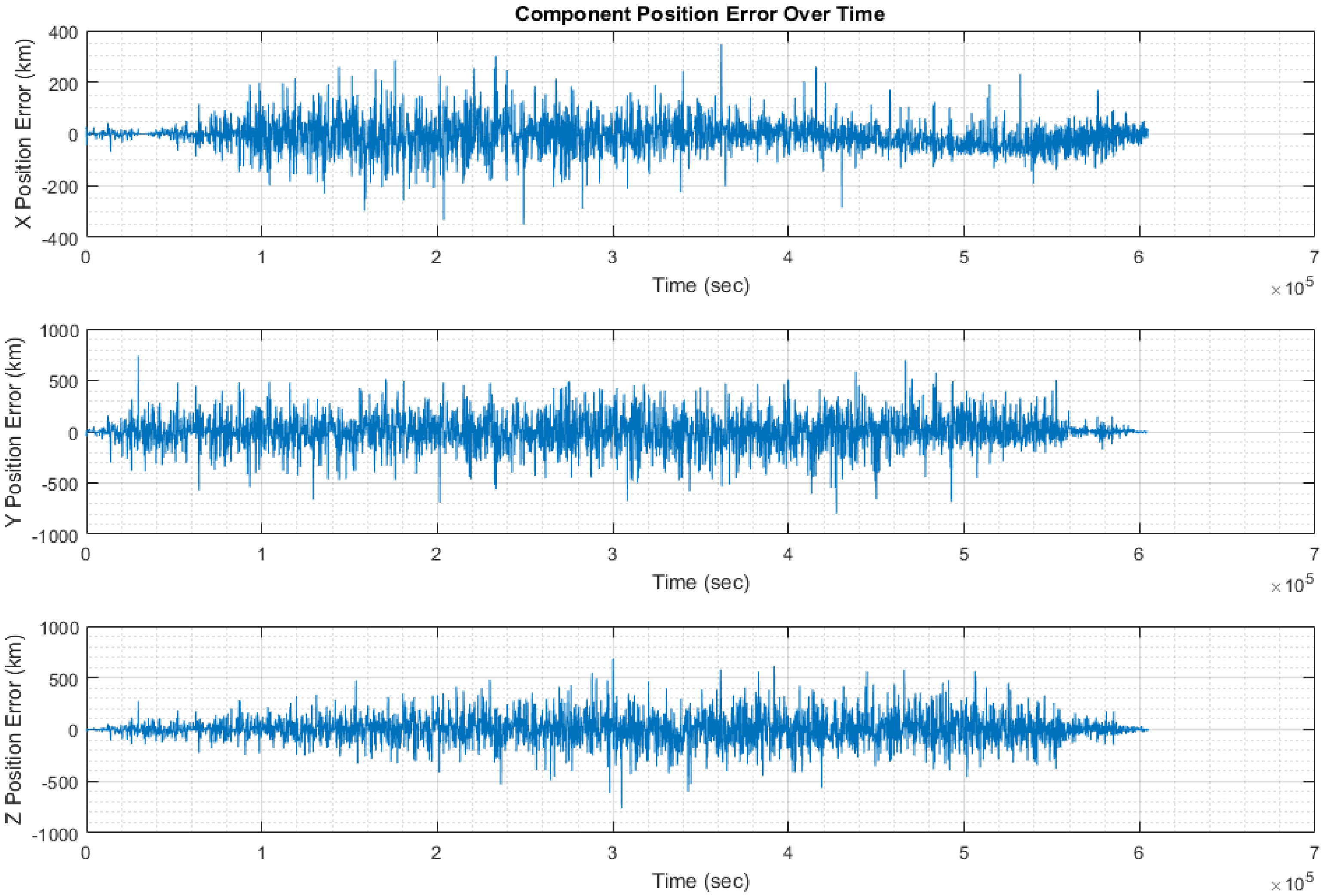



Measured State Errors
Exceeding Requirements = 43.38%
Number of Data Points
400
350
300
250
200
150
100
50
0
0
500
1000
1500
2000
2500
3000
3500
4000
Position Error (km)


Kalman Filtered Position Errors. Error Reduction = 19.49%
Exceeding Requirements = 28.74%
Number of Data Points
0
50
100
150
200
250
300
350
400
0
500
1000
1500
2000
2500
3000
3500
4000
Position Error (km)

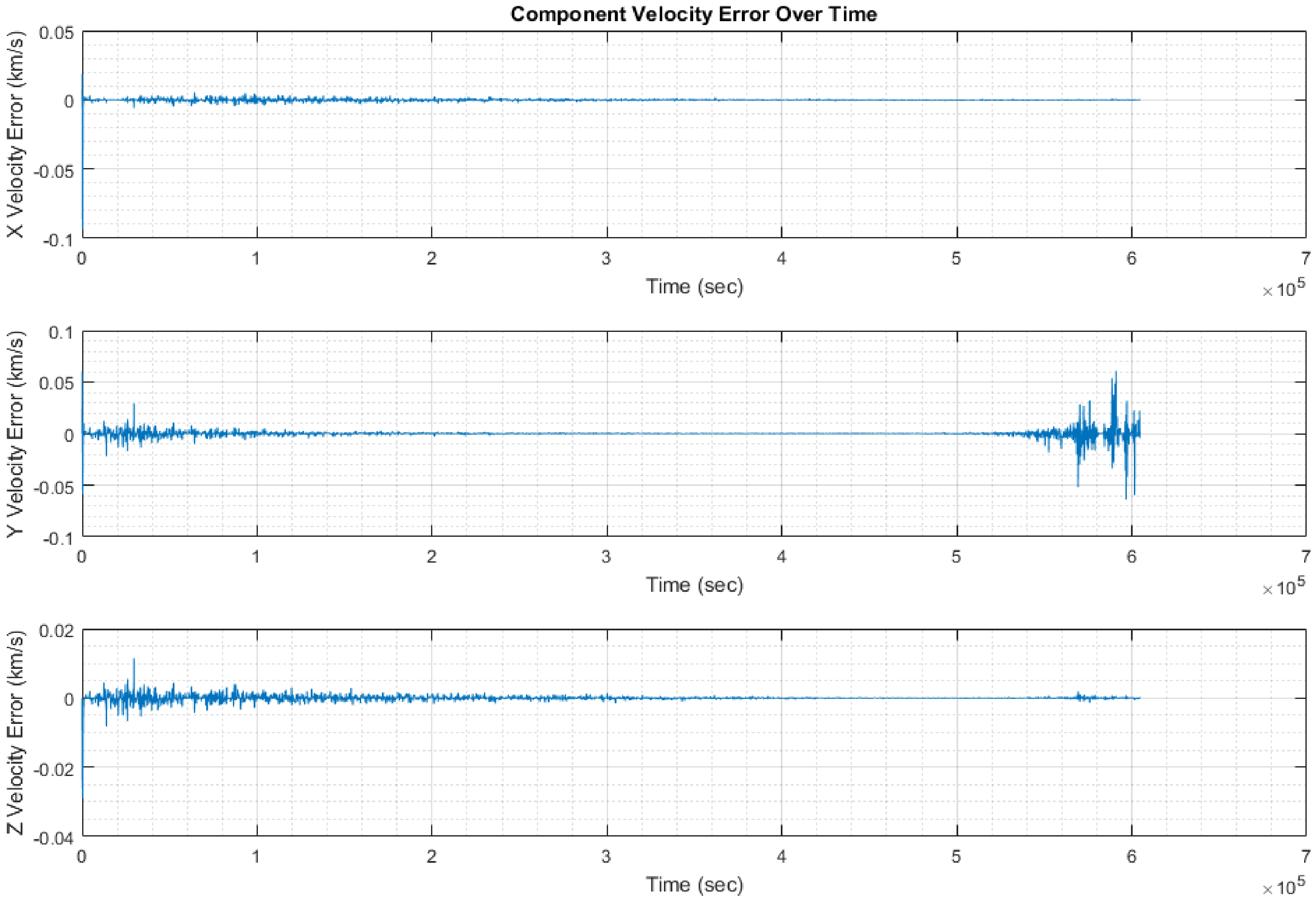
OSPRE Navigation

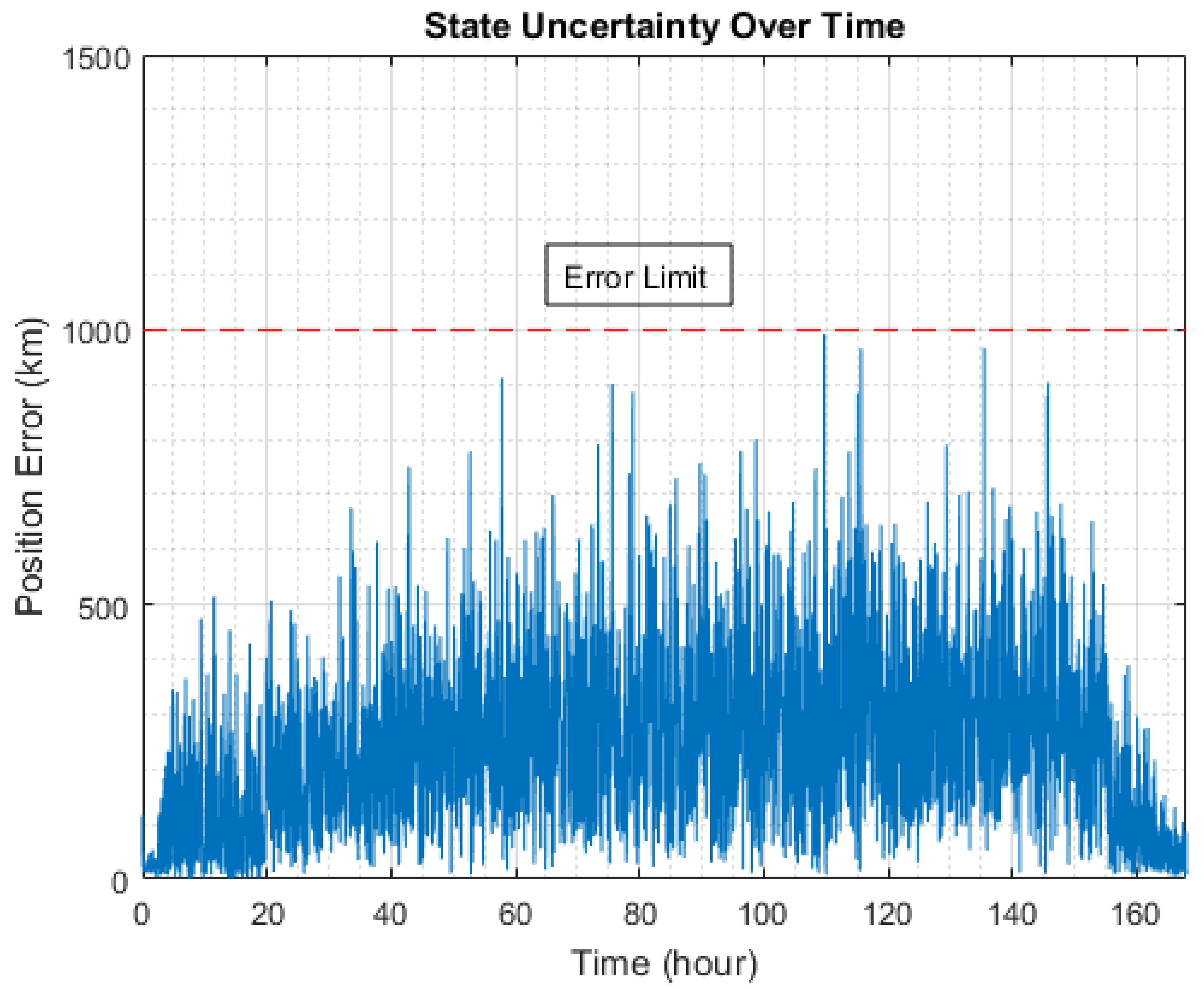
Kalman Filter Position Error

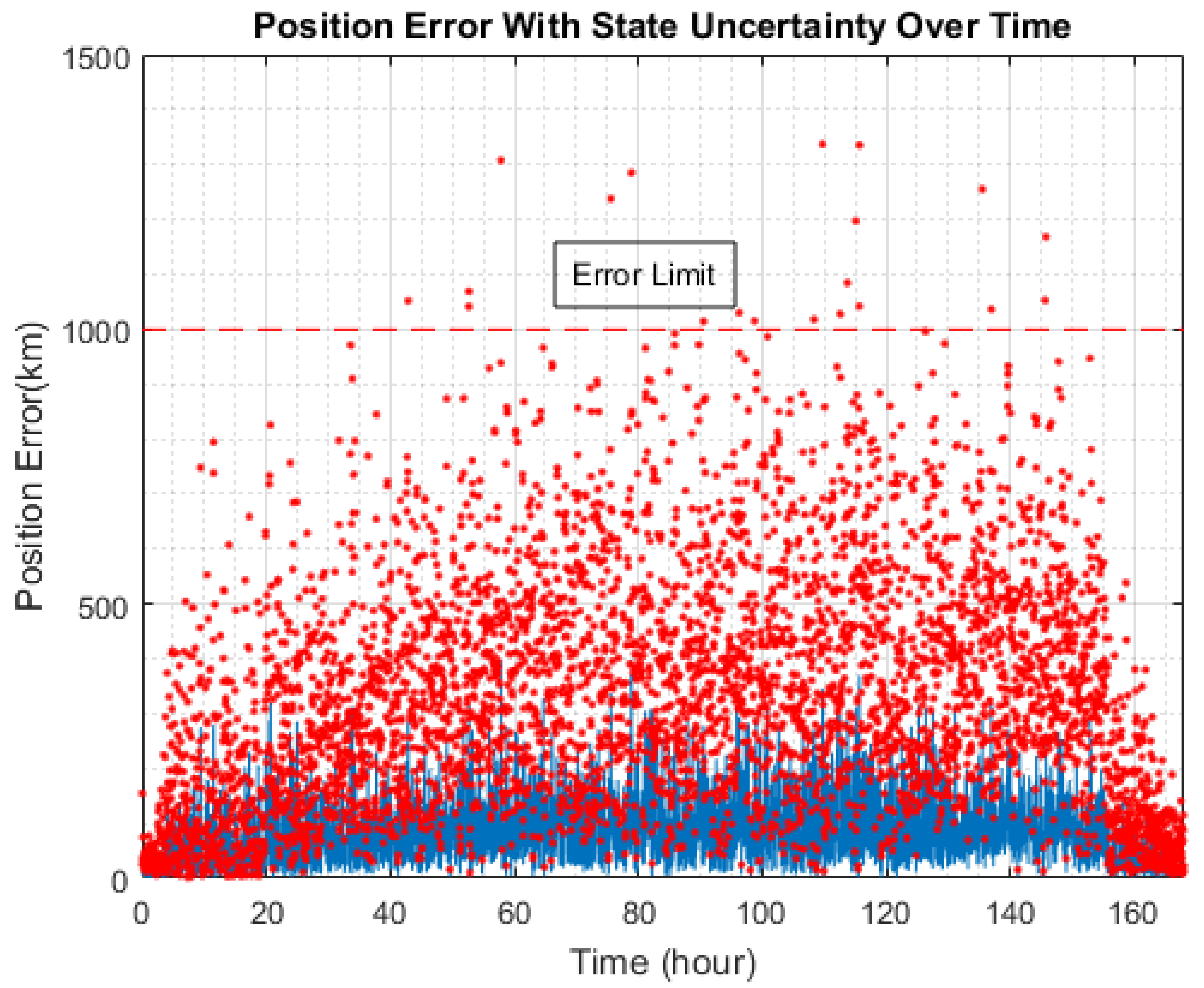


OSPRE Navigation

Kalman Filter Velocity Error

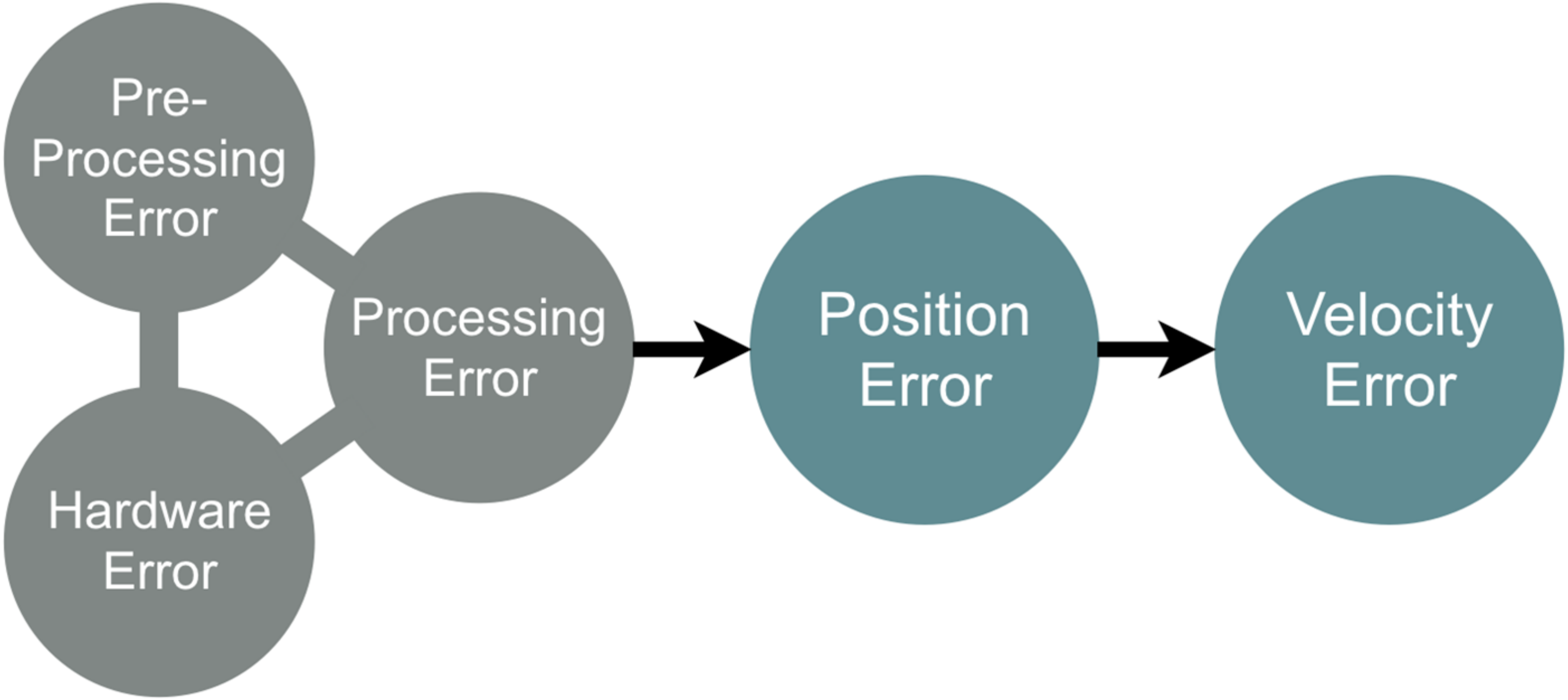



State Uncertainty Over Time
Error Limit
Position Error (km)
1500
1000
500
0
0
20
40
60
80
100
120
140
160
Time (hour)


Position Error With State Uncertainty Over Time
Error Limit
Position Error(km)
1500
1000
500
0
0
20
40
60
80
100
120
140
160
Time (hour)

TESTING BACKUP SLIDES

Test Scope



Pre-Processing Error

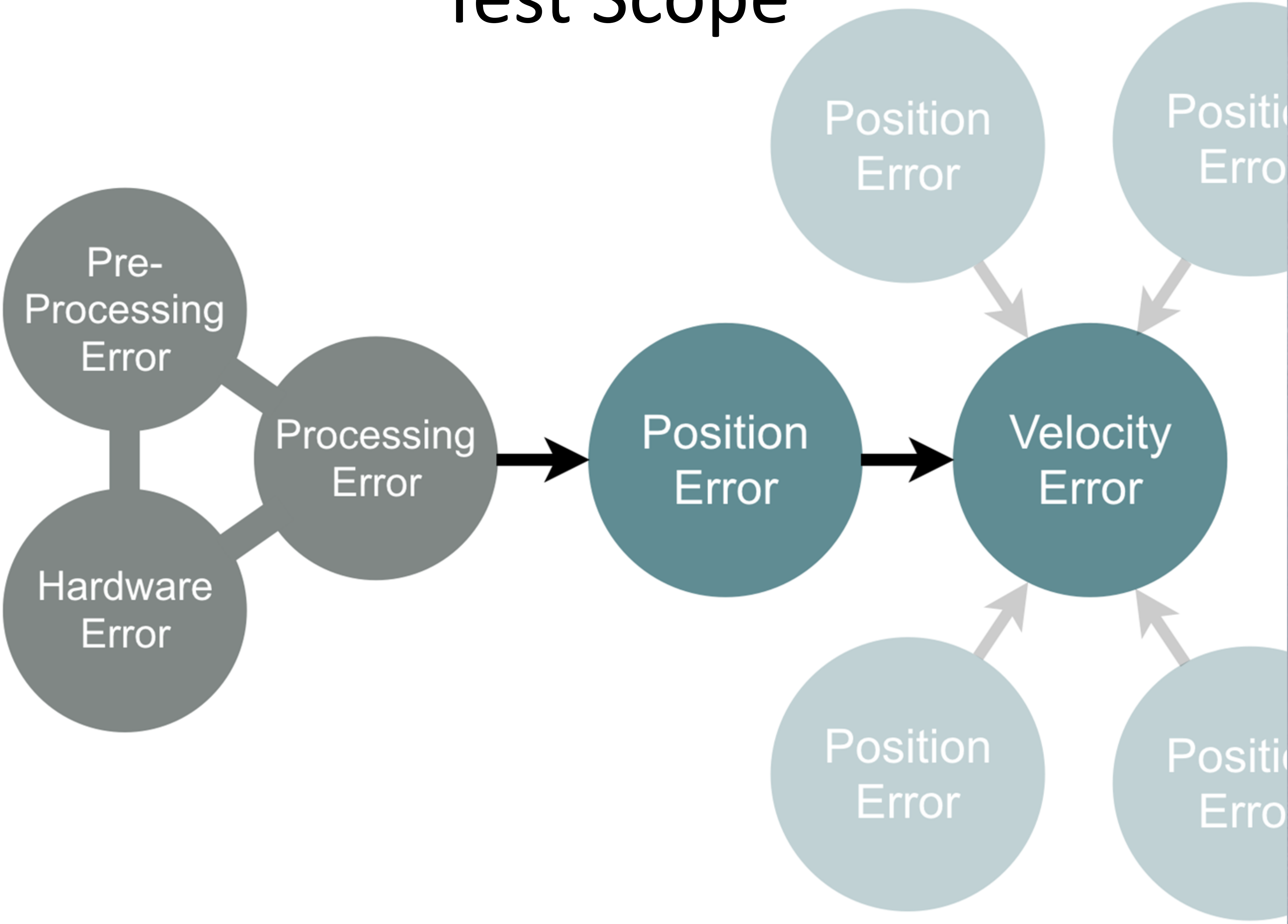
Hardware Error

Processing Error

Position Error

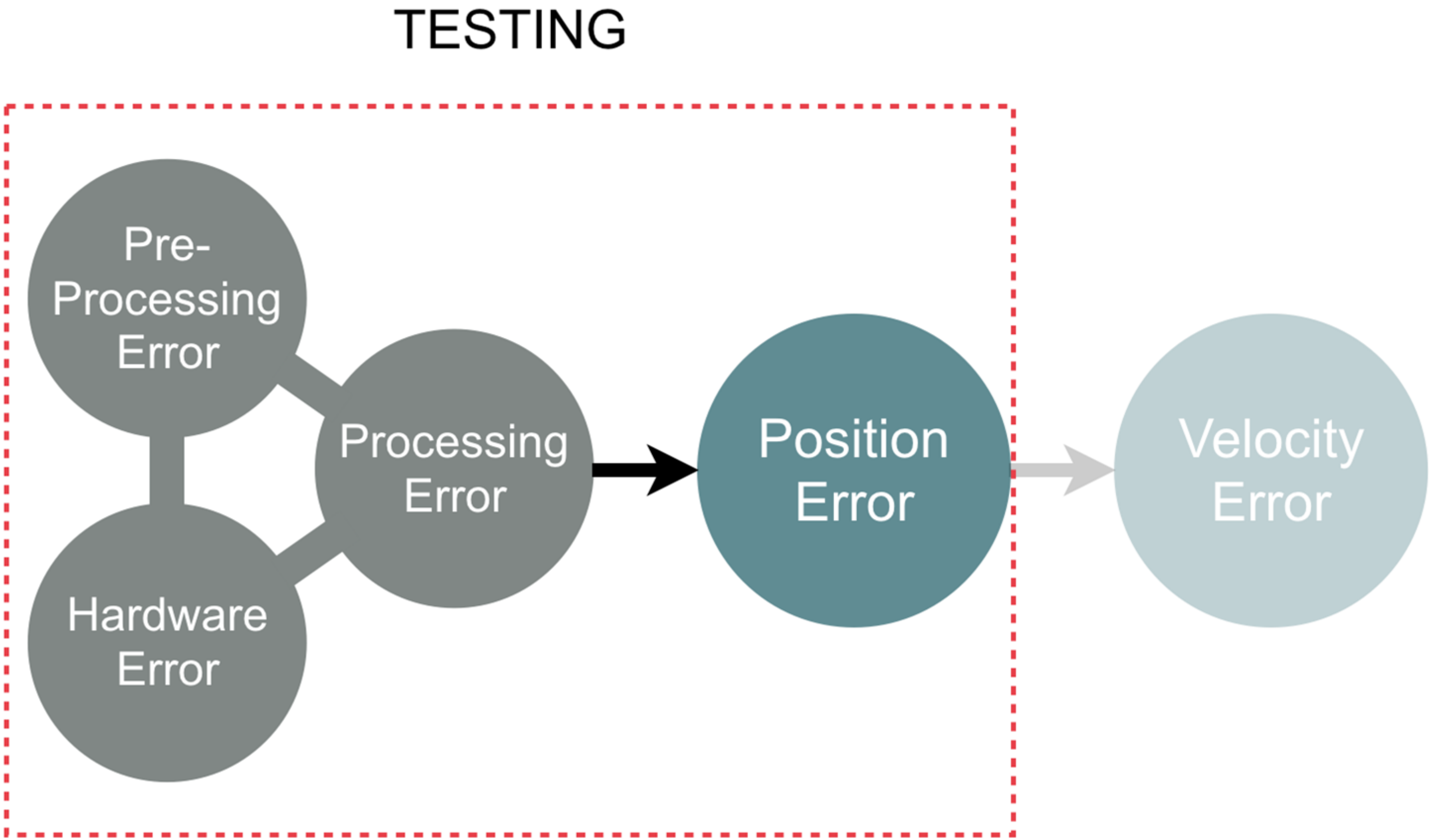
Velocity Error

Test Scope



Position Error
Positi
Erro
Pre-Processing Error
Processing Error
Hardware Error
Position Error
Velocity Error
Position Error
Positi
Erro

Test Scope



TESTING

Pre-Processing Error

Processing Error

Hardware Error

Position Error

Velocity Error

Mission Testing

Primary Objective: Quantify the OSPRE system's position error and subsequent velocity error throughout the mission.

Hypothesis: The OSPRE system meets error requirements throughout all key mission stages.

Secondary Objective: Ensure that the OSPRE system can operate autonomously and behaves as expected.

Hypothesis: The OSPRE system operates autonomously and as expected.

Mission Testing

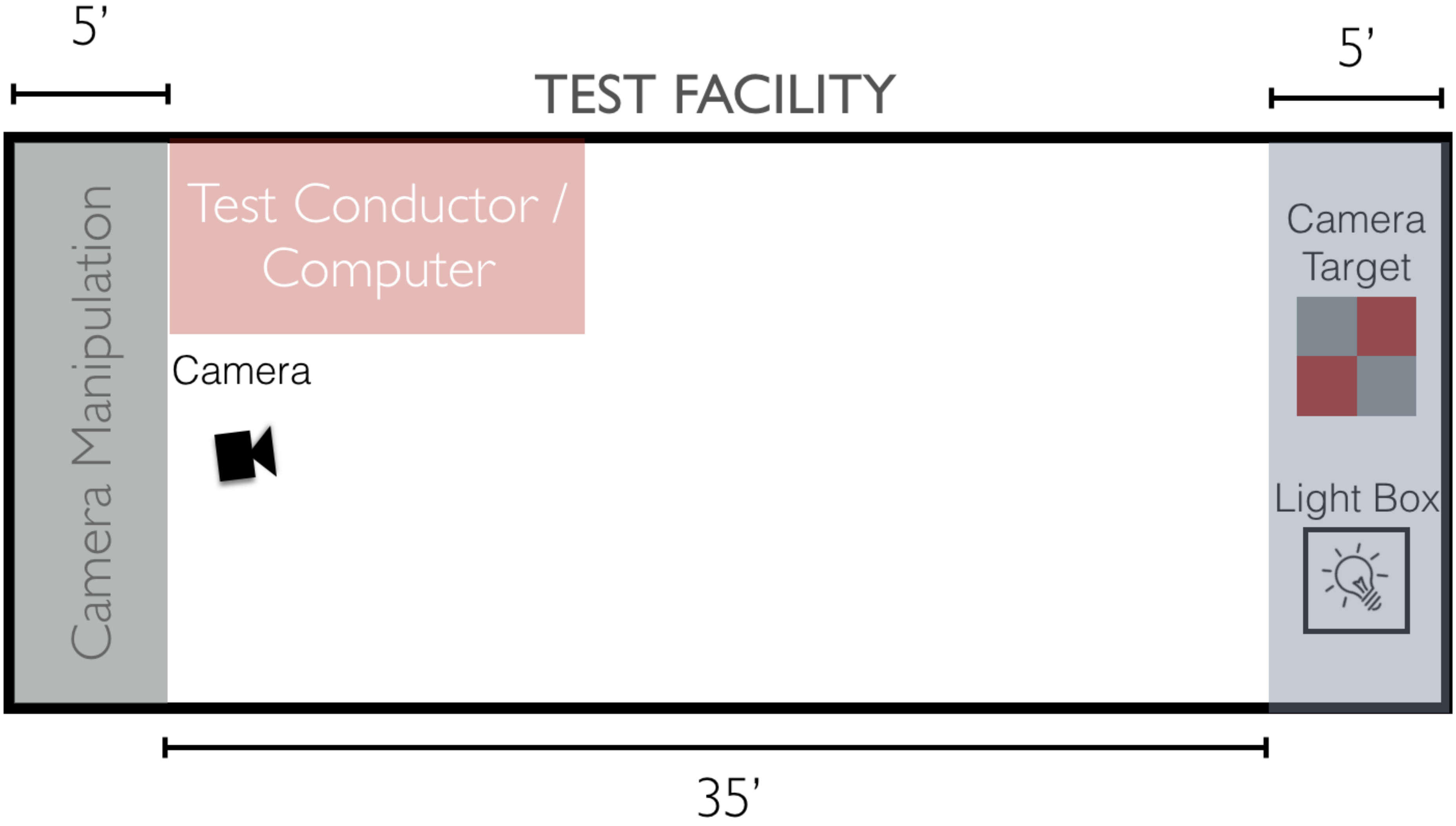
Primary Objective: Quantify the OSPRE system’s position error and subsequent velocity error throughout the mission.

Primary Design Driver: Prioritize the reduction of error introduced by the test setup

Result: Engineer a high-accuracy, simple testing solution that allows team OSPRE to test all key mission stages.

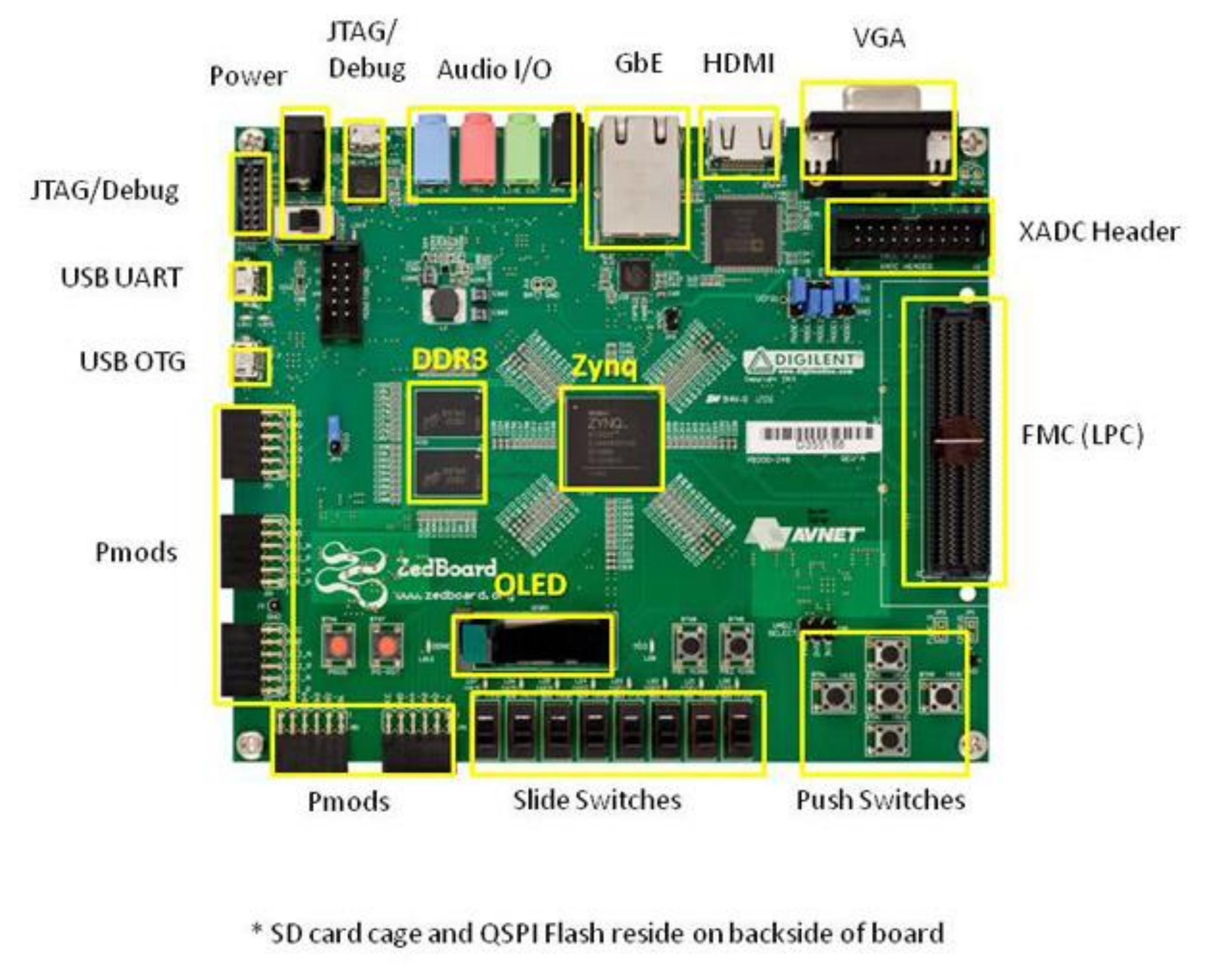
Mission Testing

Facility Layout



Simulated S/C Computer

ZedBoard



* SD card cage and QSPI Flash reside on backside of board

Image Credit: www.zedboard.org

Mission Testing Manipulation

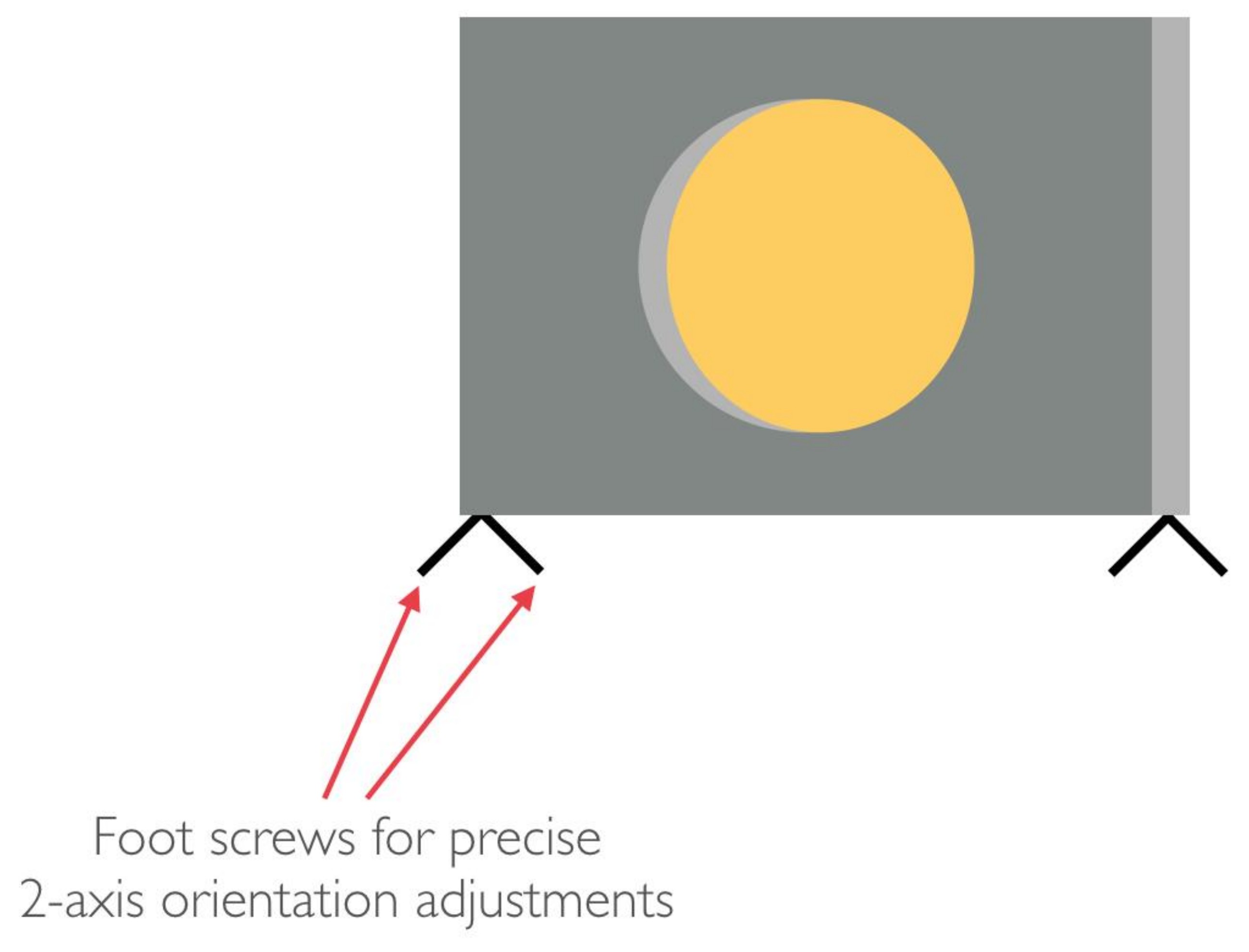
Camera Pointing

Manfrotto 410 Junior Geared Tripod Head

Manfrotto ProX Tripod

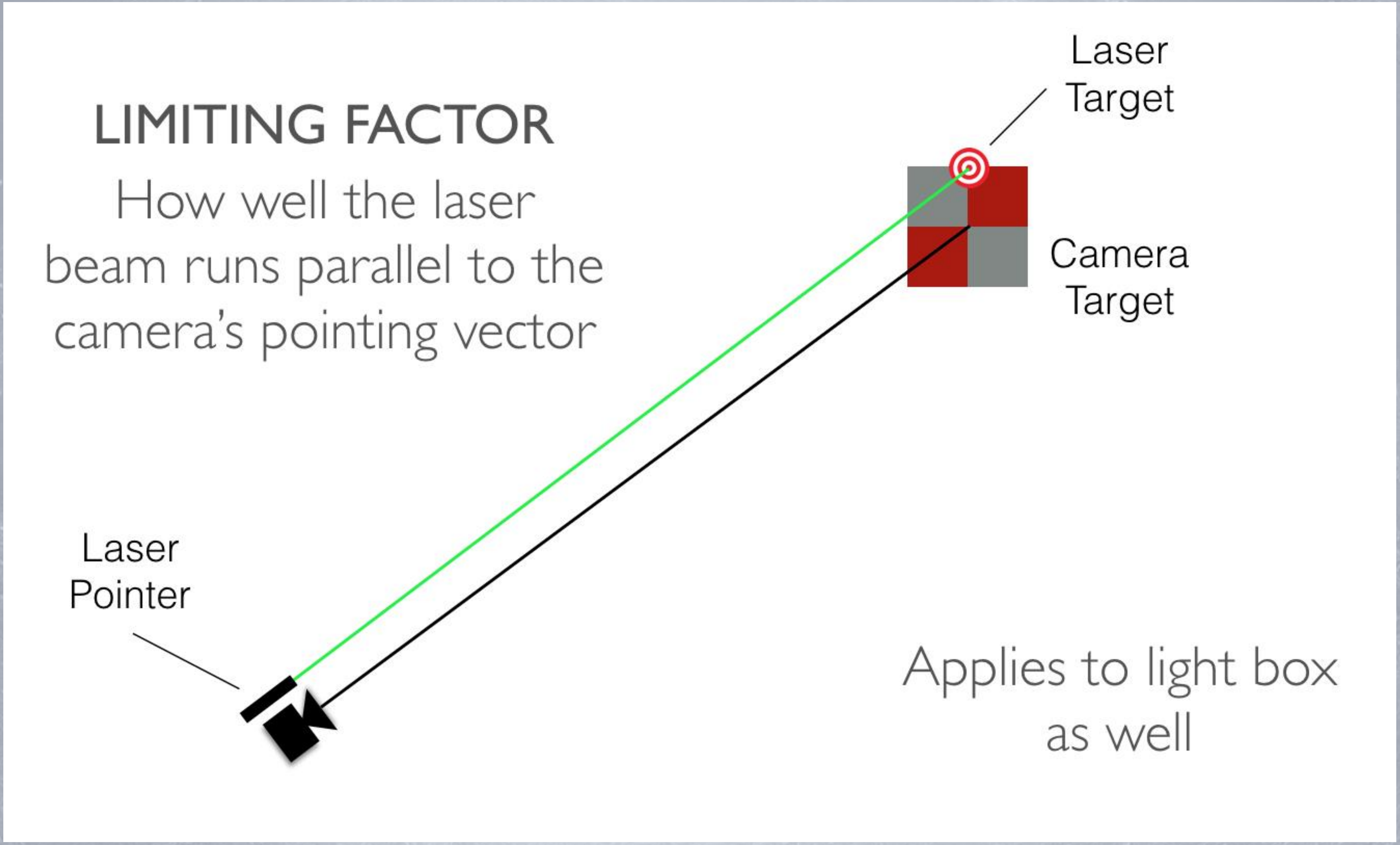
Mission Testing Manipulation

Lightbox Pointing



Mission Testing Error

Pointing Method



LIMITING FACTOR

How well the laser beam runs parallel to the camera's pointing vector

Applies to light box as well

Mission Testing Error

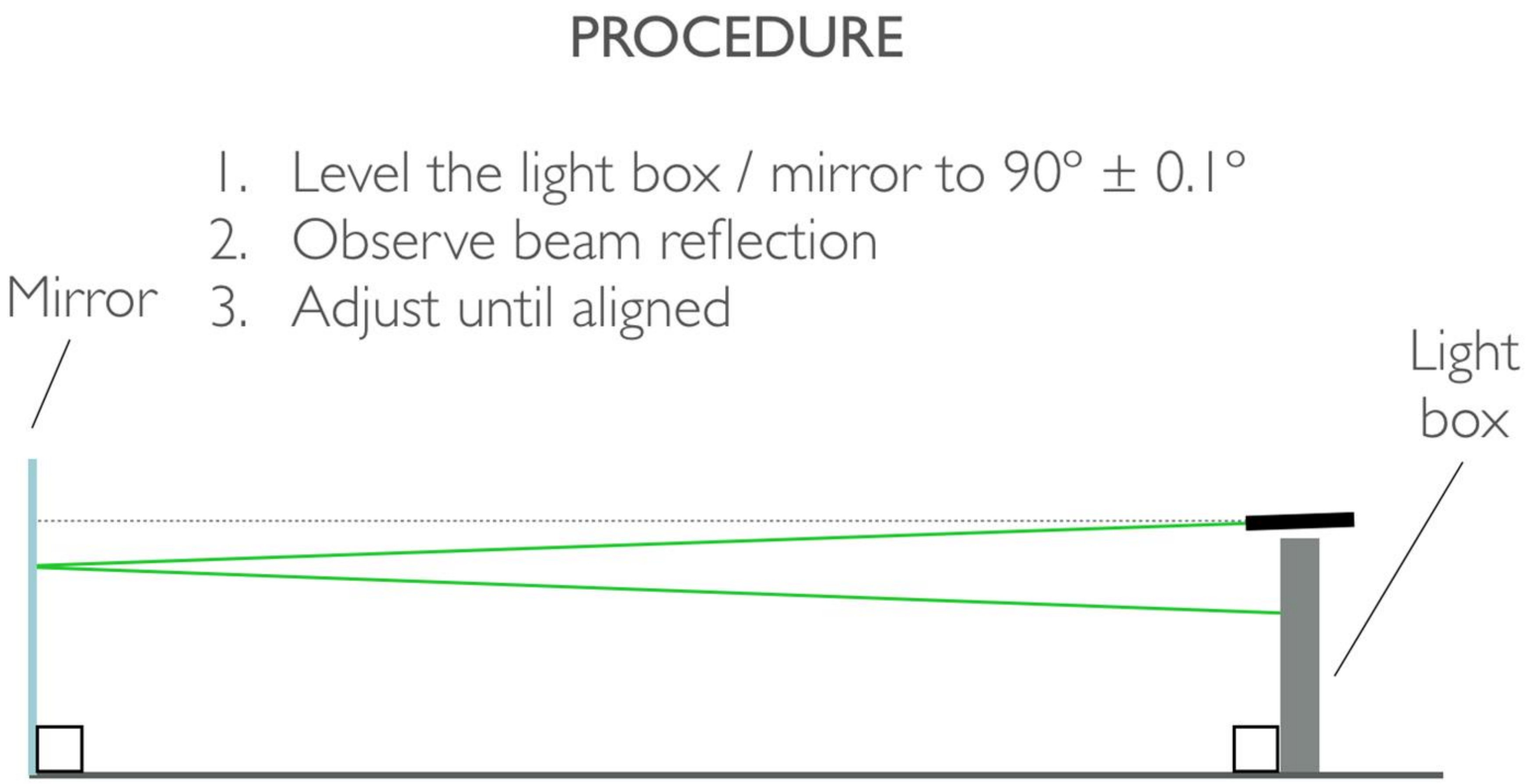
Alignment Method

TOOLS

- Laser Pointer Mount
- Digital Level
- Square

Mission Testing Error

Alignment Method



PROCEDURE

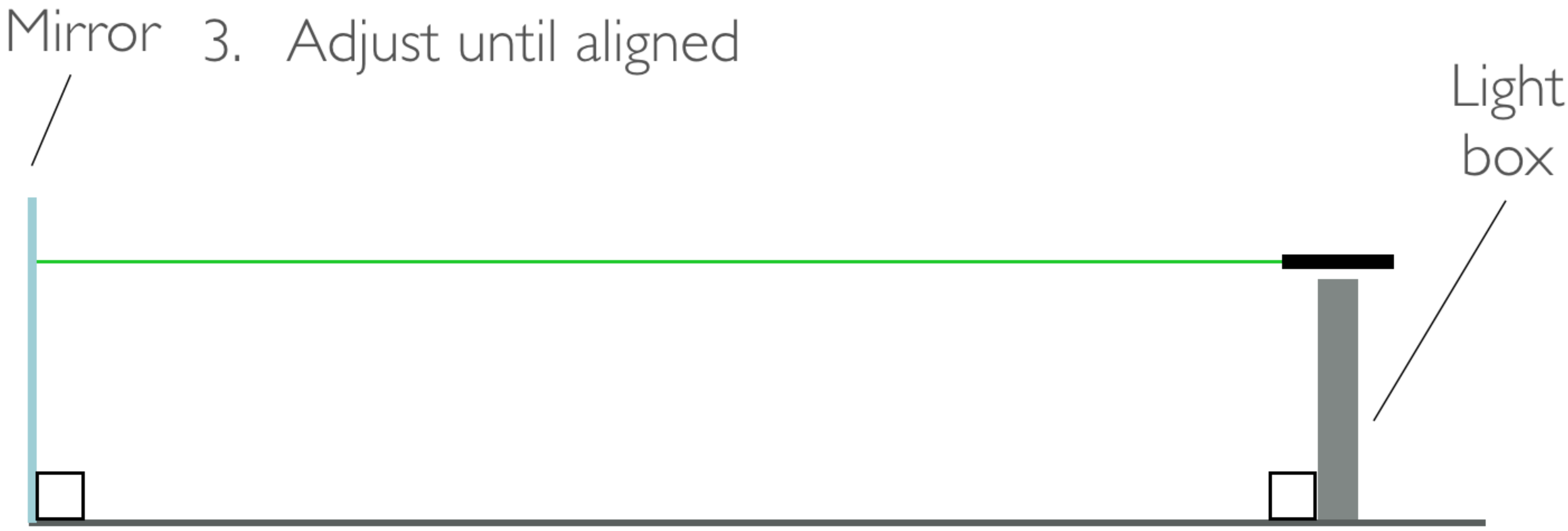
1. Level the light box / mirror to 90° ± 0.1°
2. Observe beam reflection
3. Adjust until aligned

Mission Testing Error

Alignment Method

PROCEDURE

1. Level the light box / mirror to 90° **± 0.1°**
2. Observe beam reflection
3. Adjust until aligned



PERCEIVED LOSS OF DIMENSIONS

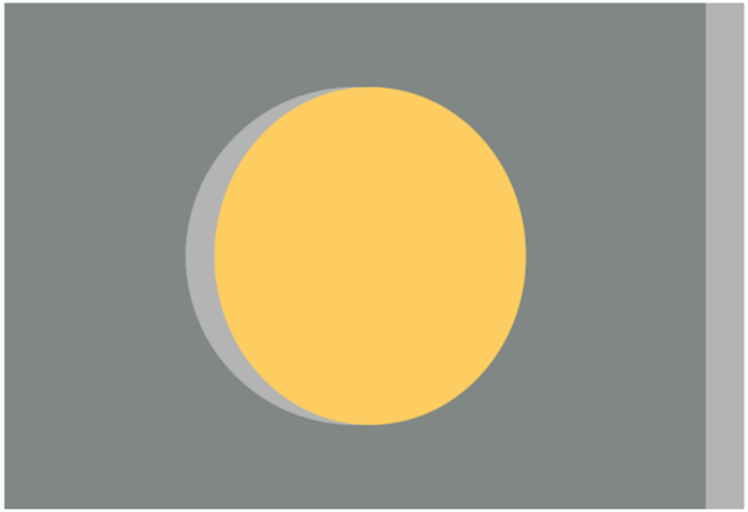
Lightbox Viewed Perfectly Straight On

No PLD

CAMERA SEES PERFECT CIRCLE

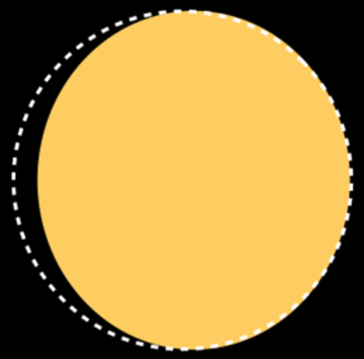
PERCEIVED LOSS OF DIMENSIONS

Lightbox Viewed From Angled Perspective



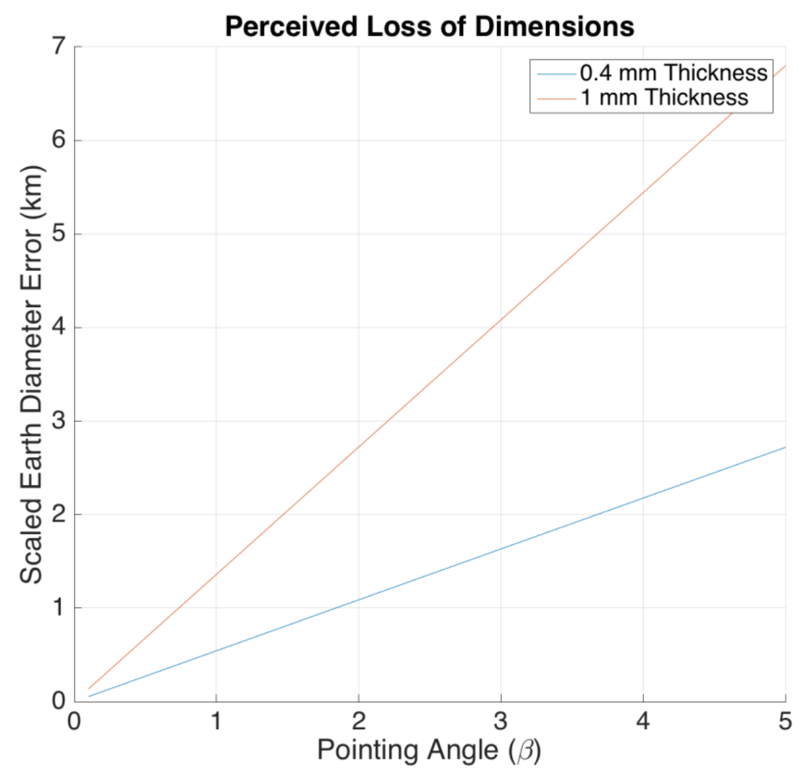
Noticeable PLD

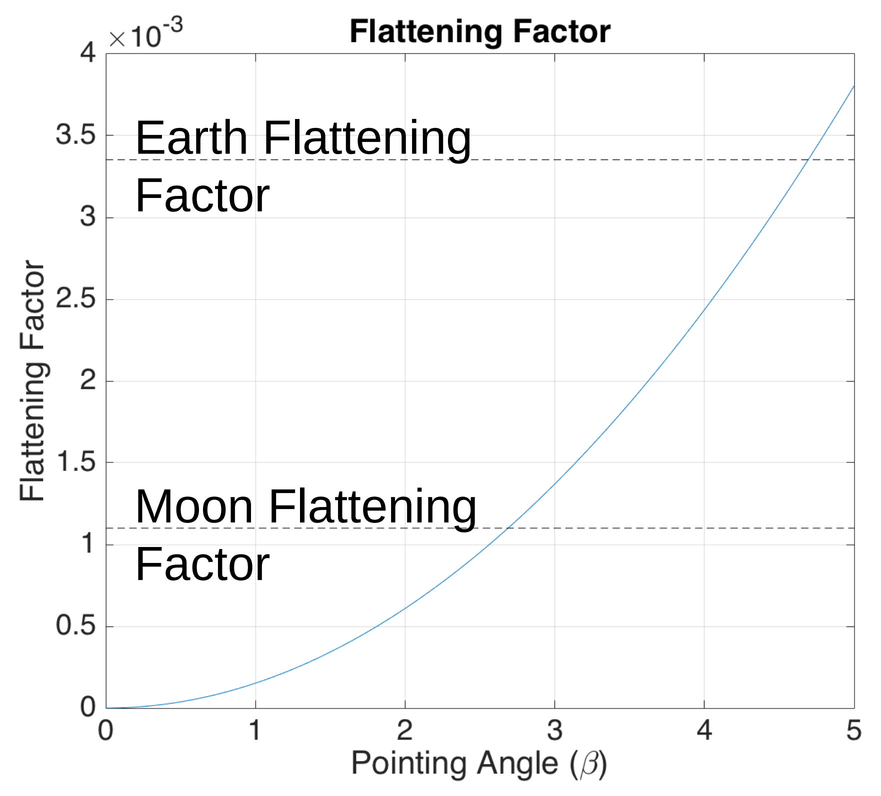
CAMERA SEES DISTORTED CIRCLE



Mission Testing Error

Lightbox Pointing Error Implications





Mission Testing Error

Camera Pointing Error

Procedure:

1. Take image of alignment target
2. Process image to find location of center of target relative to center of image frame
3. Adjust camera pointing
4. Repeat until centers are aligned

Operational Systems Testing Error

Pointing Accuracy Error Mitigation:

- Laser pointers running parallel to lightbox pointing vector
- Telescopic mounts allow for high precision pointing adjustments
- Alignment test procedure: confident to within **± 0.2°** accuracy
- Utilize image capture capability to improve camera pointing accuracy to sub-pixel pointing error

Operational Systems Testing Error

Distance / Dimensional Measurement Error

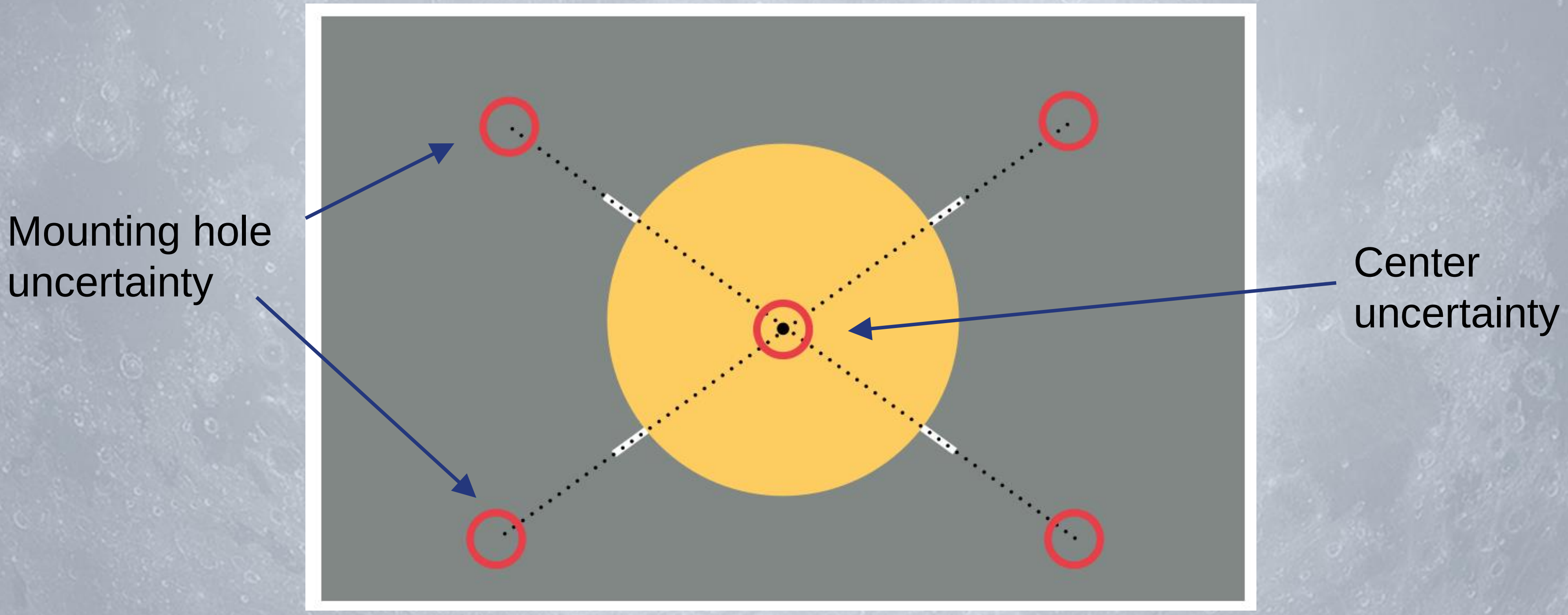
Sources

Tool	Model	Accuracy
Calipers	Glow Geek Digital Calipers	±0.025 mm
Steel Tape Measure	Hultafors Tape Measure	± 1.1 mm
Laser Range Finder	Distance Master 60	±1.5 mm

Operational Systems Testing Error

Center Finding Accuracy:

Dependent on rod mounting accuracy due to hole location uncertainty of ± 0.025 mm



Mounting hole uncertainty

Center uncertainty

Next Steps

Improved Simulation Fidelity

- Additional human error contributions
- More constrained measurement errors

Reduce Test-Attributed Error Contribution

- Pointing accuracy improvement

Finance Backup Slides

Testing Budget

Subsystem	Item	Quantity	Cost
Testing	Laser Range Finder	1	$130.00
	Calipers	1	$25.00
	Light Panel	1	$100.00
	Aluminum Sheet	2	$80.00
	Laser Pointer	2	$60.00
	Geared Tripod Head	1	$200.00
	Digital Level	1	$50.00
	Laser Pointer Mount	2	$60.00
	Mounting Materials	1	$50.00
	Texture Materials	1	$20.00
	Lightbox Legs	1	$20.00
	Phase Cutouts	1	$50.00
	Alignment Rods	1	$25.00
		Subsystem Budget:	$1,600.00
		Subsystem Total:	$870.00
		Subsystem In-Use :	54.4%
		Subsystem Margin:	45.6%

Mechanical Budget

Subsystem	Item	Quantity	Cost
Mechanical	Aluminum Encasing	1	$10.00
	Plywood and Misc	1	$40.00
	Misc Building Hardware	1	$20.00
	Black Polycarbonate	5	$20.00
	Cast Aluminum Base Plate	1	$100.00
		Subsystem Budget:	$450.00
		Subsystem Total:	$190.00
		Subsystem In-Use :	42.2%
		Subsystem Margin:	57.8%

Electrical Budget

Subsystem	**Item**	**Quantity**	**Cost**
Electrical	Snapdragon OpenQ 410 Development Kit	1	$299.00
	ZedBoard	1	$475.00
	PCB	1	$100.00
	15 MDSub Connector	1	$25.00
	Voltage Regulators	4	$2.00
	Thermocouples	2	$5.00
	Subsystem Budget:		$1,750.00
	Subsystem Total:		$906.00
	Subsystem In-Use :		51.8%
	Subsystem Margin:		48.2%

Optics Budget

Subsystem	Item	Quantity	Cost
Optics	13MP Camera	1	$125.00
	Sony FCB-MA130	2	$150.00
		Subsystem Budget:	$1,200.00
		Subsystem Total:	$275.00
		Subsystem In-Use :	22.9%
		Subsystem Margin:	77.1%