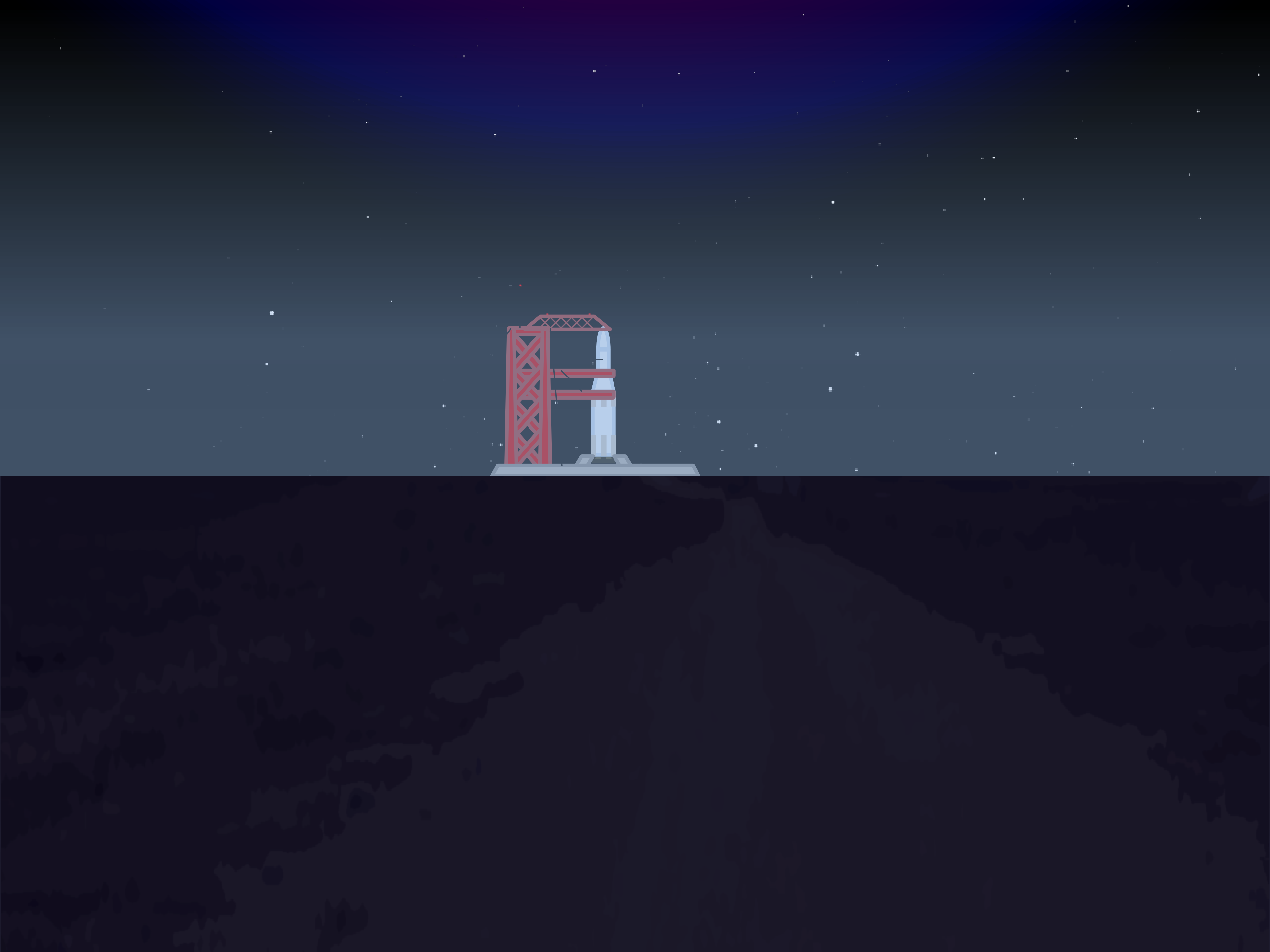
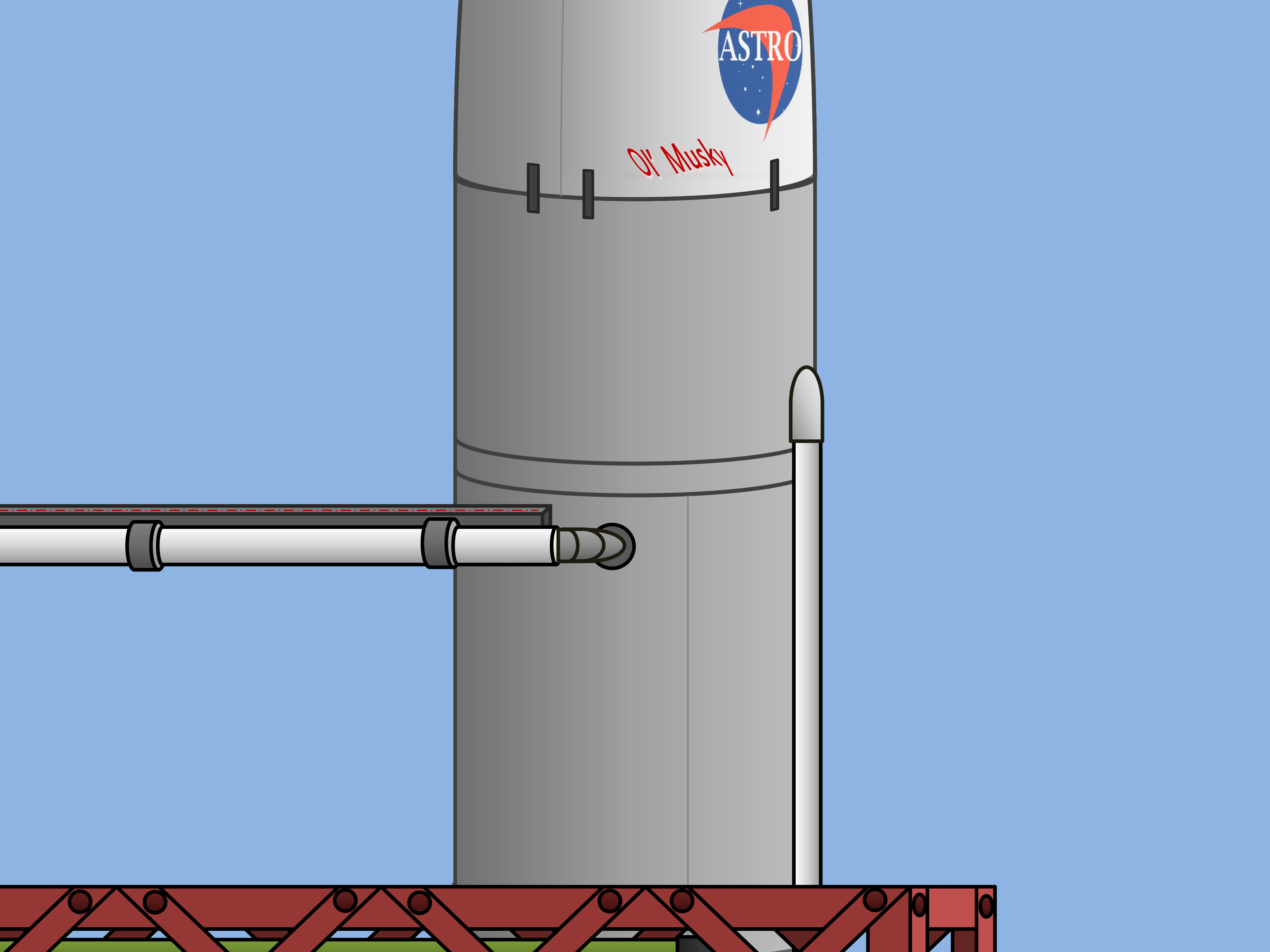
ASTRO



February 19 2018

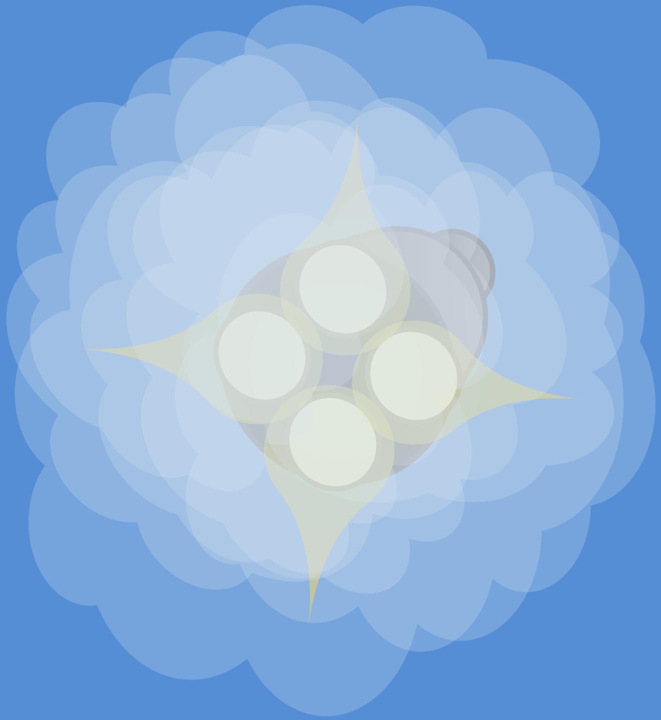
Woomera Launch Area 5

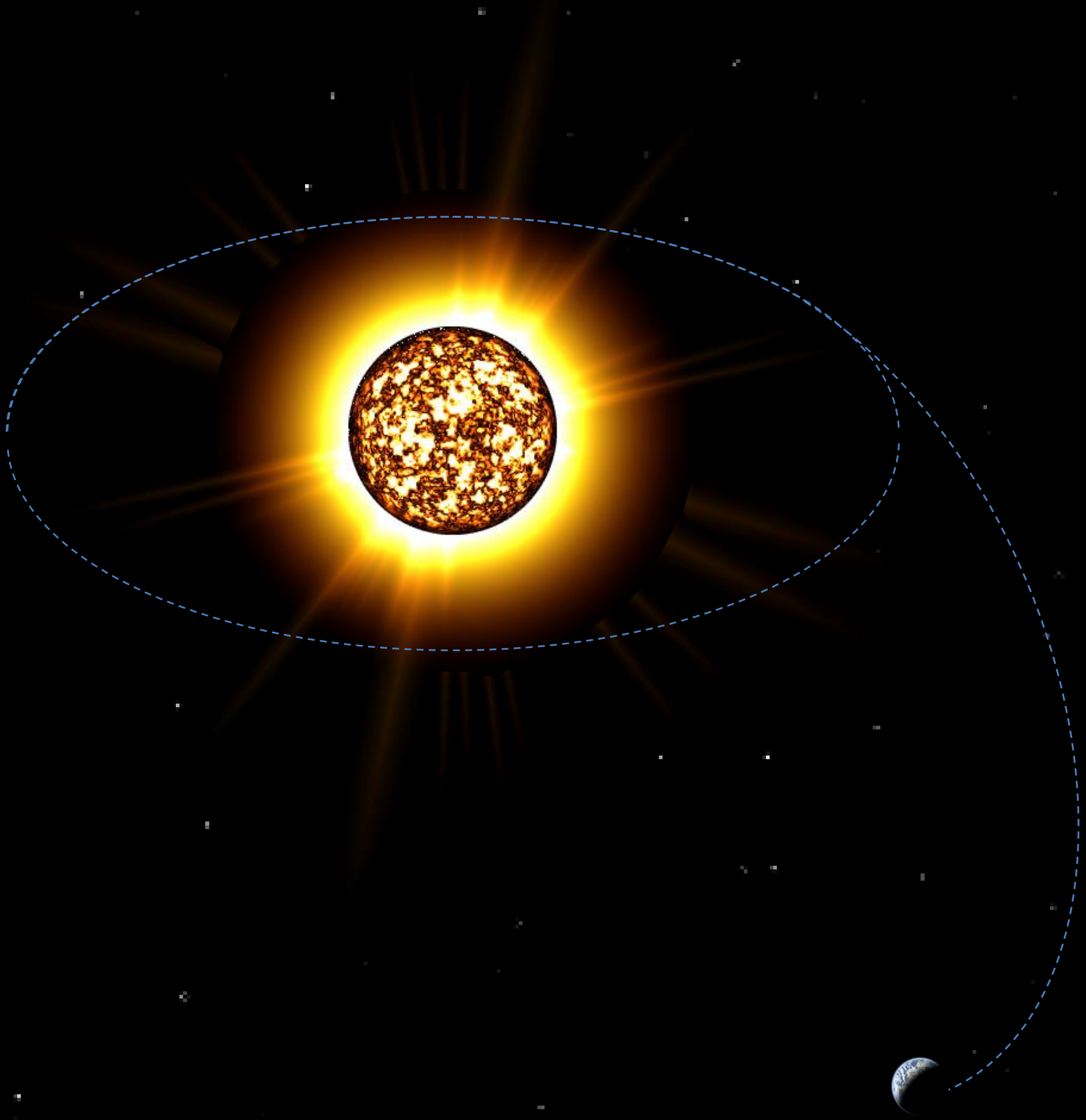
T-minus 00:00:28

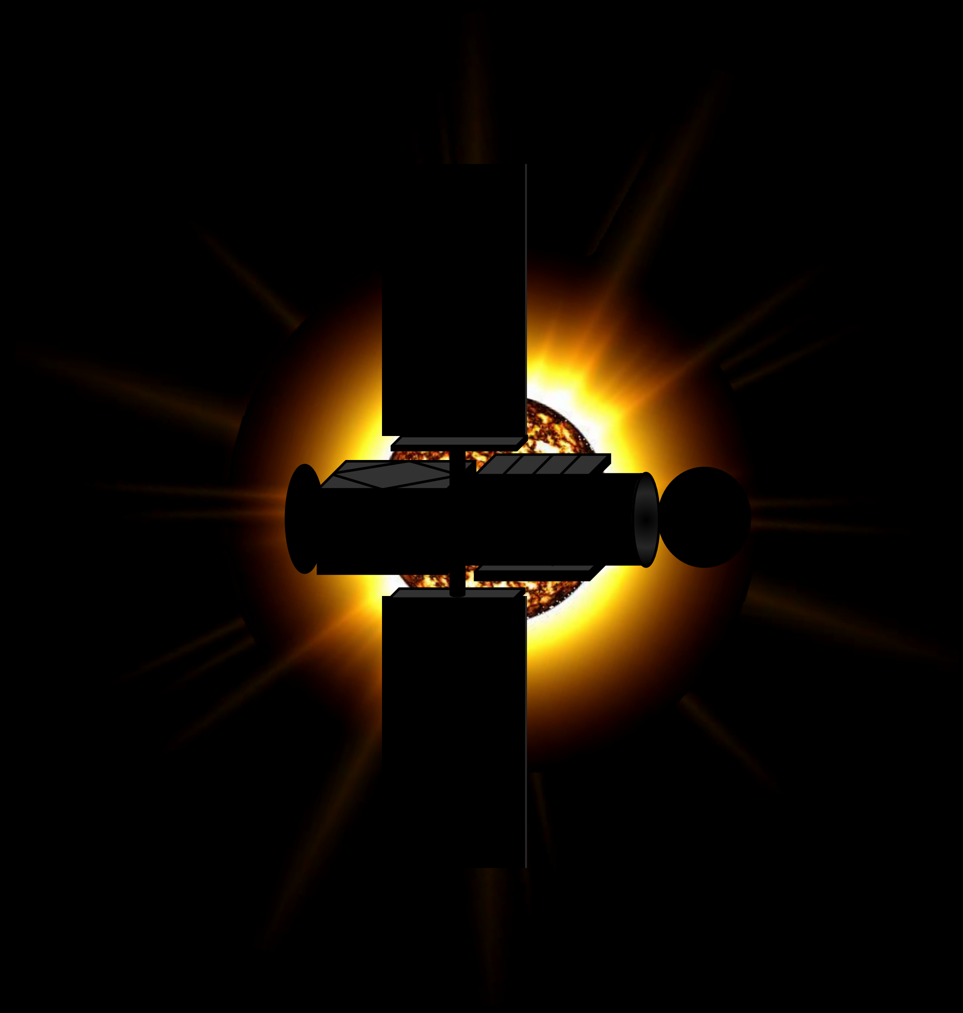

ASTRO
Ol' Musky

T-minus 00:00:0[illegible]

T-minus 00:00:00







AUSTRALIAN SPACE TECHNOLOGY RESEARCH ORGANISATION
ASTRO

The John Tebbutt Space Telescope

or

“In space, no one can hear you scream”

Paul Pounds

19 February 2017

University of Queensland

METR4810

- **What**: Mechatronics team project course
- **When**: Starting now, going until week 13
- **Where**: Hawken 50-c404 (mostly)
- **Who**: Cast of thousands
- **How**: Lots of work
- **Why**: Get experience developing complex mechatronic and robotic systems... and

because it’s awesome

Specific class objectives

- Explore the trade-offs involved in complex mechatronic/robotic systems
- Gain experience in multi-variable analytical design synthesis
- Exercise practical cyber-electromechanical integration and trouble-shooting techniques
- Build interpersonal skills working in teams

A quick note on objectives

Your objective: 7/7 grade

My objective: 5/5 SECaT

Shared priorities:

- Meet course objectives
- Reduce unnecessary work
- Have fun!

What this class is

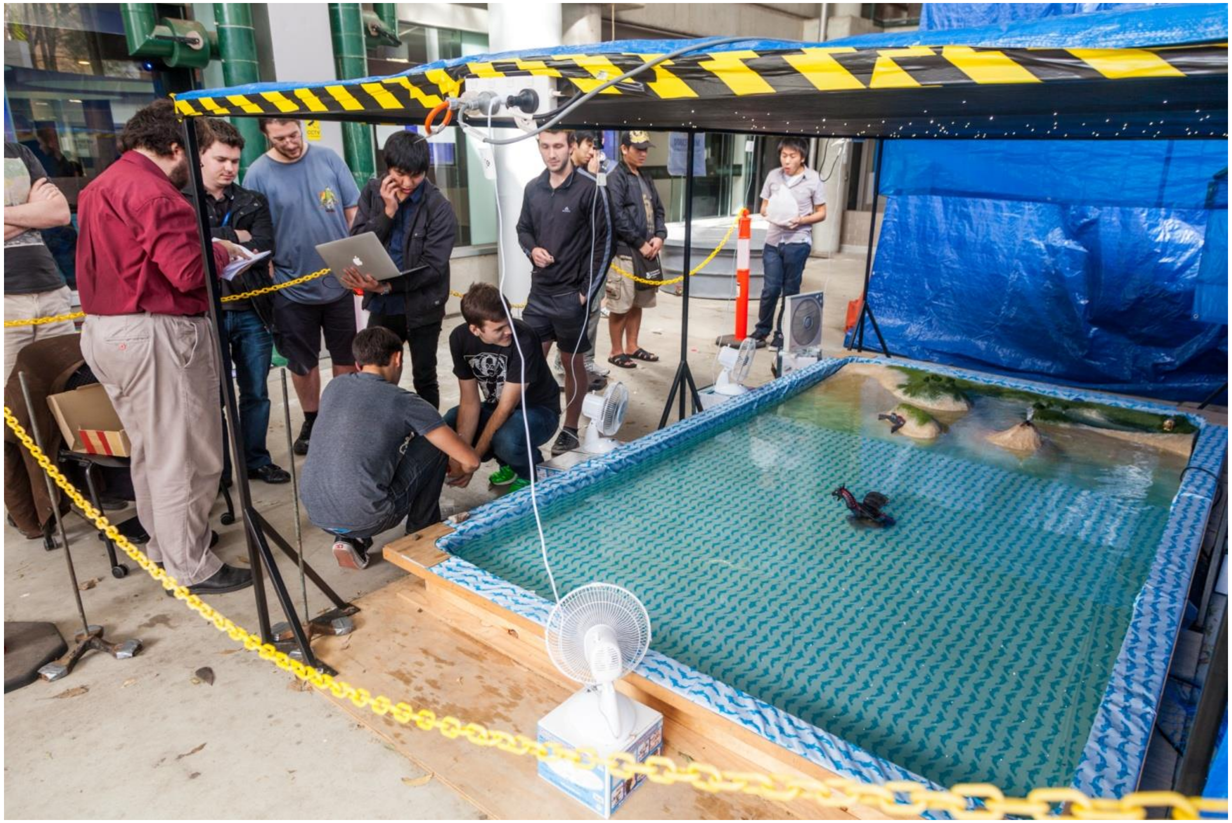
- Taking the safeties off
  - Real challenge, no hand-holding
- Unconstrained design, broad horizons
  - Very few limitations or constraints
- Focus on communication, design process, teamwork

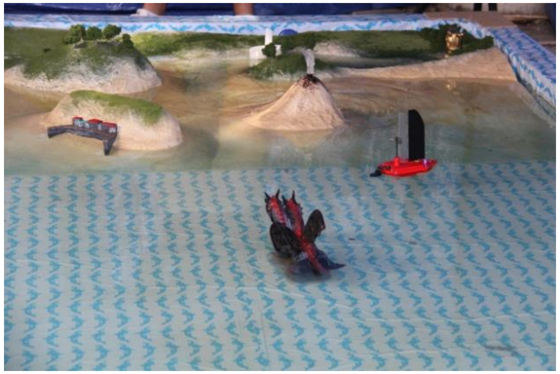
What this class is not

- Not about the project (not really)
  - It’s about how you go about solving it
- Not teaching you technical engineering
  - You already know how to do math, etc.
- Not giving you one single, clear path
  - It’s scary out there, and much is unknown

The Ghost of Projects Past

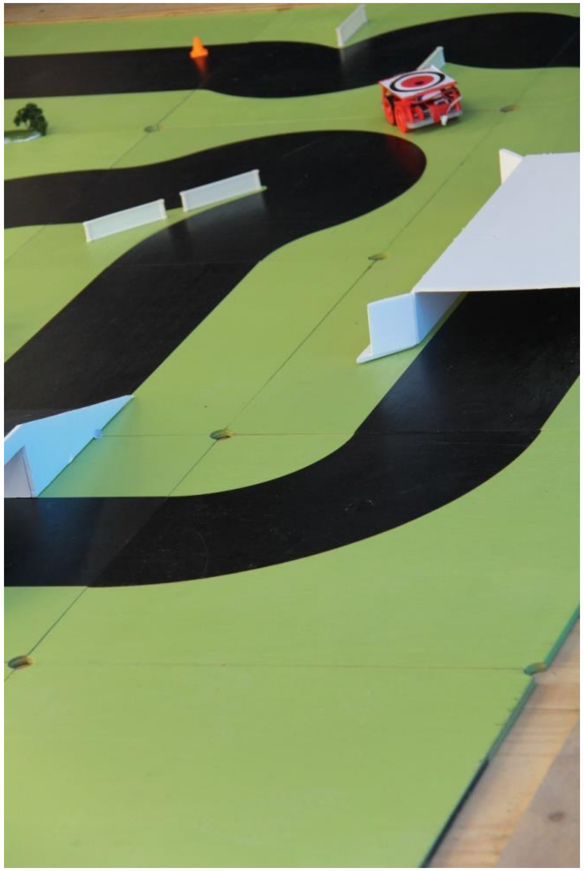
2013: Autonomous sailing and navigation





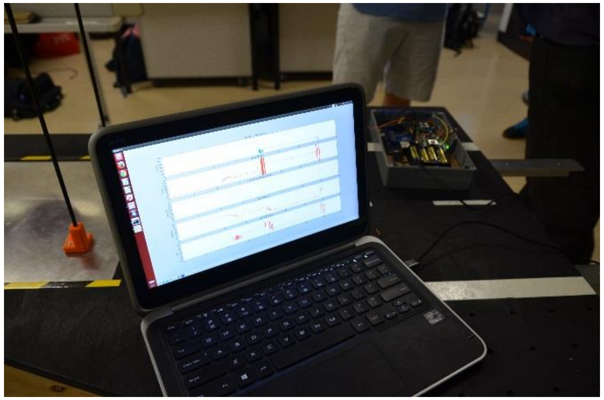
The Ghost of Projects Past

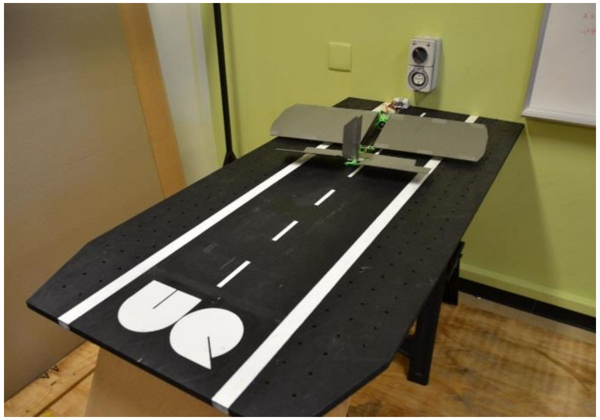
2014: Autonomous race car challenge



The Ghost of Projects Past

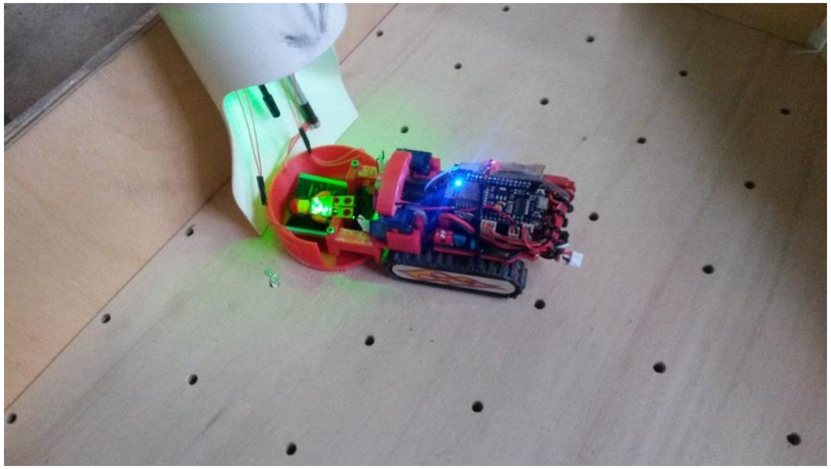
2015: Autonomous Carrier Operations

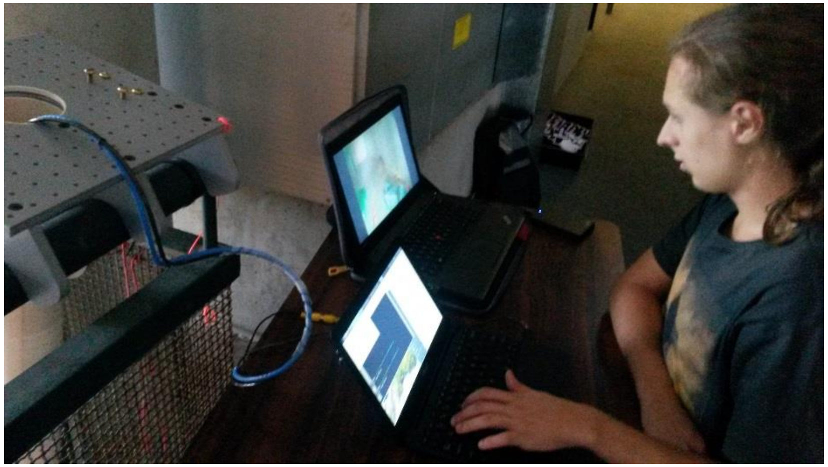




The Ghost of Projects Past

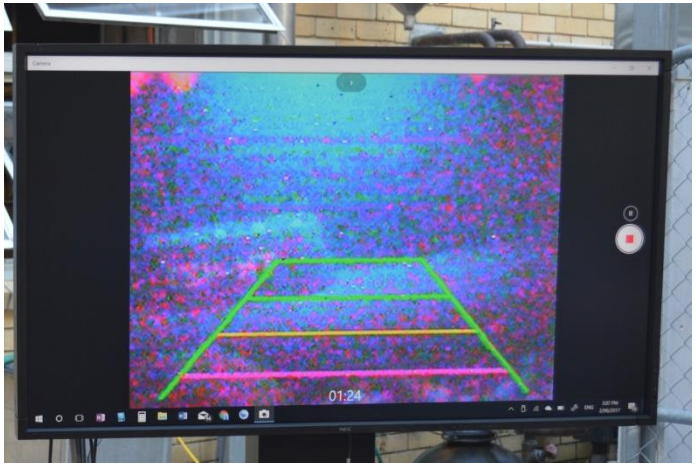
2016: Subterranean Mine Rescue

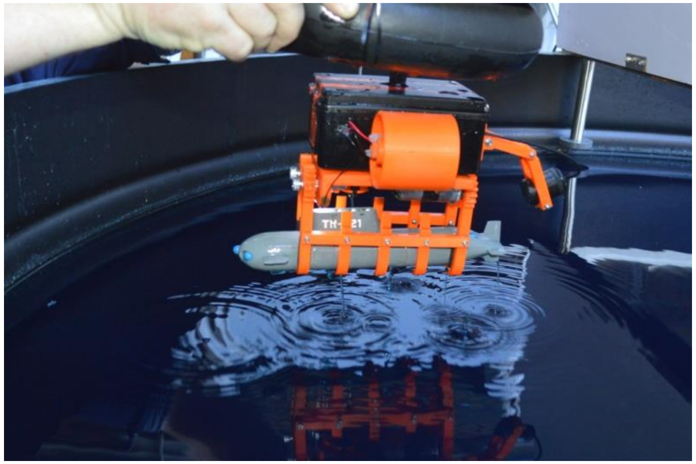




The Ghost of Projects Past

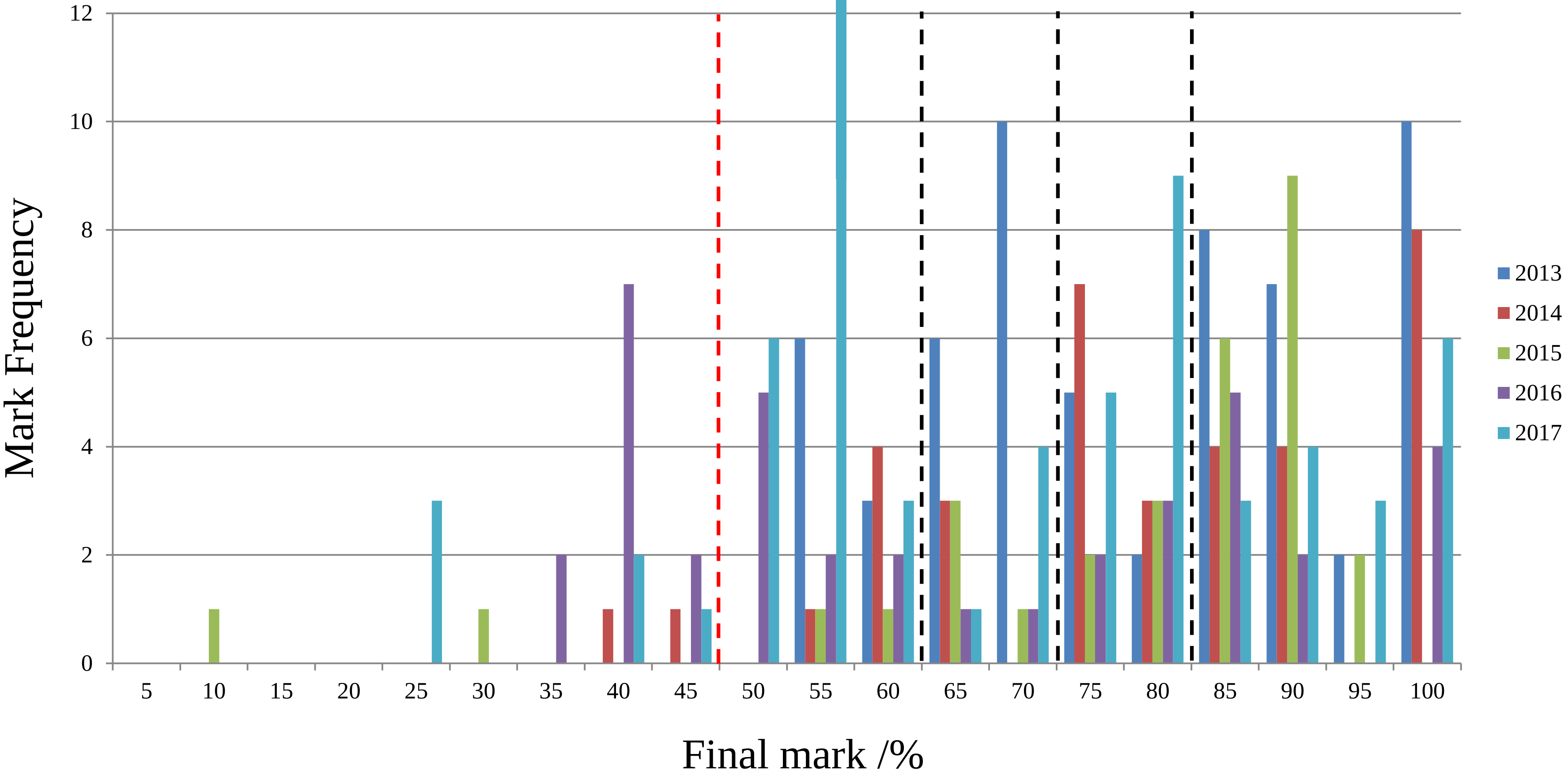
2017: Sunken Submarine Recovery





Assessment results

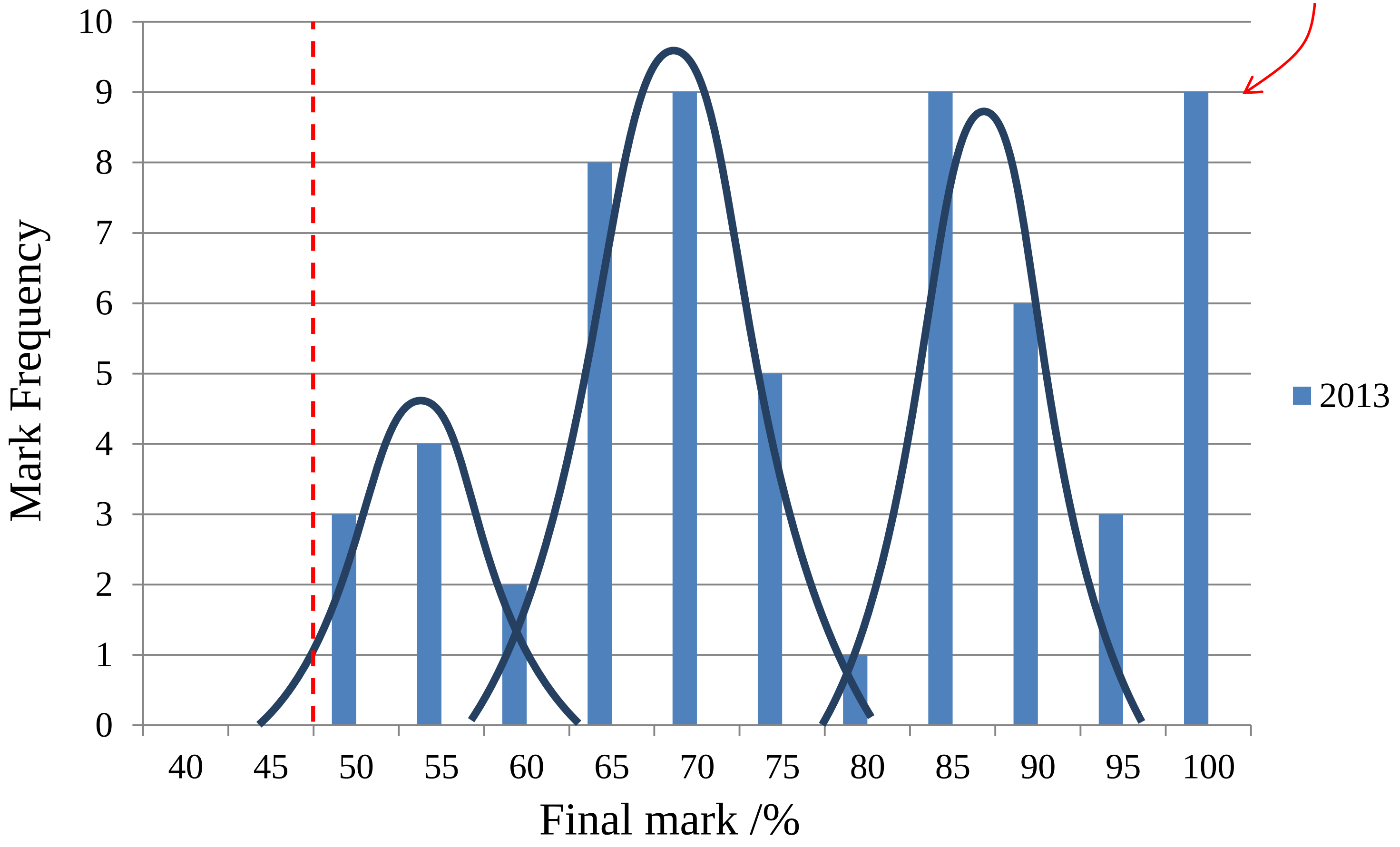
- Atypical mark spread: not a real bell curve



Assessment results

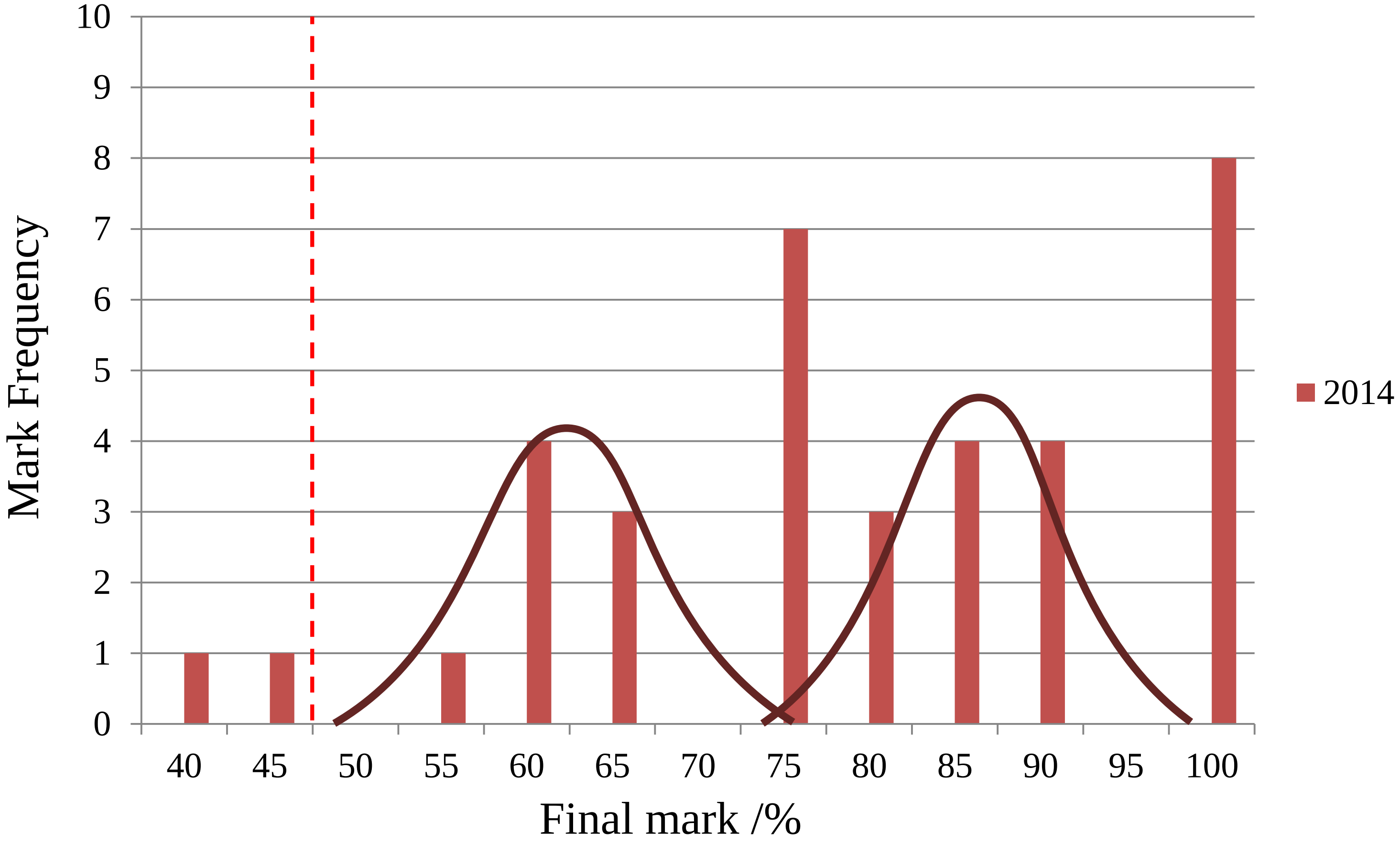
- Distinct mark clusters

“Awesome peak”



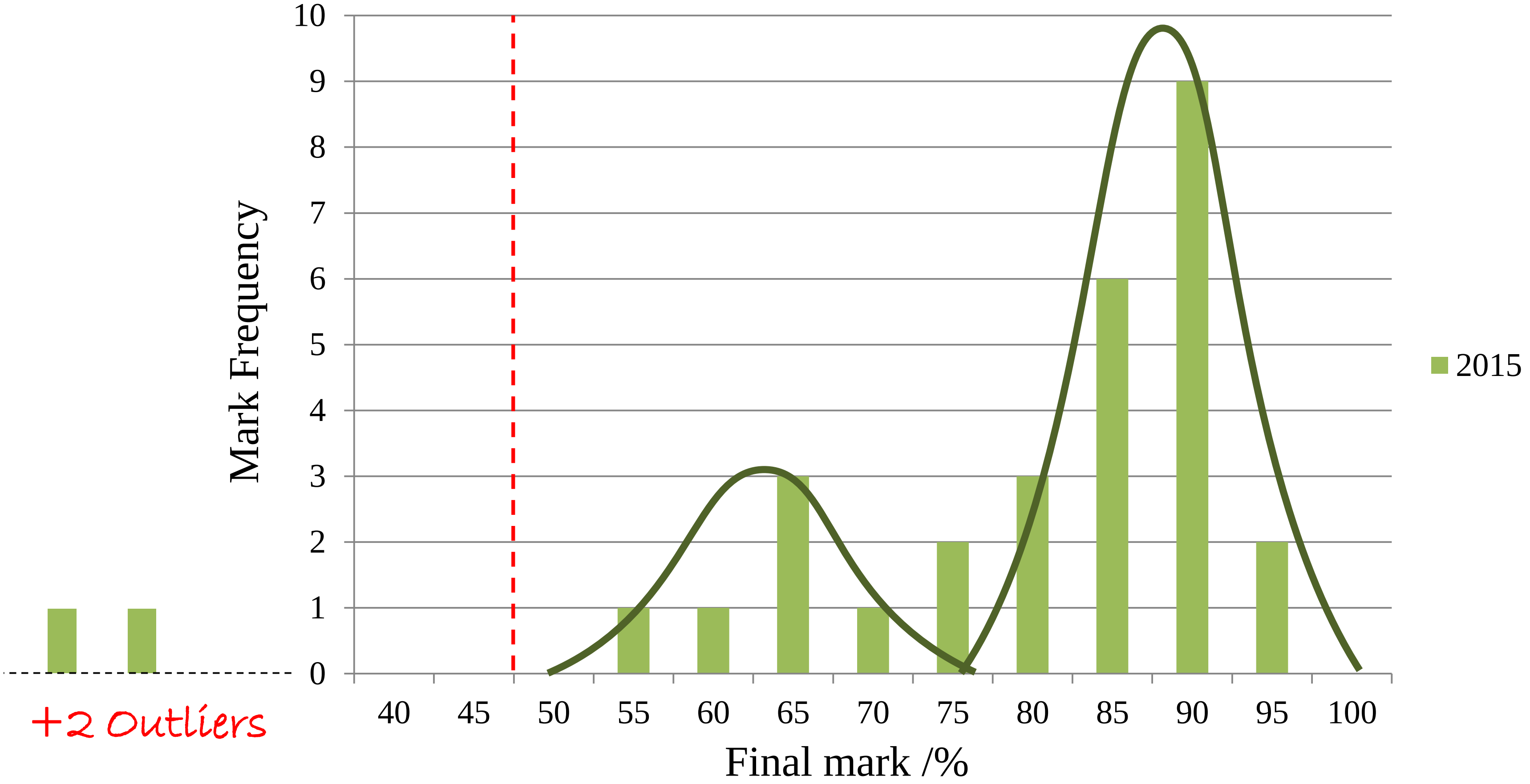
Assessment results

- Mark clusters move over time:



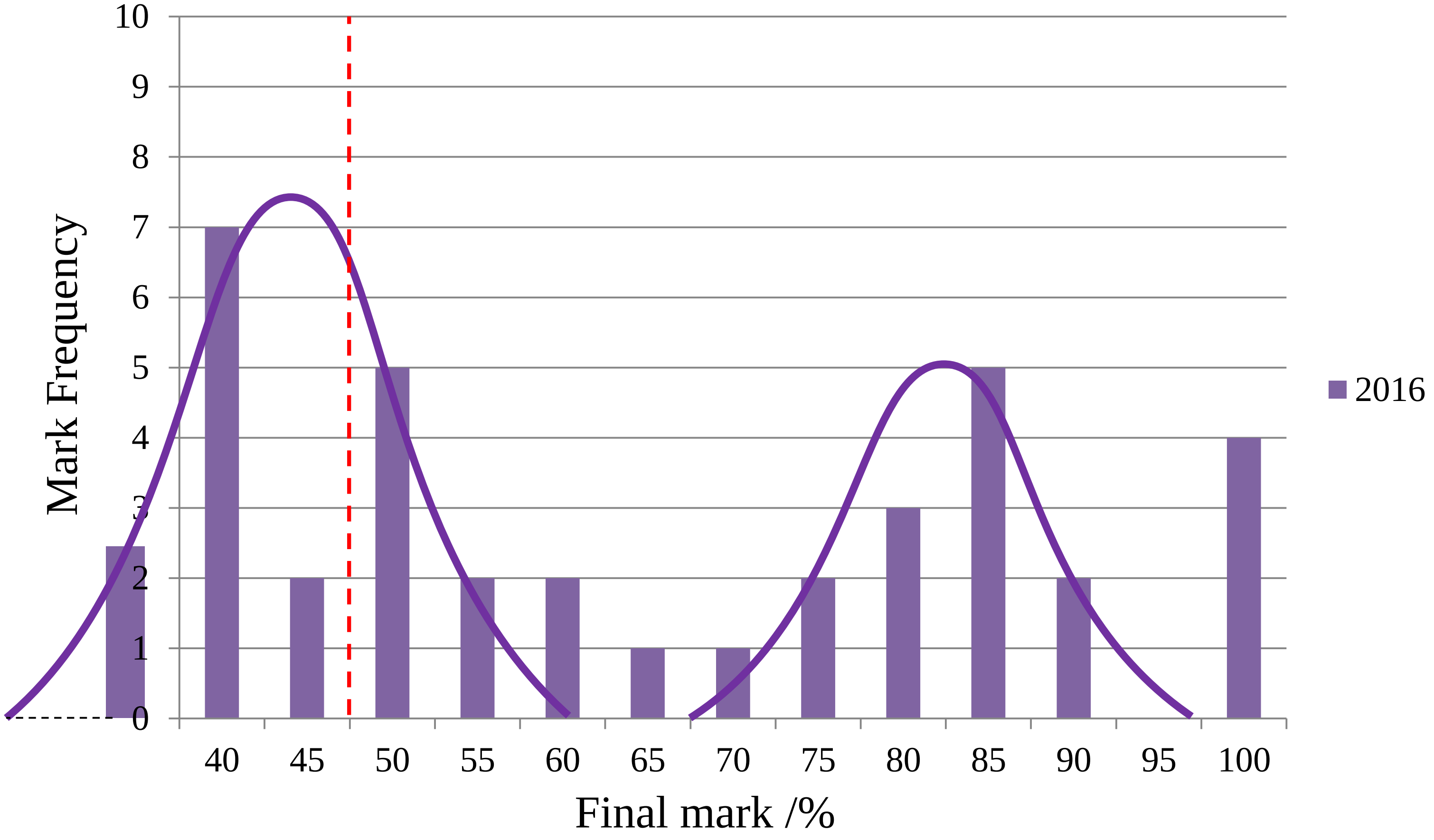
Assessment results

- Increasing performance, but more failures



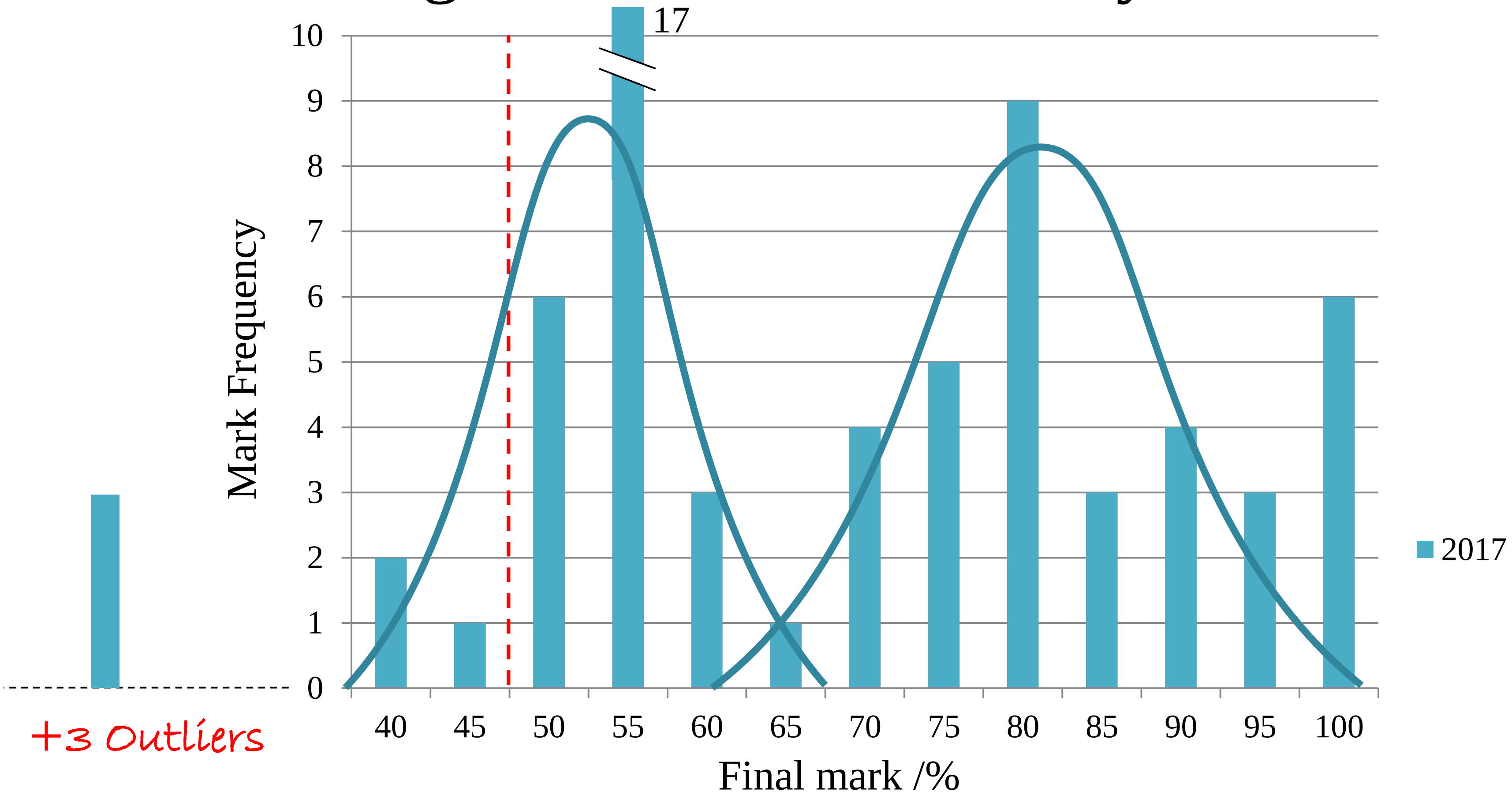
Assessment results

- Strong successes but higher failure rate, still



Assessment results

- Marking criteria saved so many butts!



Assessment results

- Or, put another way:

	Percentage of class					
	≤3	**4**	**5**	**6**	**7**	**Avg**
2013	**0**	25	24	**19**	**32**	5.58
2014	**6**	14	8	**28**	**44**	5.90
2015	**7**	16	10	**30**	**37**	5.66
2016	**29**	24	5	**21**	**21**	4.52
2017	**18**	31	13	**18**	**19**	4.94

Typical student outcomes

Student tend to fall into two broad groups:

The **Gets-its** and the **Don’t-Gets-its**

- The Gets-its work as a team, deconstruct the task rationally, try to understand the real problem, and implement a solution well

- The Don’t-Gets-its... don’t

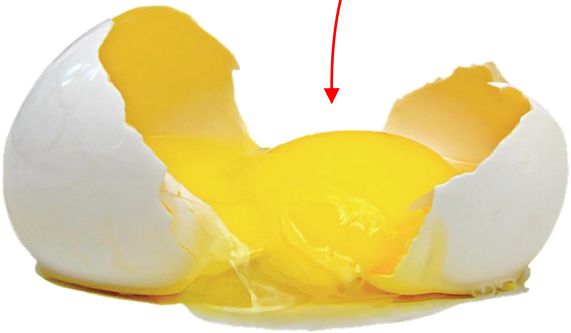
Sins of the forebearers

- 2013: Mortal ability, immortal ambition.
- 2014: You cannot 3D print a passing grade
- 2015: Balsa, electrical tape, hot glue and paperclips – *zero engineering analysis*.
- 2016: “Testing? What testing?”
- 2017: Assumed maximum hand-in volume limitation wouldn’t be enforced – it was.

Welcome back, frequent fliers

This course breaks eggs

- You are probably going to find this course technically or socially challenging (or both!)
  - This is intentional



Don't let this happen to you!

How to pass this course

- Work as a team
- Get started early
- Deconstruct the task logically
- Understand the *real* problem
- Implement a solution well

How to fail this course

- Don’t contribute to your team
- Do it all at the last minute
- Don’t play nice with others
- Fixate on your pet approach
- Do lazy, effortless hacking

Enough about other people…

Now it’s your turn

PART 1

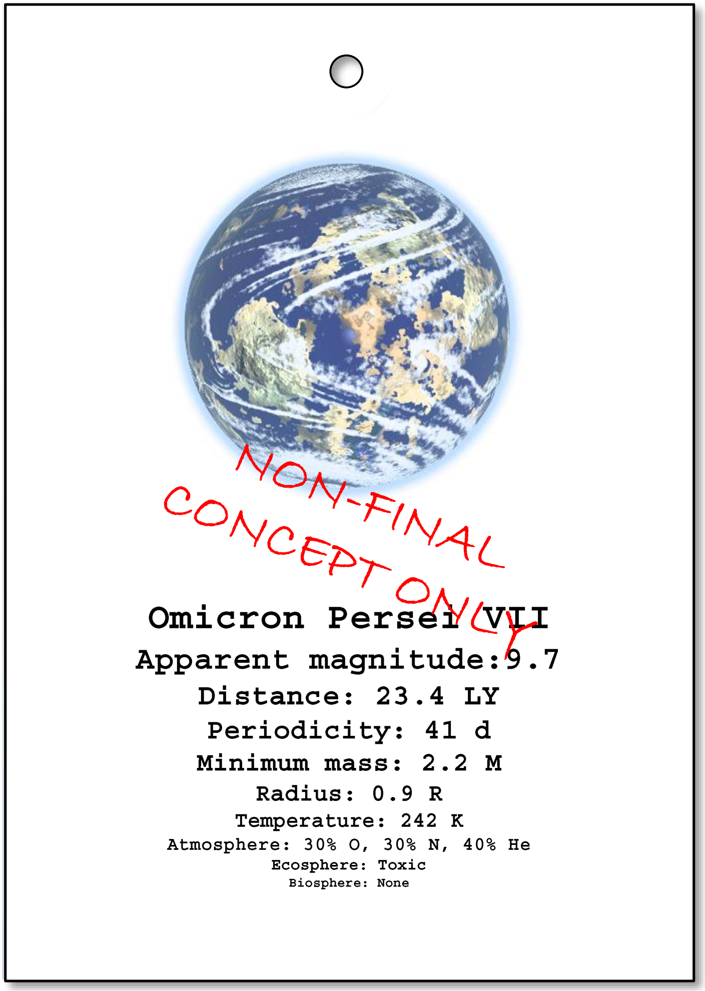
The Project

The task

Build a (miniature) space telescope and ground control station for imaging placards representing stars and exo-planets at the far end of the Hawken gallery and transmit the pictures to ground.

Apparatus

- Celestial targets consist of placard “eye charts”.
  - Each contains a picture of a planet, plus data text.
  - Reading smaller font text data yields more marks.
  - Some targets have ‘notable’ features for bonus points.

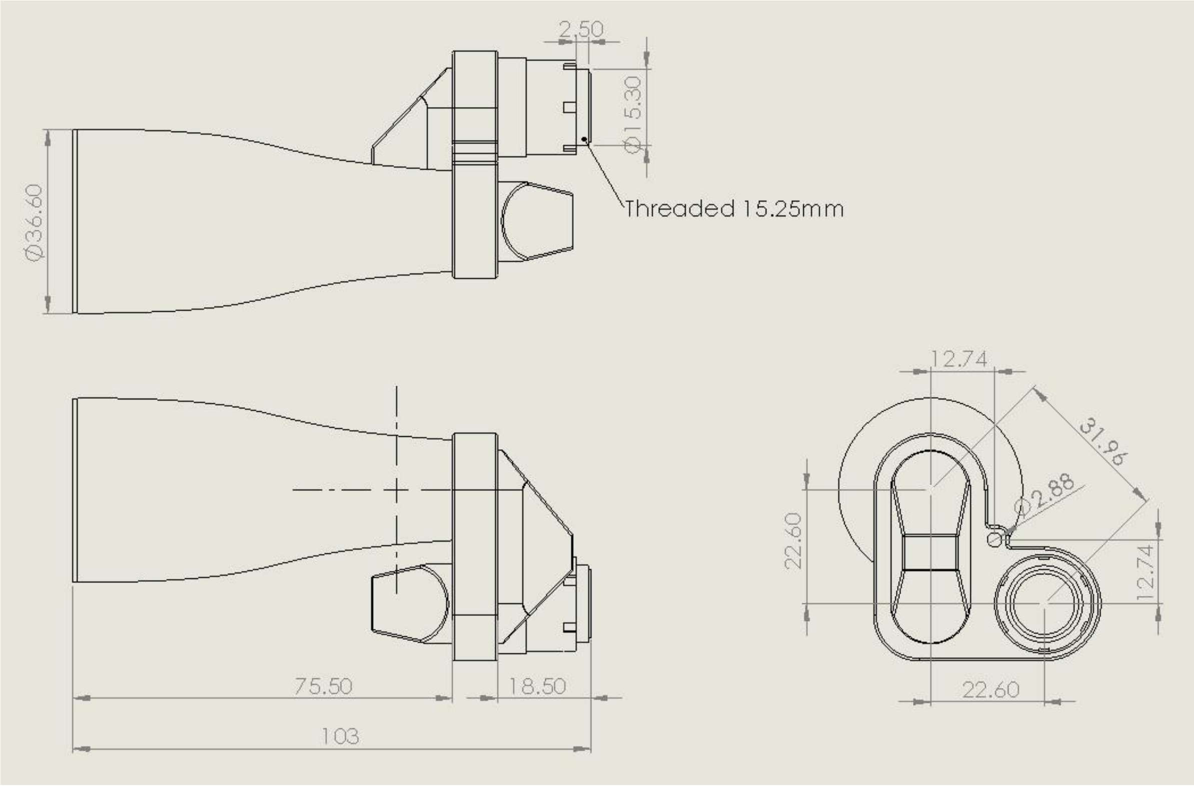


NON-FINAL CONCEPT ONLY

Omicron Persei VII
Apparent magnitude:9.7
Distance: 23.4 LY
Periodicity: 41 d
Minimum mass: 2.2 M
Radius: 0.9 R
Temperature: 242 K
Atmosphere: 30% O, 30% N, 40% He
Ecosphere: Toxic
Biosphere: None

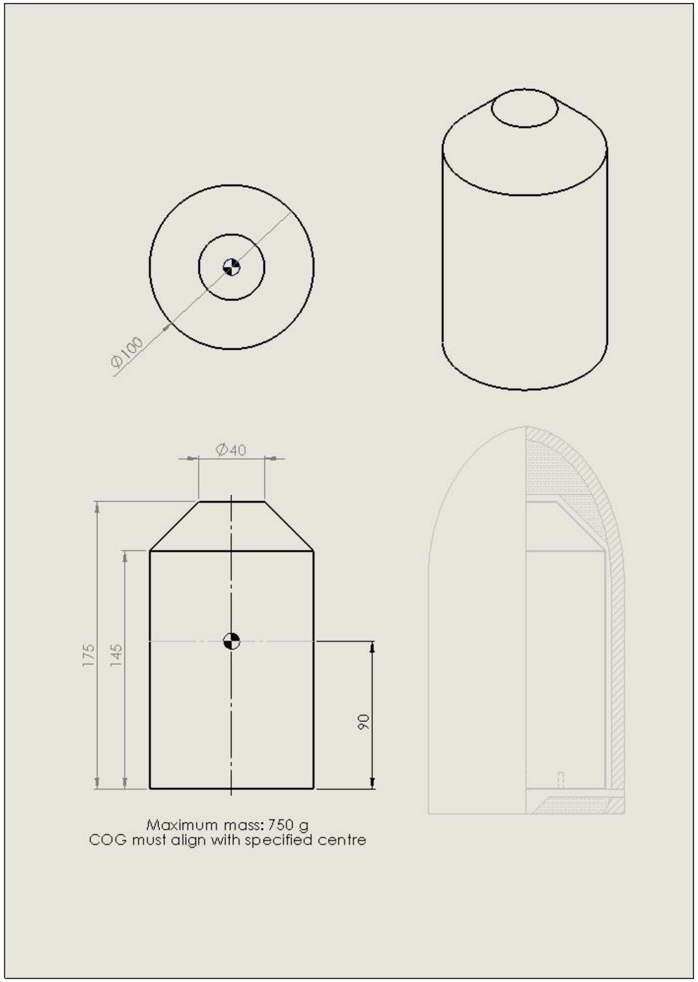
Apparatus

- A stock “optics module” will be provided
  - You still have to provide an imaging sensor tho’



Apparatus

- The telescope must fit within *strict* payload size/weight limits
  - Pieces will be removed with side cutters until it fits within the limits.
- Must be self-contained
  - No cables, wires, etc.



Maximum mass: 750 g
COG must align with specified centre

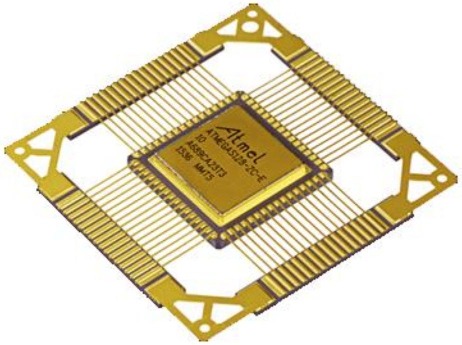
Zero-gravity vacuum environment

- The zero-g environment of the telescope will be simulated with a low-friction gimbal.
  - Large enough to accept the full payload volume.
  - Singularities (gimbal lock) above and below.
  - Multiple gimbals will be available for testing.

- No part of your satellite may depend on gravity or an atmosphere to function.
  - Ie. no accelerometers/propellers/pendulums/etc.

Space-rated microcontrollers

- The deep space environment requires a space-rated supervisory microcontroller
  - Doesn’t have to be an *actual* rad-hardened micro; just show that it comes in a space-rated version
  - All key subsystems must be able to be power-cycled by the space-rated microcontroller



Lead developers

- Each team member shall be ‘lead developer’ for one of the four key subsystems:
  - Mechanical and power management bus
  - Orientation control system
  - Telemetry and imaging
  - Ground control interface.
- The lead developer is responsible for monitoring (and ensuring) the progress of his or her subsystem

Other things

- No 3D printed parts in the satellite
- No Atmega 324/328 microprocessors
- Each subsystem must be demonstrable in isolation or using stubs

Full details on restrictions and constraints are in the task specification document

Key points

- Unlike previous years, you are not being asked to build something fully autonomous
  - Much like an open-book exam, the expectations will be correspondingly higher
- This task is intended to be *challenging*
  - Focus on getting readily achievable marks first
  - Don’t underestimate the effects of variability
  - Consider limitations on testing availability

Scoring

- Performance will be measured with a point system for demonstrated functionality

- Points will be awarded during scheduled demonstration sessions in week 13
  - 30 minute total time for set up and test
  - Last 5 minutes reserved for pack-down/marking

See rules and description document for full details

Functionality and scoring

Build quality	**10/10 Points**

Basic Functionality	**25/25 Points**
Satisfy the payload flight specification	5
Demonstrate subsystem module power cycle sequence	10
Return an image to the ground	10
Maintain a constant orientation in space	10
Return an image of a target planet	15

Protip: Passing the class pretty much requires you to be able to do this

Advanced Functionality	**40/40 Points**
Target planet image returned	1 per target
Target large text deciphered	2 per target
Target medium text deciphered	3 per target
Target small text deciphered	4 per target

Bonus Functionality	**10/10 Points**
Search board survey planet small text deciphered	1 point each
Life detected	5 points

The low energy solution

- There is often a simple, elegant low-energy solution to an engineering challenge
  - There is no ‘right’ way to solve any problem
  - Some people spend much energy on a complex solution, only to get frustrated when someone else finds a much simpler way
  - The simpler way is **more correct**; if you are struggling with your approach, maybe you need to rethink your assumptions?

PART 2

Assessment

My philosophy

- Engineering is the highest, purest and most noble pursuit of the human experience
  - All else is artifice or drudgery
- You are training to be engineers, and this is a chance to actually practice engineering
- You are not your grade*
- There will be second chances

* They make me assign you a grade

What to expect

- Expect to learn new things on your own
  - You need will need to know *more* than just what you've been taught at university thus far
- Expect to apply real effort
  - This course **actively** punishes freeloaders
- Expect to be involved
  - Lots of peer assessment; PAFs can be ***vicious***
- Expect change
  - The specifications will change (intentionally)

A common theme

- Present analysis to justify design decisions
  - – Motor torque/power calculations
  - – Chassis structural load simulations
  - – Clearance and tolerance of components
  - – Microcontroller control cycle overhead
  - – Decision matrices... and such!

If you can’t back up it up with numbers,
you‘re really just **guessing**

Deliverables

• Problem analysis	– 10%
• Progress Review 1	– pass/fail†
• Progress Seminar*	– 10%
• Progress Review 2	– pass/fail†
• Preliminary Report	– pass/fail†
• Final Product Demo*	– 60%
• Final Project Report	– 20%

* Team assessment with peer and tutor weightings

† More on this later

Problem analysis

Due 2nd March– 10% (2 pages max)

- Break down the design problem, determine its scope, requirements and constraints.
- Describe the key underlying engineering design challenges – what makes this hard?
- Present a candidate solution, and explain how your approach addresses the problem.
  - Analysis is golden.

Progress Reviews 1 and 2

Due 12th – 16th March and 8th – 11th May

- Tutor-mediated meetings
- Demonstrate your progress in the preceding period with tangible **evidence** of your contributions – eg. prototypes, code, etc
- Pass/fail mark based on quality of work and relative progress towards the goal

EXPECT NO MERCY.

Statement of roles

- At Progress Review 1, your team will be required to present a statement of lead developer roles signed by the whole team.
- You will be expected to account for your responsibilities at subsequent reviews
- *Remember:* You don’t have to be the only monkey working on your system, and you should help others!

Progress Seminar

Due 10th – 14th April (team assessment) – 10%

- Provide a 10 minute seminar outlining progress towards developing a solution to the problem.
  - Focus on the progress, not the approach
  - Each student presents for roughly equal time
- Assessed by course coordinator and tutors

Preliminary Report

Due 18th May

- Describes the methodical analytical approach to solving your subtask, how it relates to the other subsystems within the project and the analytical process that was used in developing the solution.
- Show the formal, disciplined, quantitative engineering process followed, demonstrating the feasibility of the approach taken.

Final Product Demo

Due week 13 (team assessment) – 60%

- The Main Event – show your system works!
- Marks awarded for functionality, achievements and build quality.
- Hand in everything needed to make your system work, including documentation and printouts of design schematics.

Above all: Convince me you can *engineer*.

Final Report

Due 1st June – 20 %

- Identical to the preliminary report, but incorporating corrections and reflecting any changes from the final two weeks.
- Preliminary report will be returned with comments so that you have an opportunity to revise your work and improve upon it,

Just like in real life!

Incremental demos

- Spontaneous night-before failure of hardware systems is **brutal** and **unfair***.

Just like real life!

- If your system is sort-of working early, you can have it tested in an incremental demo.
  - If the final demo mark is less than what was scored in an incremental demo, you will be awarded the incremental demo mark.

Incremental demos

- Scored just like the final demo, but the mark is capped according to time left in semester
  - Week 7: 25%
  - Week 9: 50%
  - Week 11: 75%

- Incremental demos are by appointment only*

* Do not attempt a demo with an obviously non-functional system or you may forfeit future incremental demo privileges

Pass/fail penalties

- Subpar (or absent) pass/fail submissions incur a deduction from your final grade
  - Project reviews: 5% each
  - Preliminary report: 10%

- These deductions are *cumulative*
  - If you were to fail all of them, your maximum achievable grade for the course would be 80%

PAF and TAF

- A substantial fraction of assessment is peer-moderated; others are tutor-moderated
  - Regularly adjusts results by up to **2 grades**

- It’s vital your team recognises your efforts
  - A bitter or frustrated team means a low PAF!

- Ultimately, peer and tutor weighting is mediated by the course coordinator

Peer assessment

- At progress reviews, progress seminar and final demo, you will fill out PAFs
- Your demo mark will be scaled by all of the PAFs, according to a weighting scheme:

– Progress review 1:	10%
– Progress seminar:	20%
– Progress review 2:	30%
– Final demo:	40%

Calendar at a glance

You are here

Teams assigned here

Week	Dates	Lecture	Reviews	Demos	Assessment submissions
1	19/2 – 24/2	Introduction			
2	26/2 – 2/3	Principles of Mechatronic Systems design			Problem analysis
3	5/3 – 9/3	Professional Engineering Topics			
4	12/3 – 16/3	PCB design tips	Progress review 1		
5	19/3 – 23/3	Your soldering is (probably) terrible			
6	26/3 – 29/3	Introduction to firmware design			
Break	30/4 – 13/4				
7	16/4 – 20/4	Switch to	Progress seminar	25% demo	
8	23/4 – 27/4	Q and A			
9	30/4 – 4/5	sessions		50% demo	
10	8/5 – 11/5	No lecture	Progress review		
11	14/5 – 18/5			75% demo	Preliminary report
12	21/5 – 25/5				
13	28/5 – 1/6	Closing lecture		Final testing	Final report and reflection

Hyoooge break! Try to work ☺

Madness week

PART 3

Class Organisation

Heads up

- I had intense surgery at the end of November
  - They opened me up, pulled out my insides, shuffled them around, and put them back in.
  - Bonus: They let me keep the bits they took out!
- This course was designed while in hospital, under the influence of *serious* painkillers.
- If anything in the course doesn’t make sense, it’s probably due to that – just let me know!

Blackboard and splashy website

- This class has a Blackboard page and a "splashy" outwards-facing website
  - If the two ever differ (which they won't), the Blackboard page is considered authoritative

Blackboard:	learn.uq.edu.au/
Splashy:	robotics.itee.uq.edu.au/~metr4810/

Weekly schedule

- Lectures – 2 hours once per week
  - – Professional topics and Q&A sessions
- Practicals – 2 hours ~~twice~~ once per week
  - – Tutors available in lab (but 24/7 access)
- "Contact" – 2 hours ~~twice~~ thrice per week
  - – Time set aside for meetings, demos, etc.

Your team should meet and interact continuously outside of class – *at least* once per week

Weekly schedule

- I asked for two practical sessions spread across two rooms and I got two sessions each in one room… at the same time.
  - I was too high on morphine to notice the problem when the draft was published*

- We’ll fix it by making the Wednesday contact slot a practical session instead.

*From the surgery, I promise!

Class clashes

I am aware of some clashes with other classes

- Most notably METR4900
  - This will ruin your life – plan accordingly
- Any others I’ve missed?

All lecture content will be online

Major announcements go out via Blackboard

… but you’ll hear it first in class, by design

Lectures

- Boring, useless lectures help *nobody*
- I will endeavour to provide lectures that are educational, useful and (sort of) entertaining
- Lectures will be student-driven: you tell me what you want to learn about and I’ll teach it

Lectures

- Lecture 1: Introduction to the project
- Lecture 2: Principles of mechatronics system design
- Lecture 3: Professional engineering topics
- Lecture 4: PCB design tips
- Lecture 5: Your soldering is terrible (probably)
- Lecture 6: Introduction to firmware design

Topics may be nominated by emailing me, and then voted for on a doodle poll

Lectures

- No, you don’t *have* to attend lectures, but if you don’t you’re really missing out
  - Protip: Students who attend lectures historically do better than those who don’t!
- Lectures are the first and most immediate way of hearing about what’s happening and getting your questions answered
  - Note: recordings aren’t interactive

Some suggested topics

- Principles of teleoperation control
- Orientation control on SO(3)
- Digital control
- Electromechanical devices
- Computer vision
- Sensor-fusion and filtering
- Localisation
- Schopenhauer and philosophical pessimism

Teams

- Teams will each consist of four people
  - Except for when they don’t
- Teams will each be assigned a tool kit
  - Complete kit must be returned ***or else***
- Work together! Contact sessions are set aside for team meetings and collaboration

Teams

- You will have to work with people you hate*

Just like in real life!

- You may email me and request one person with whom you do not want to work
  - Exclusion requests must be in by Friday
- Otherwise, teams will be allocated by *magic*
  - Teams will be assigned in week 2

*If you don’t hate them now, you will by the time you’re done

Laboratory space

- A reasonable number of students this year?
  - I’m as surprised as you are...
- Consequence: hopefully less space pressure
  - New (smaller) lockers for project work
  - Shared space and resources
  - Get started early; consider how you can work most effectively in the final two crunch weeks

<eyeofsauron>

Hey, about that lab...

Laboratory space

- The laboratories are governed by the UQ risk management policy
- To work in the lab:
  - You **MUST** have completed the induction
  - You **MUST** have read the lab risk assessment
  - You **MUST** wear appropriate footwear
  - You **MUST** abide by all safety requirements
- If you do not follow the guidelines you will be barred from the lab

Laboratory space

- Just in case you forgot:
  - No eating/drinking in the lab
  - No sleeping in the lab
  - No non-METR4810 students in the lab
  - The lab is not for facebook/tindr/grindr/gaming/ socialising/having a life etc.
  - <u>I am held personally responsible for the safety and condition of the lab and I get ***very*** grumpy.</u>

So don’t say you weren’t told.

Laboratory space

- Every year, I go out of my way to find ~~people~~ violators to make an example of
  - Don’t let this happen to you!

Proposed lab management policy

Laboratory space

- Keep the lab clean and orderly
- Cleanliness “warning light” system in effect
  - Status noted on Blackboard/class website

Green: Full speed ahead

Yellow: Clean up needed

Red: *Danger, Will Robinson!**

Black: “*Uh oh.*”**

*Lab will go to limited hours until cleaned.

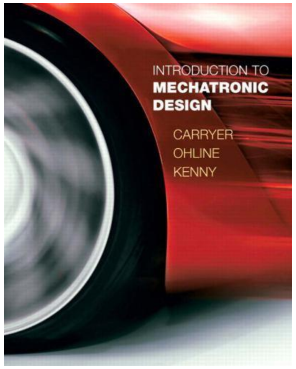
**Lab will be locked until further notice.

</eyeofsauron>

Keeping the lab tidy makes for a nicer place to work and makes it easier to get stuff done

Resources

- Website
  - Everything will be posted on the Blackboard class website: (learn.uq.edu.au)
  - Better-looking class website will mirror course materials: (robotics.itee.uq.edu.au/~metr4810)
- Textbook
  - “Introduction to Mechatronic Design” by Carryer, Ohline and Kenny



(recommended but not required)

Knowledgeable people

- Course Coordinator and Chief Conspirator:
  - Paul Pounds

- Technical Staff
  - Peter Bleakley
  - Ray White
  - Grant Tayles

- Tutors:
  - Iain Rudge

- Emergency Auxiliary Temporary Back-Up Replacement Stand-in Teaching Faculty
  - Dr. Michael Kearny
  - Prof. Stephen Wilson

Contact info

If anything is bothering you, bring it up *early*

- Rules questions
- Technical issues
- Ordering
- Assessments
- Group problems
- Enrolment
- Disenfranchisment with the sociopolitical gestalt

➢ Serious? Email first to arrange a meeting

➢ No? Just stop on by! (but email is good too)

On that topic…

- I often get comments in the SECaTs about things that *could* have been addressed during the semester if I'd been told about it earlier

- Don't wait until you're angry in week 13
  - Tell me about it as soon as it comes up so I can explain it/solve it/fix it/find it/sort it right away

- I'm always happy to help! ☺

Contact info

Who: Me!

Why: Questions, issues, concerns, ennui!

Where: GPS 78-529 or Wordsmiths

When: 10ish to 4ish – by appointment (or drop in)

What: Coffee or coke (either kind)

How: paul.pounds@uq.edu.au

What happens next?

- Send me group exclusion requests
  - – Email me ASAP!
  - – Groups will be posted next week
- Attend the afternoon practical session in Hawken c404 Thursday 8th March
  - – Toolbox handouts
  - – Room induction, 3D printer induction

And start thinking about solutions!

Tune-in next time for…

Principles of Mechatronic Systems Design

or

“Striking a Balance is Making Everybody Equally Unhappy”

Fun fact: John Tebbutt was one of Australia’s first and most famous astronomers. He built his telescopes himself, by hand.

Questions?

?