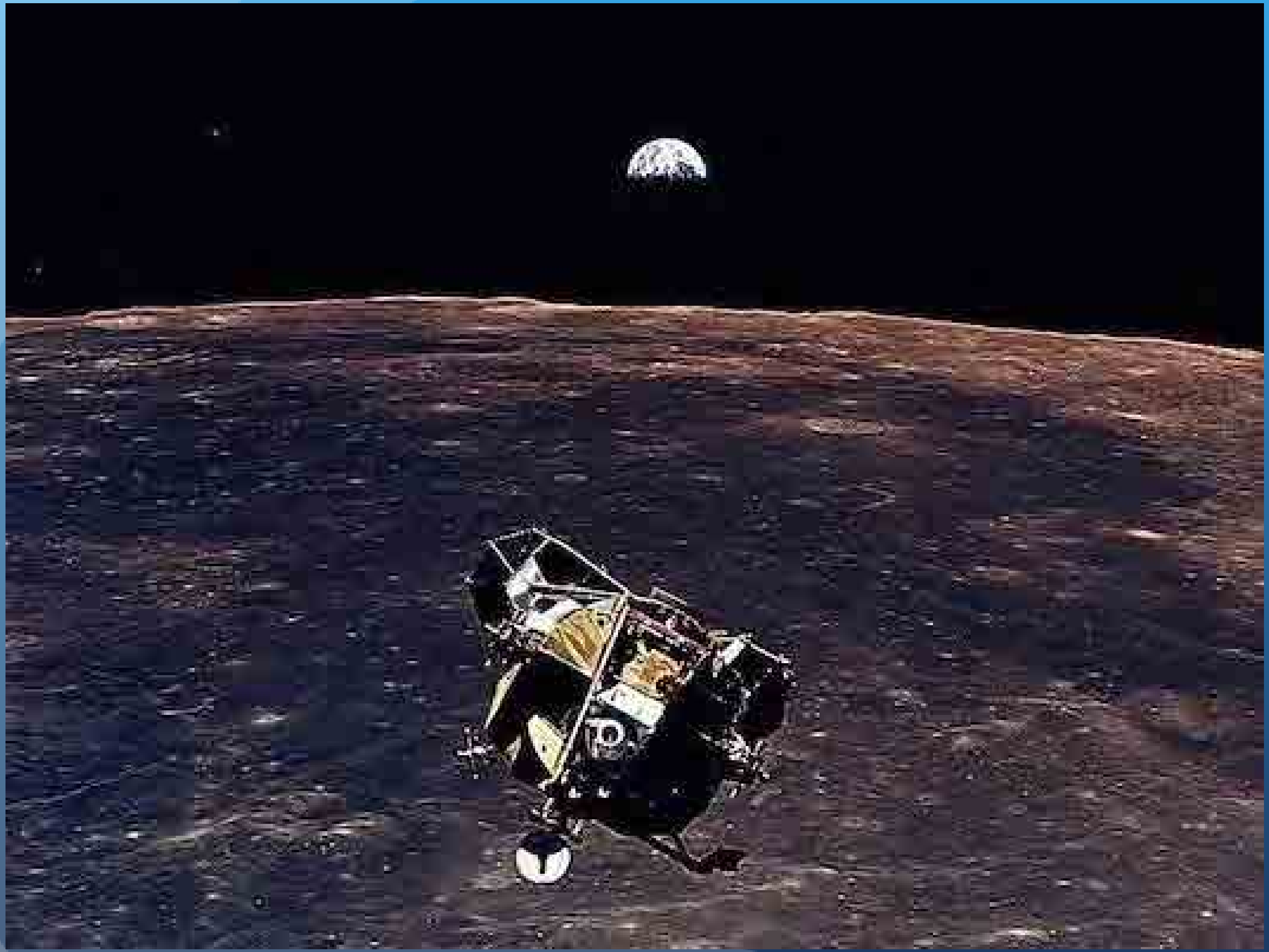
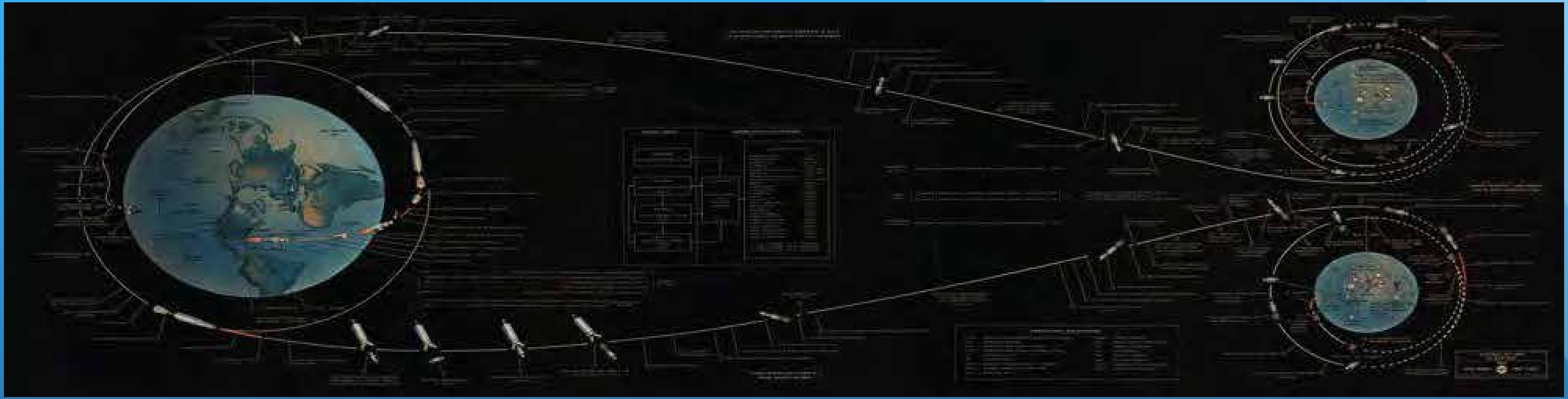
Fifty Years Beyond the Giant Leap

Spacecraft Navigation from Apollo into the 21st Century



How to reach the Moon?



- A system combining inertial navigation, the Apollo guidance computer (AGC), and on-board flight software
- Milestones - performance of this system and its legacy:
  - Apollo 8: leaving Earth to orbit the Moon
  - Apollo 11: first lunar landing
  - Highlights of Apollo 12-13
  - Spaceflight today: innovations and the legacy of Apollo

Goal: avoid extremes

So what? I learned that watching PBS

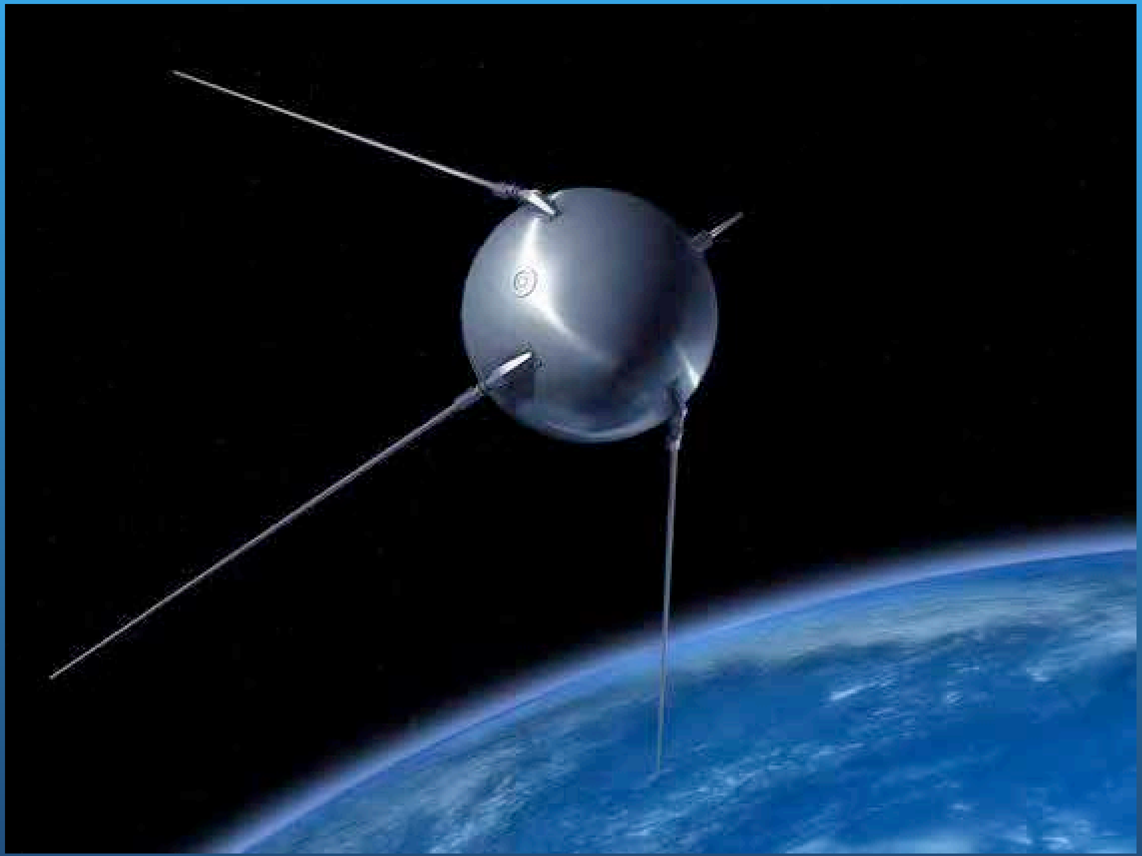
AIO

Apollo information overload

GNC
Guidance, navigation, and control

- Guidance: How do we reach a destination?
- Navigation: Where are we now? In what direction are we moving?
- Control: What do we tell the spacecraft to do?

1957

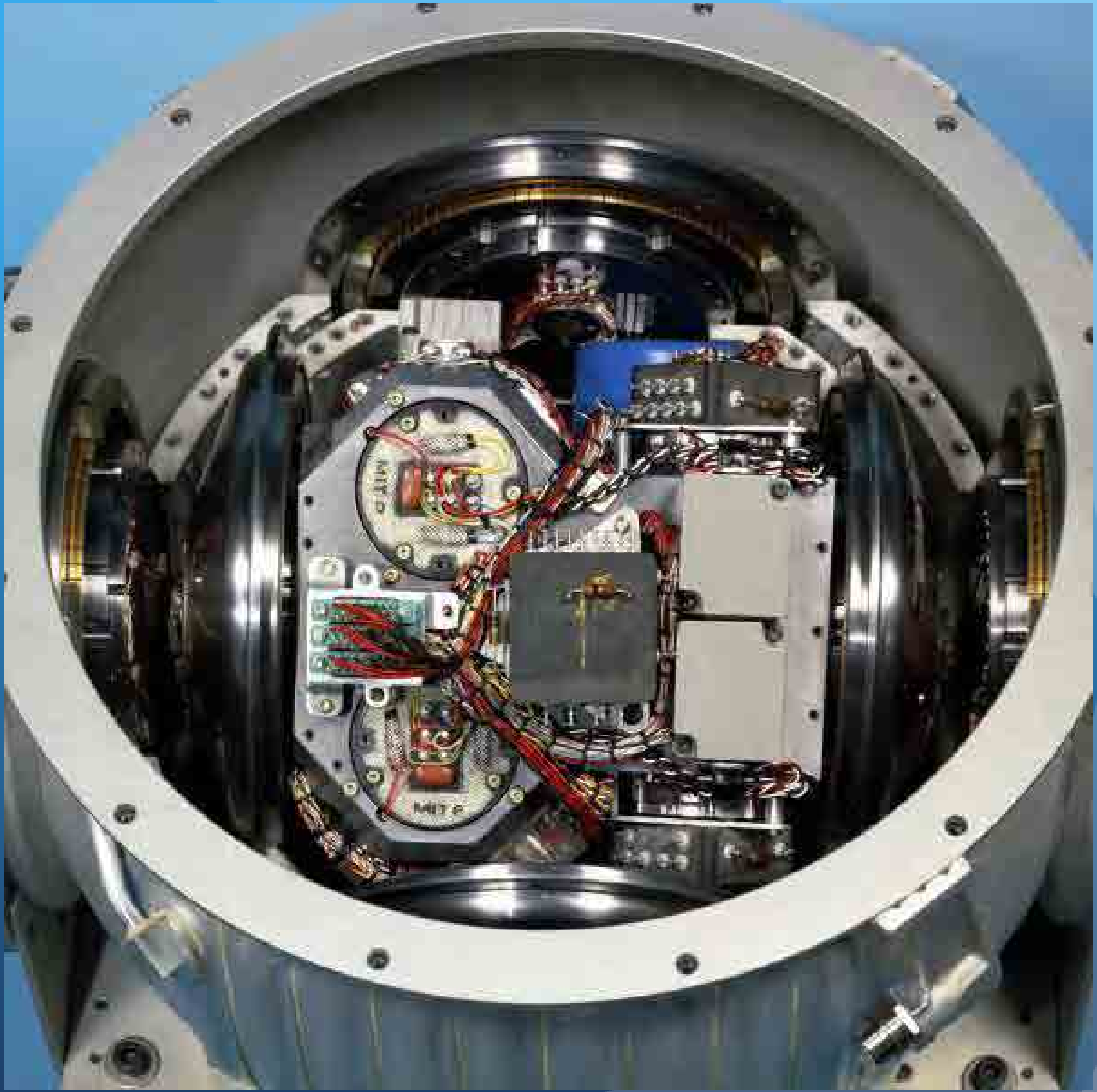


Sputnik

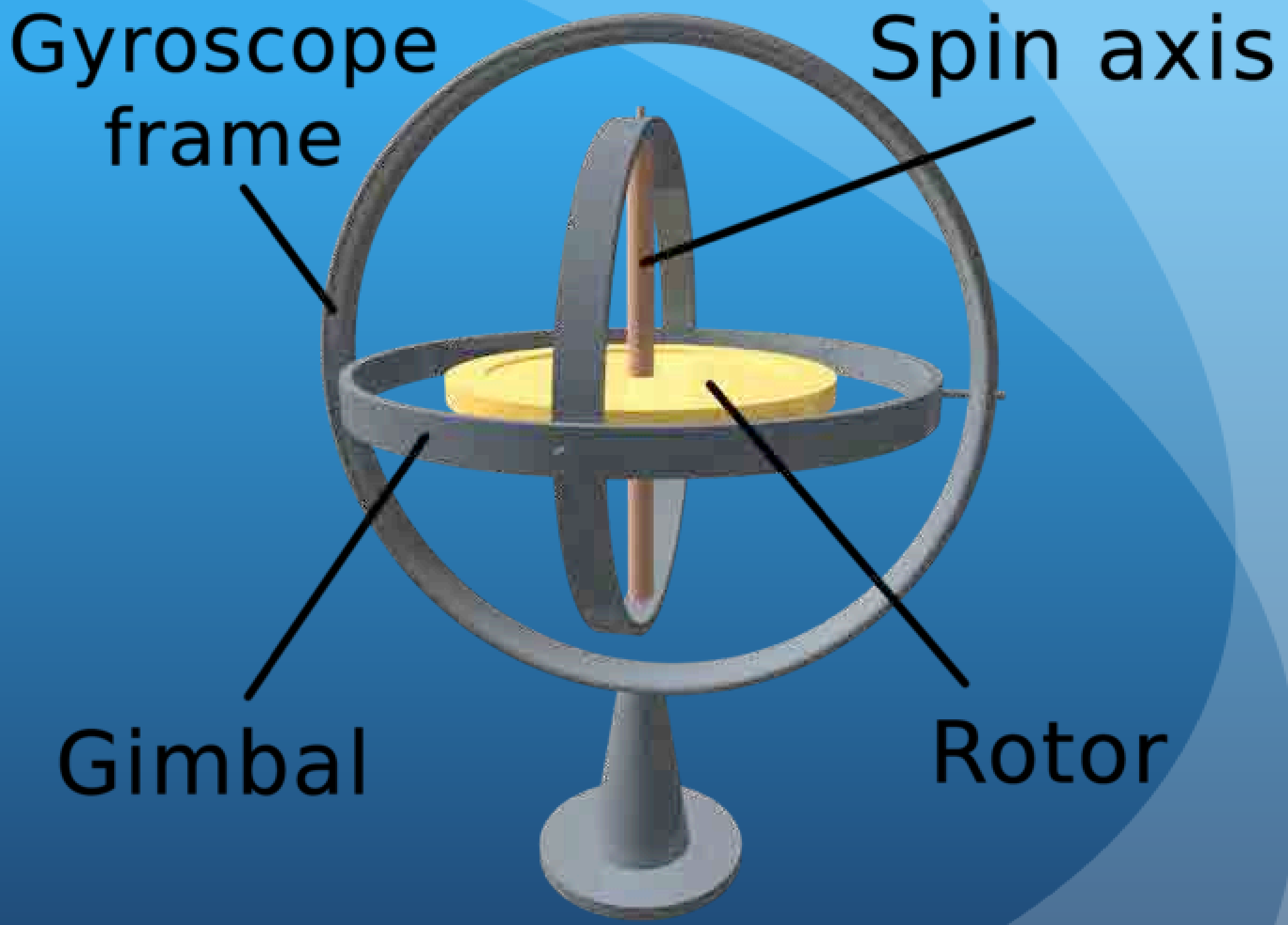
Mars Probe Model

Inertial navigation system (INS)

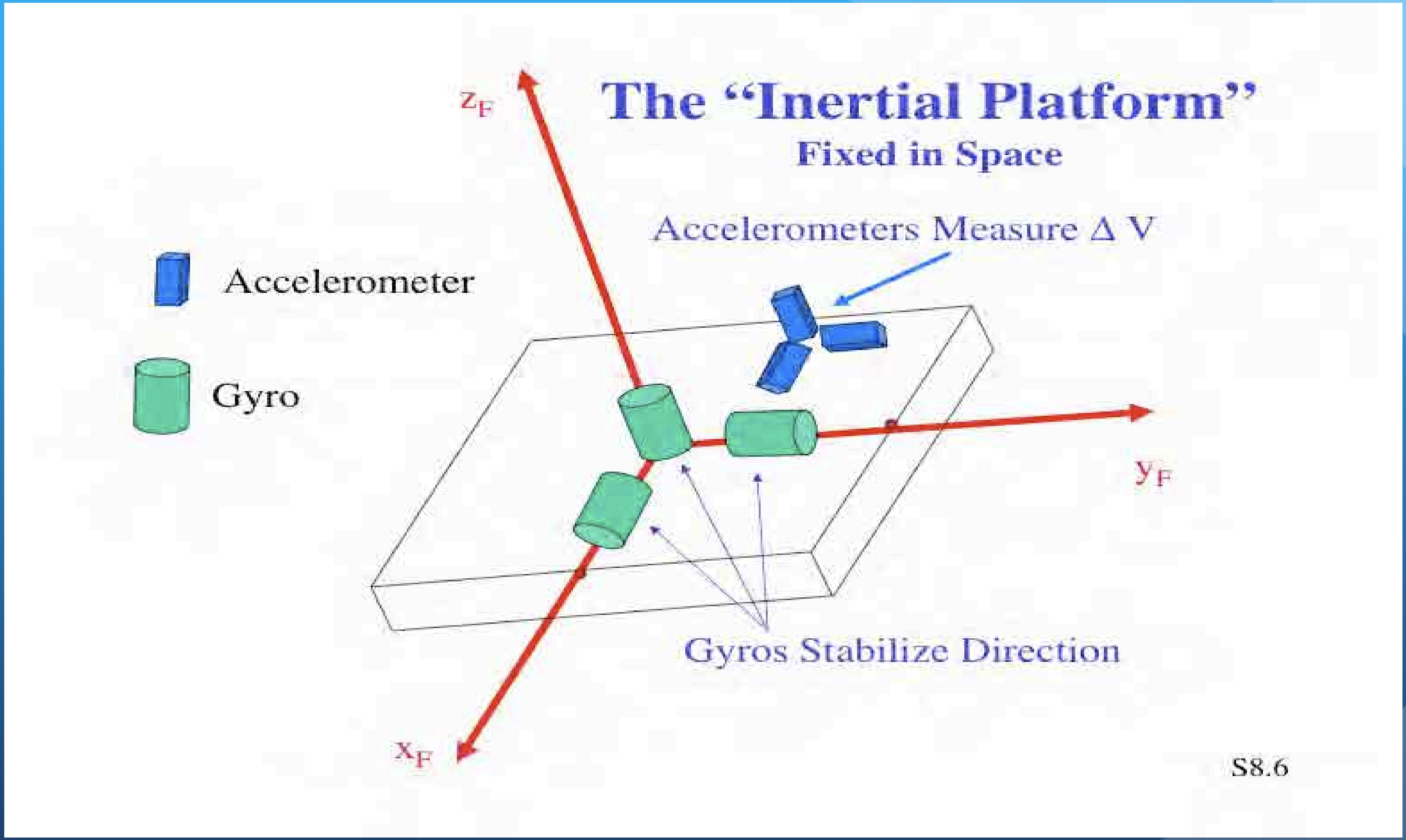
Apollo inertial measurement unit (IMU)



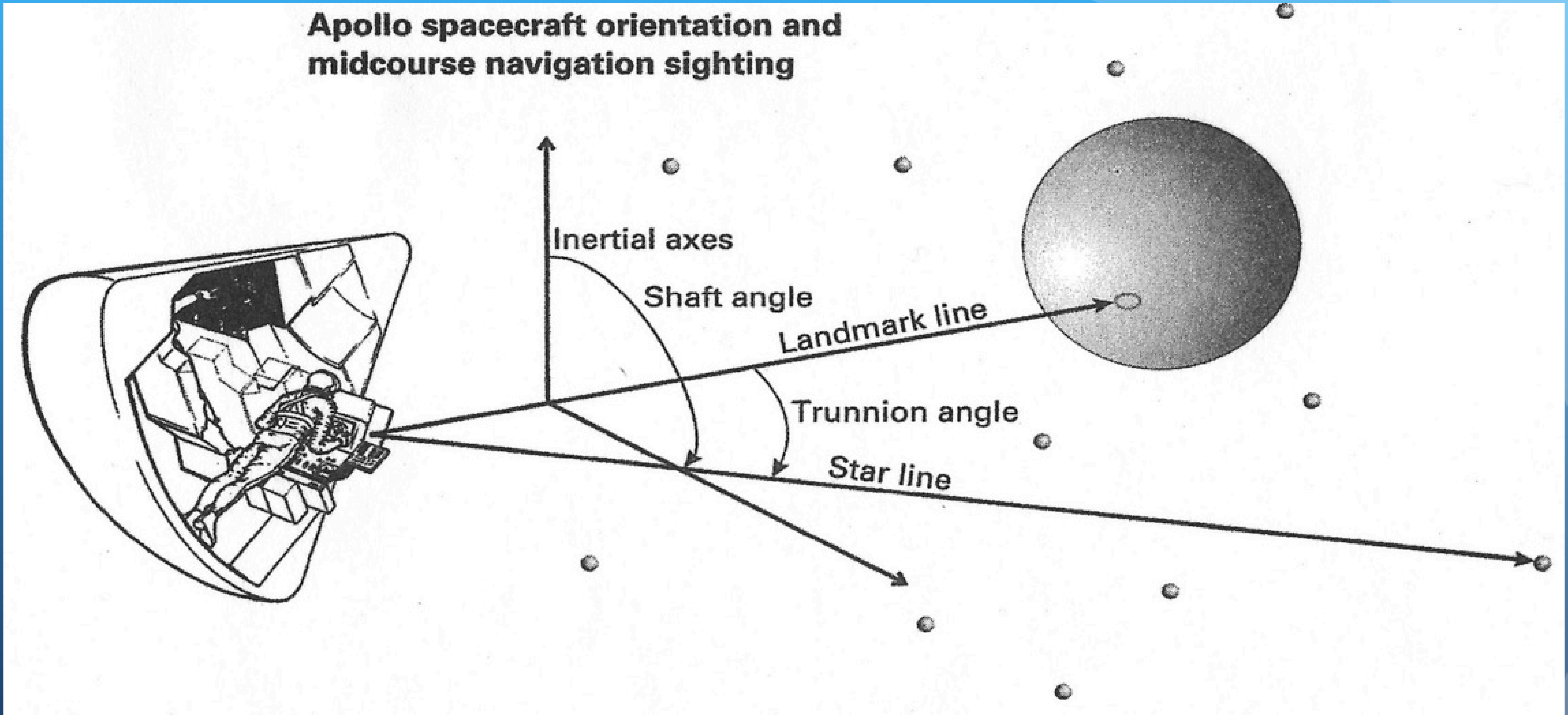
Key instrument: gyroscope



Key instrument: accelerometer


The "Inertial Platform"
Fixed in Space
Accelerometers Measure Δ V
Accelerometer
Gyro
z_F
y_F
x_F
Gyros Stabilize Direction
S8.6

Backup: the space sextant



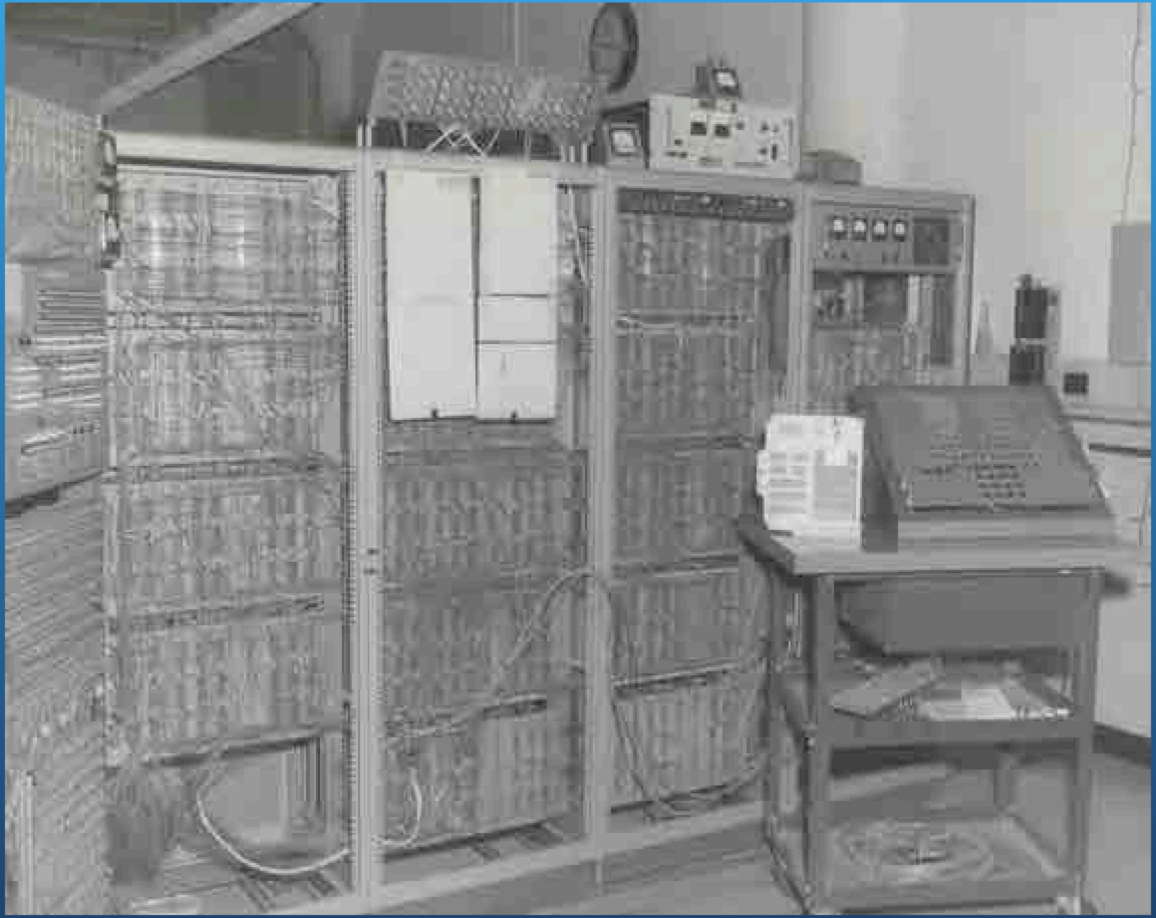
The Apollo guidance computer (AGC)

"The fourth astronaut"

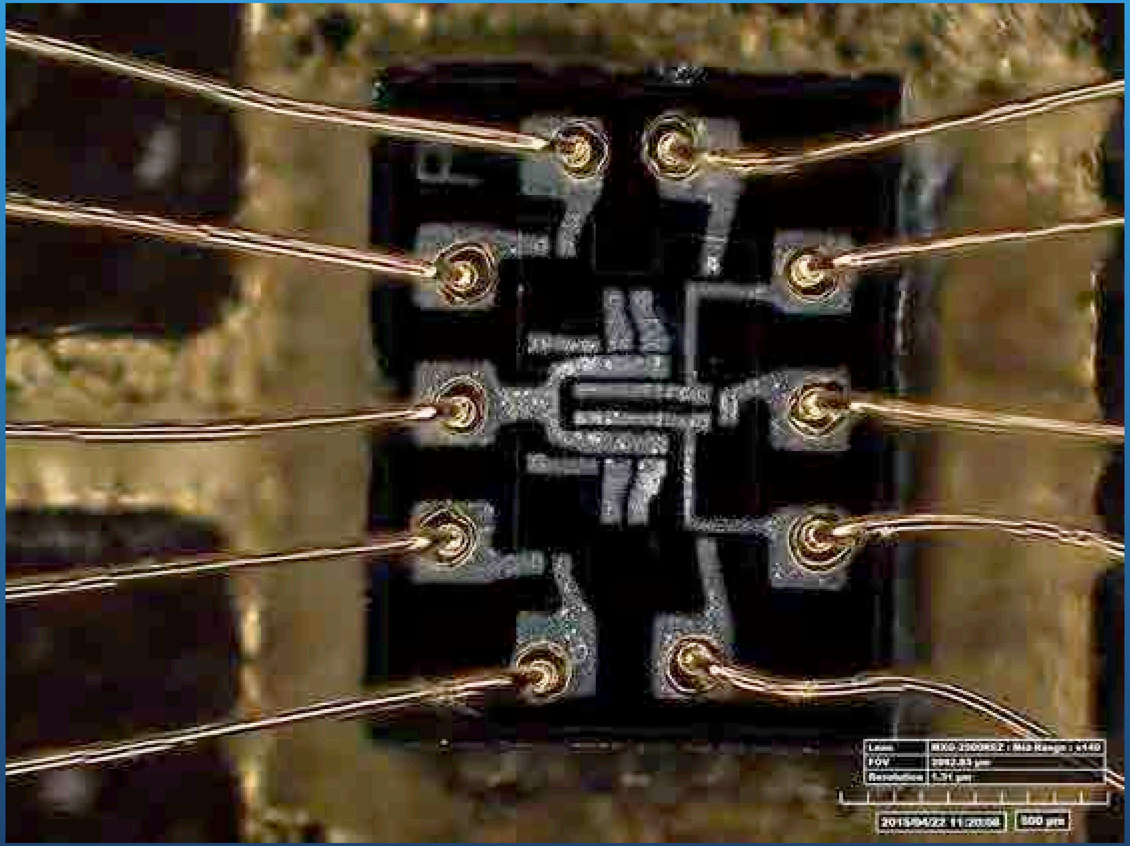
Essential AGC features

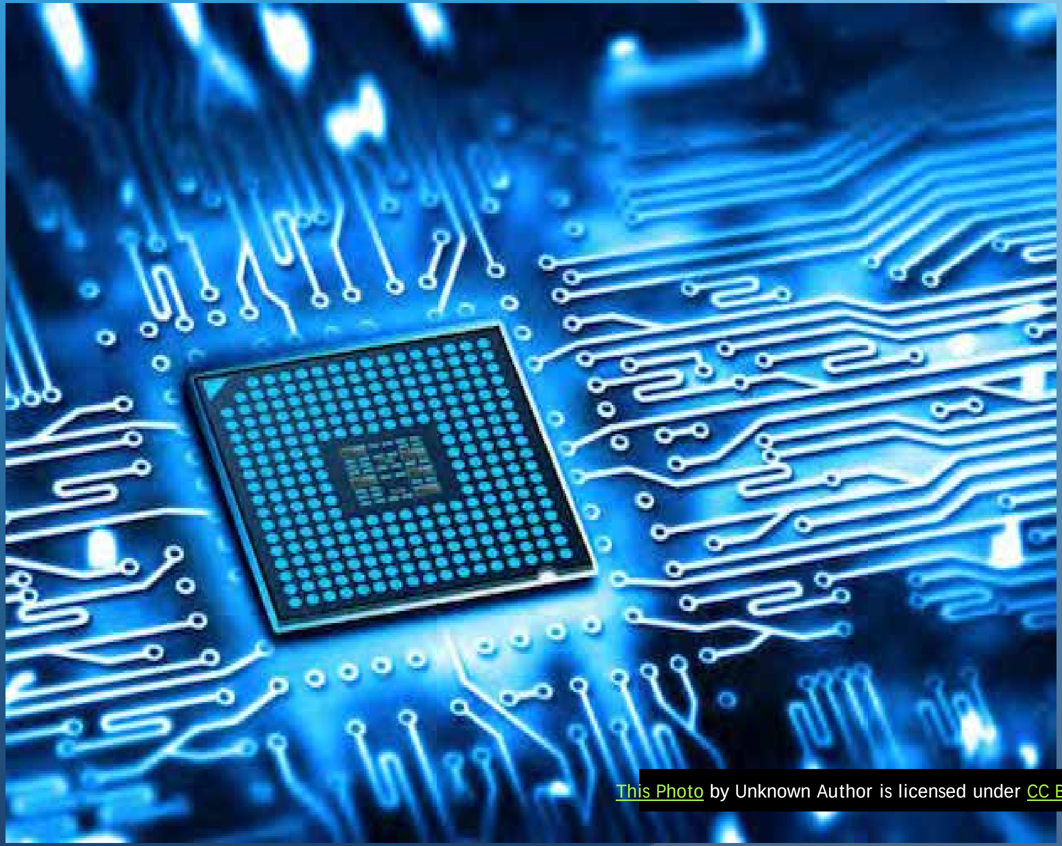
- Design (size)
- Interface (use of language)
- Reliability (integrated circuits - memory - software)

The AGC
from model to the Moon



Design and reliability
integrated circuits





Risk

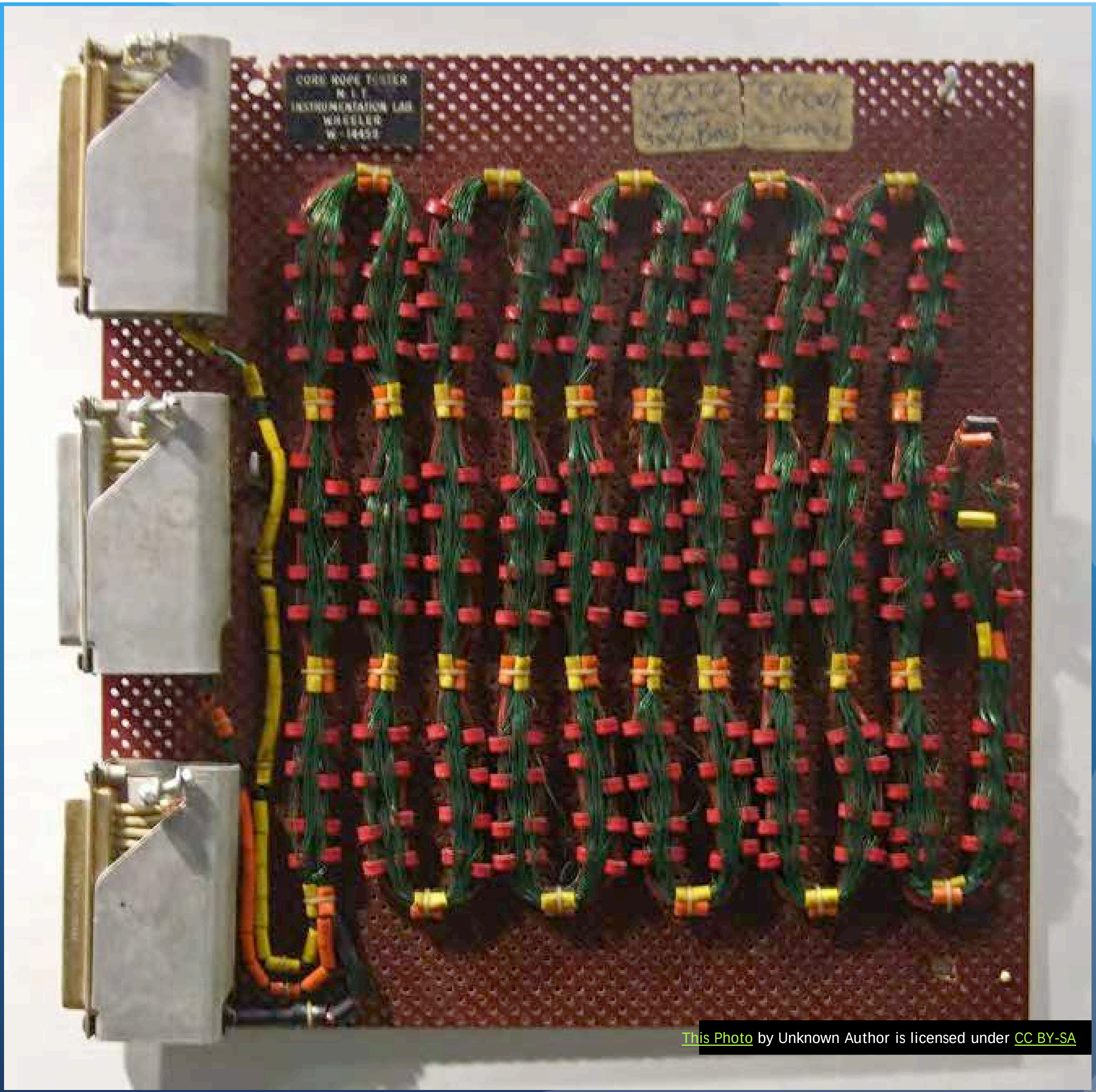
- AGC: 1st digital, portable, general purpose computer
- Apollo: 1st time lives depended on computer performance: if AGC failed, *astronauts could die.*
- Safety target (astronauts survive): 99.9%
- Reliability target (mission succeeds): 99%

“It *had* to work.”

Margaret Hamilton, Director of Apollo
On-Board Flight Software Development, MIT-IL 1968-1972

Reliability:

Core rope, read-only memory (ROM) 72 KB



Making core rope memory

Interface: Display Keyboard (DSKY)

- 19 keys that astronauts used to enter commands: verb + 2 digits, noun + 2 digits, Enter

Sample DSKY commands for lunar landing

- V37N63 = *run powered descent program (start of lunar landing)*
- V16N17 = *display attitude*
- V16N68: *display range-time to target*
- Estimate: 10,500 keystrokes round-trip

Reliability: on-board flight software

- Benefits: saves space and weight, connects systems (displays and switches, engines), and automates maneuvers (software: *digital autopilot /* system: *digital fly-by-wire*)
- Risks: Software is new and complex, hard to manage; it must detect problems and recover in real time; software errors or bugs can abort a mission or kill the astronauts
- Laugh lines: soft-w-e-a-r and *Please don't tell our friends*

How to balance multiple AGC operations?

- 1960s computers:
  - All operations have = priority, execute sequentially (like a washing machine)
- AGC software prioritized operations:
  - Higher-priority operations could interrupt those with lower-priority (asynchronous - unpredictable)

What power failed?

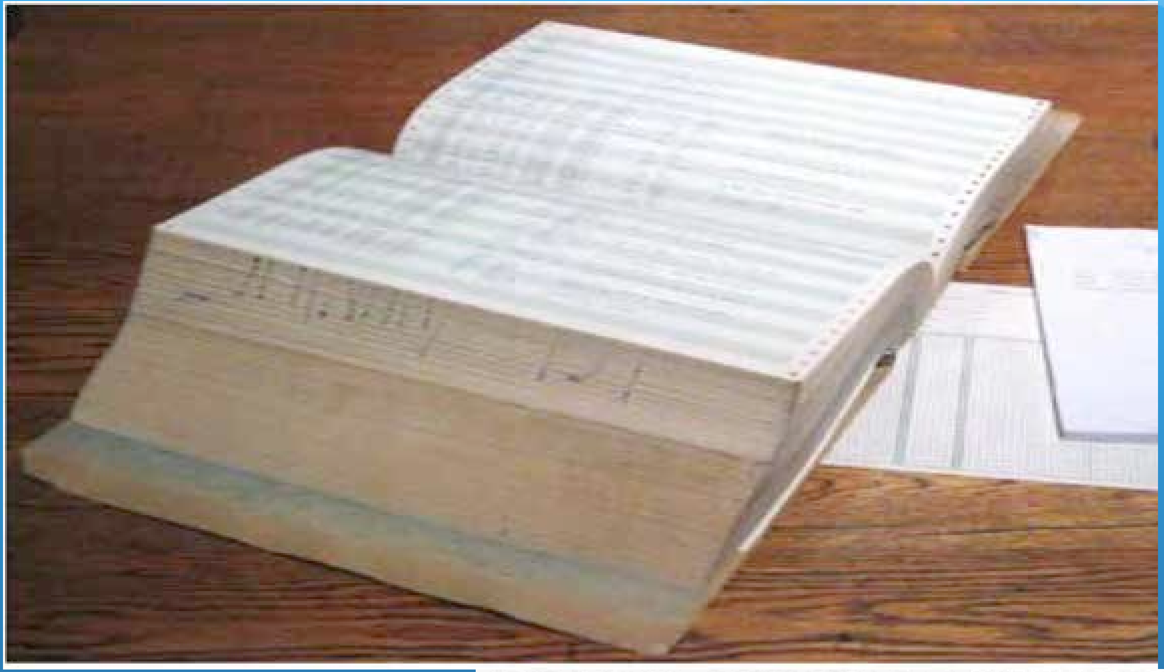
restart protection

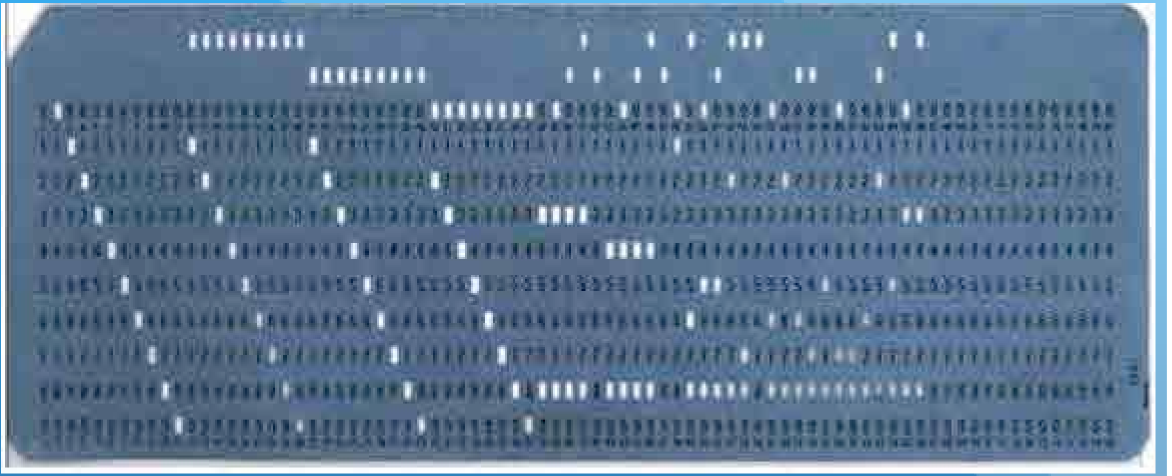
- Software can restart, two words in a line of code for vital operation allow it to resume:

register waypoint

- *"Giving the computer an enema"*

Software testing





Writing AGC code: does this sound familiar?

- *There are as many ways to write a 10-word line of code as a 10-word sentence.*
- *You need a unifying aesthetic that becomes a personal style and that is something that may keep evolving throughout a long career.*

Don Eyles, *Sunburst and Luminary: An Apollo Memoir,* page 59

Early software engineering

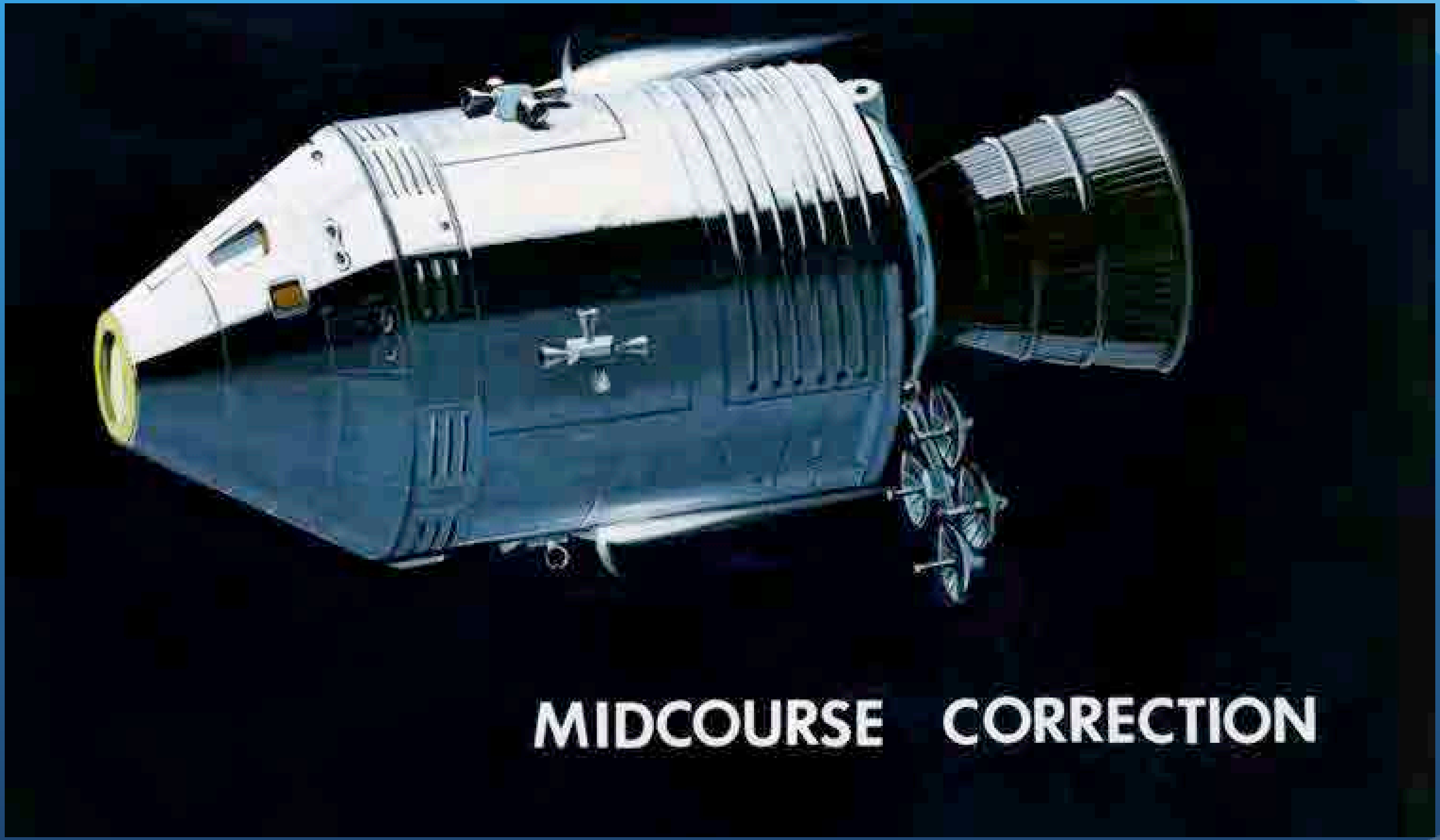
Margaret Hamilton to BBC – *13 Minutes to the Moon*

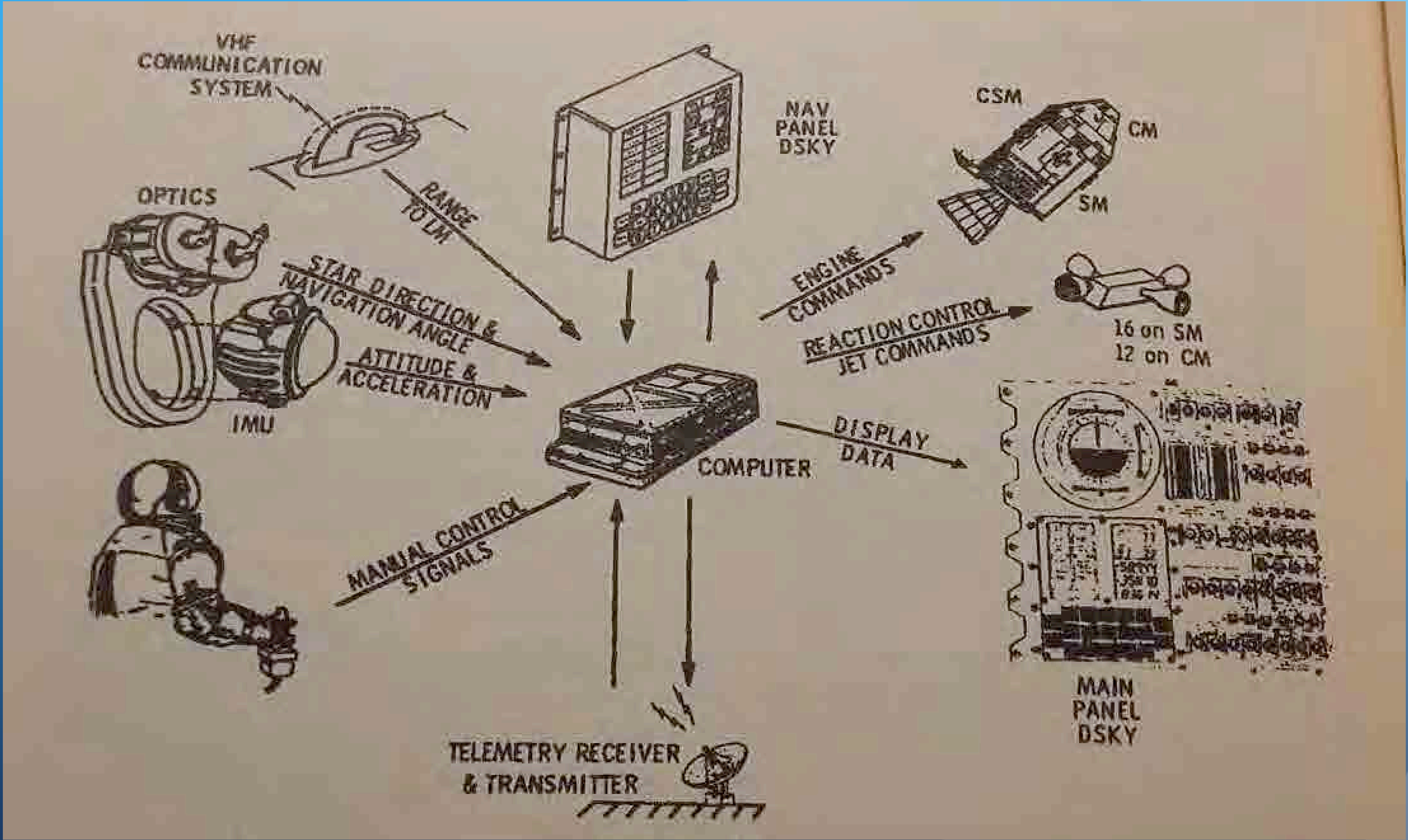
We tried to understand the errors and find new ways to prevent them. I began to worry about everything working together, what the astronaut might do by mistake. What would happen as a result of hooking up the wrong radar? The radar in the wrong position could affect the software. How do you let them know when they're busy and don't even notice there's a problem?

The big 3: *Wrong data, wrong time, wrong priority*

Apollo 8
Leaving Earth to orbit the Moon

Service - Command Modules
SM-CM (CM + SM = CSM)

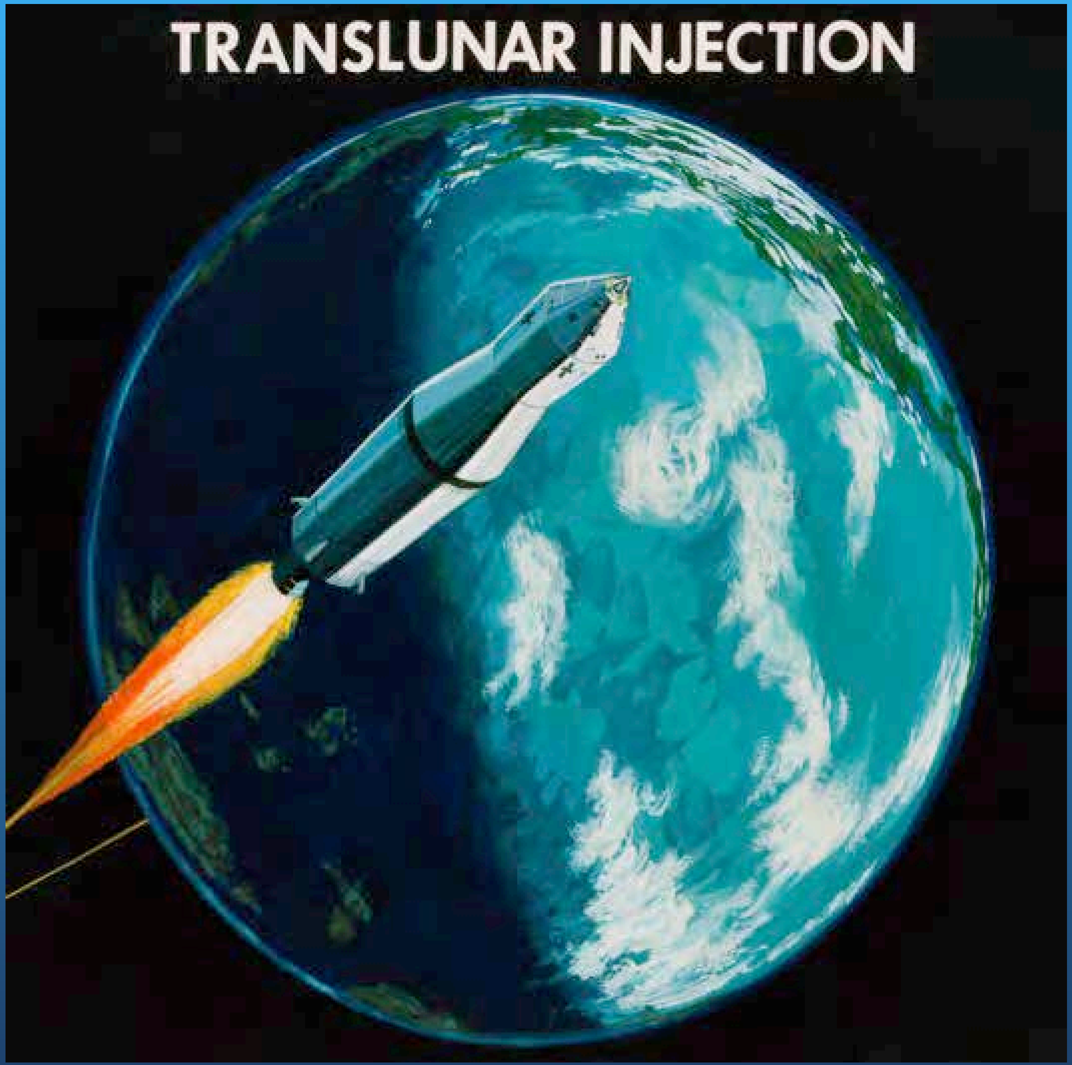



VHF
COMMUNICATION
SYSTEM
RANGE
TO LM
NAV
PANEL
DSKY
CSM
CM
SM
OPTICS
STAR DIRECTION &
NAVIGATION ANGLE
ATTITUDE &
ACCELERATION
IMU
ENGINE
COMMANDS
REACTION CONTROL
JET COMMANDS
16 on SM
12 on CM
DISPLAY
DATA
COMPUTER
MANUAL CONTROL
SIGNALS
MAIN
PANEL
DSKY
TELEMETRY RECEIVER
& TRANSMITTER

What the AGC did

- Collect data from IMU, other on-board sensors, switches, and controls; and from ground transmissions
- Run software that guides, navigates, and controls the spacecraft
- Display information to the crew
- Send telemetry data to the ground
- Command engines, maneuvering jets, antennas, other components
- “...write the plot, apply the higher-level logic that pulls the strings to accomplish the mission.”

Translunar injection (TLI)



AGC performance

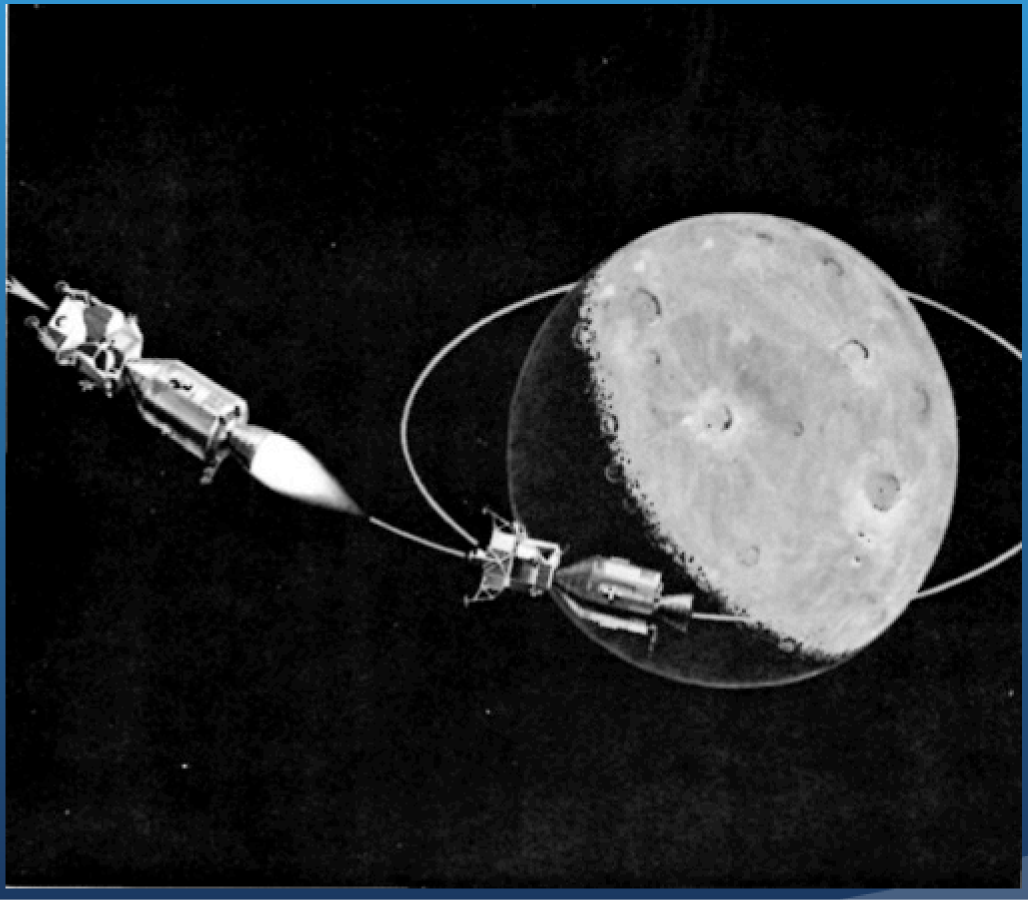
- Jim Lovell used sextant > 100 times check AGC navigation
- NASA compared this data to ground tracking:

“essentially identical”

Lunar orbit insertion (LOI)

Target orbit: 170 x 60 miles

Actual orbit: 169.1 x 60.5 miles

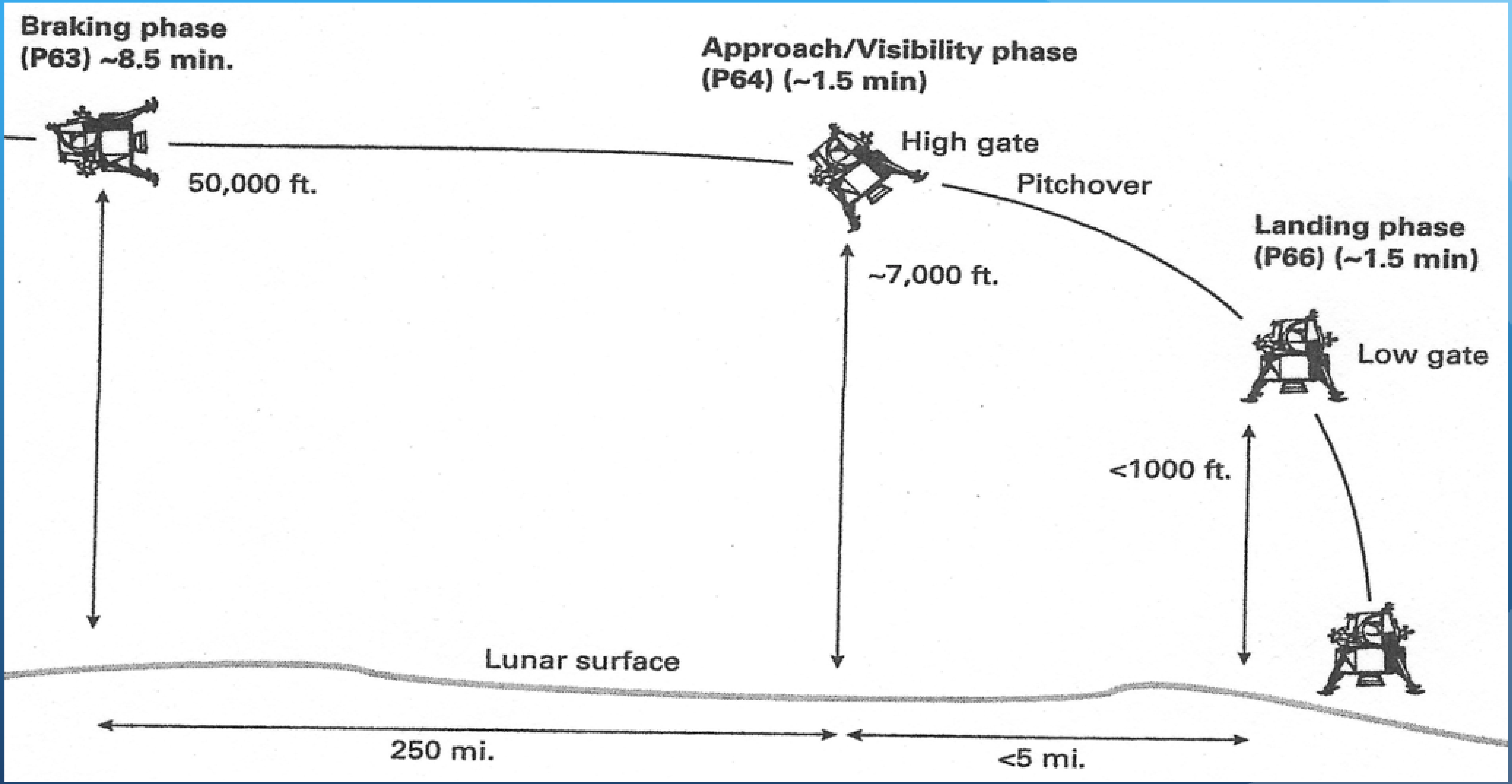


SCL-22 APOLLO 8 SPACECRAFT IN ORBIT AROUND THE MOON VIEWS THE EARTH

Next steps

- Apollo 9: testing Lunar Module in Earth orbit
- Apollo 10 – “The dress rehearsal:” orbited Moon, LM descended to 47,000 ft. above surface but did not land

Apollo 11: reaching the Moon



DSKY display 2:19 before landing

COMP ACTY	PROG	
	66	
VERB	NOUN	
06	60	What the numbers mean
+ 00601		HORIZ. VELOCITY (XXXX.X FT/SEC)
- 00100		ALTITUDE RATE (XXXX.X FT/SEC)
+ 00410		ALTITUDE (XXXXX FT)

1201 - 1202 program alarms
What was the LGC doing?

- Hardware problem: rendezvous radar sending pulses of meaningless data to LGC
- This occurred in 2% of simulations and a few engineers in Houston recognized it, astronauts didn’t know what alarms meant
- Data pulses consumed 13% of LGC capacity, overloading computer and triggering

restart protection

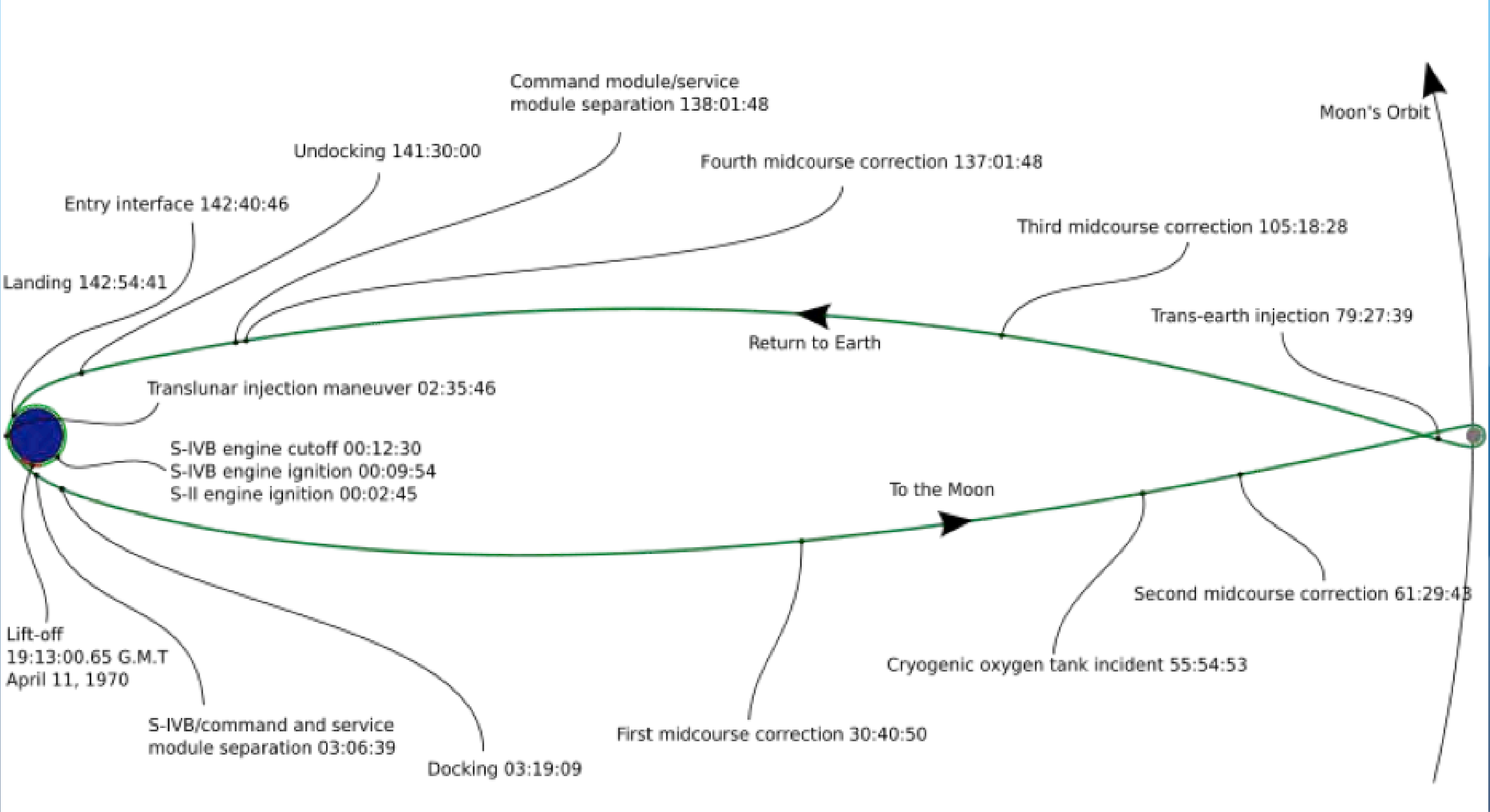
- LGC continued descent uninterrupted - *real-time problem detection and recovery*


Aviation Week
& Space Technology
A McGraw-Hill Publication $1.00

Apollo 12: Lightning strikes twice (in less than a minute)

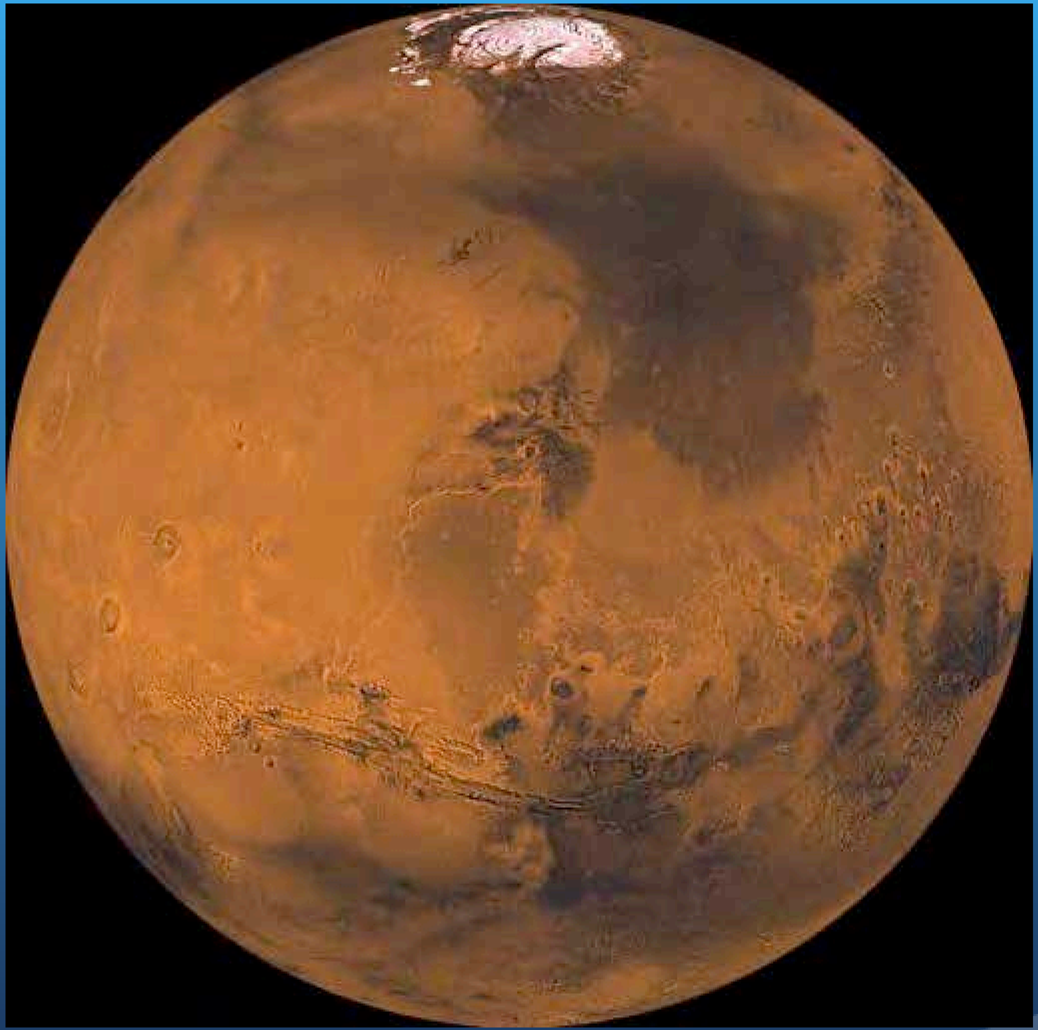
Apollo 13

Apollo 13 rescue trajectory



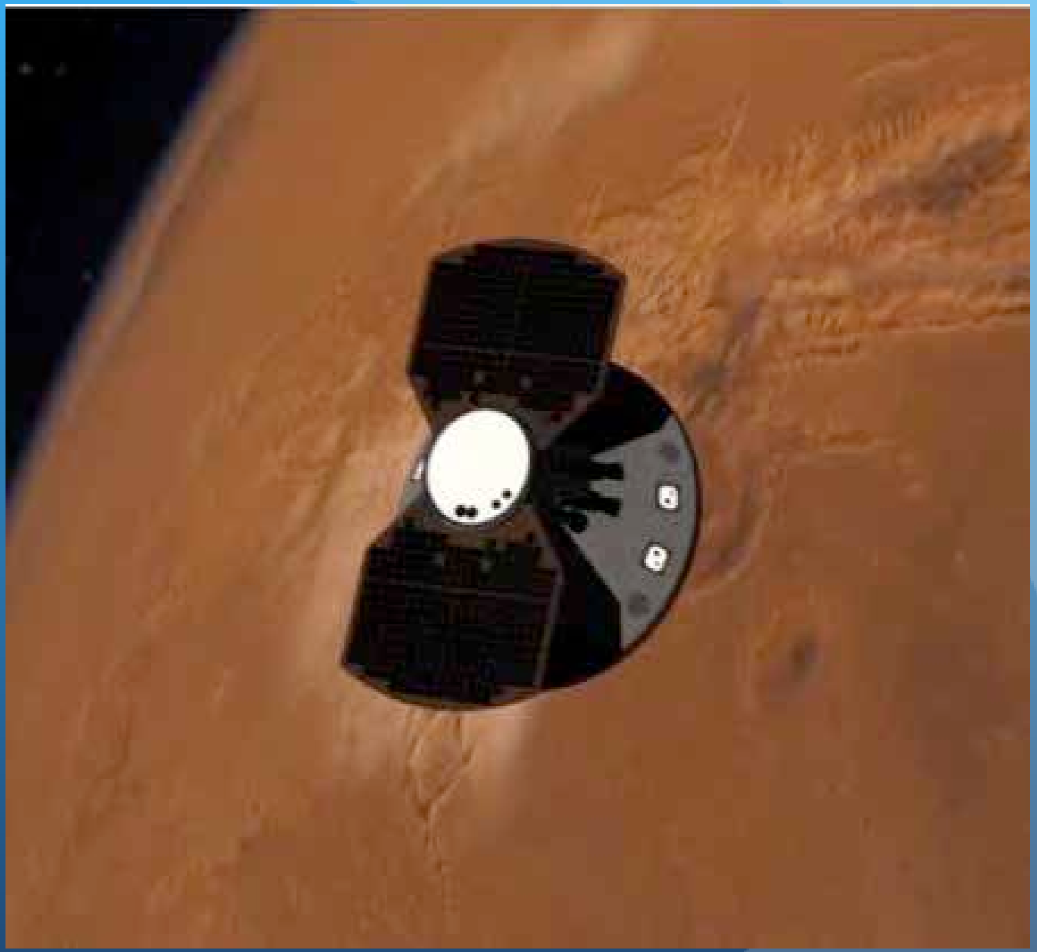
Leaving Earth is risky, challenging

1962-2018: 22 of 45 attempts to fly by, orbit, or land on Mars have failed



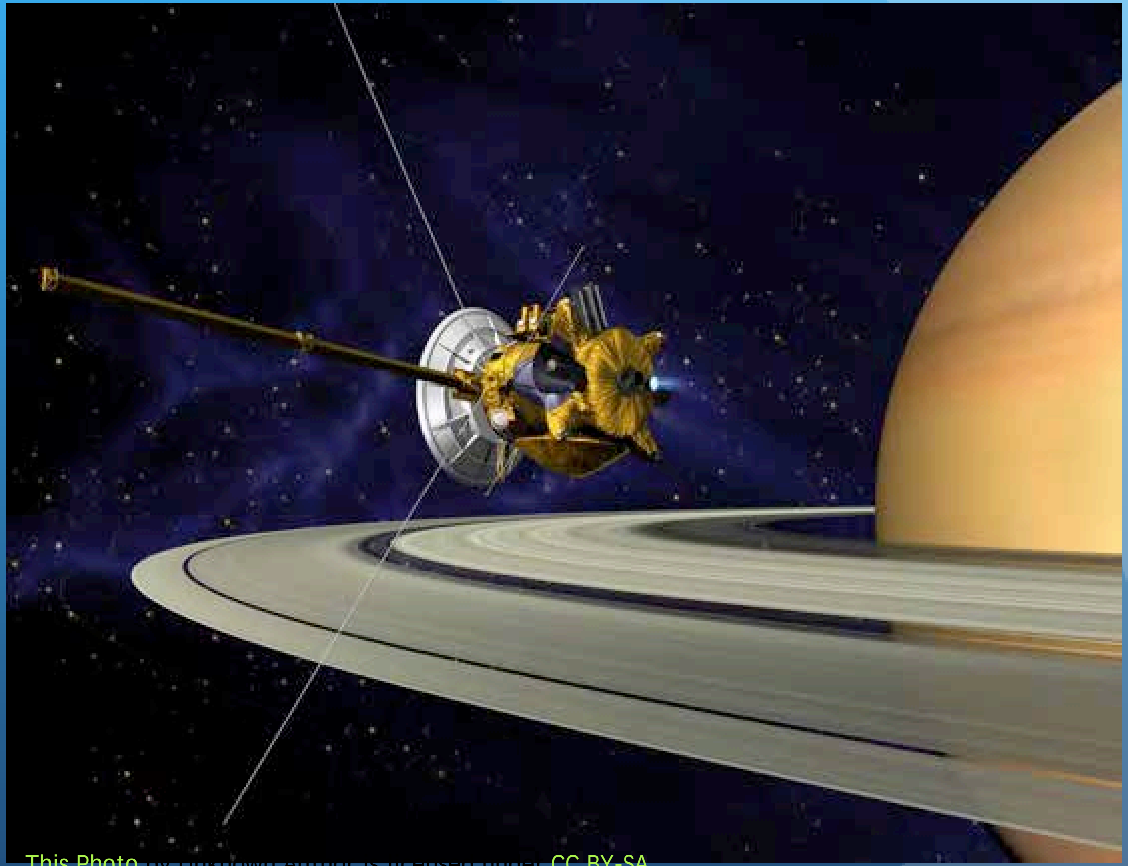
Insight: what it takes to reach Mars

- Launched 5/5/18, rover landed 11/26/18
- 12 thrusters navigators on Earth control using data from:
  - Optical star tracker
  - Inertial measurement unit (IMU)
  - Sun sensors
- Required six Trajectory Correction Maneuvers (TCMs) to reach Mars
- Had to enter Martian atmosphere at 12˚ angle (Apollo re-entry corridor: 5.5˚ to 7.5˚)

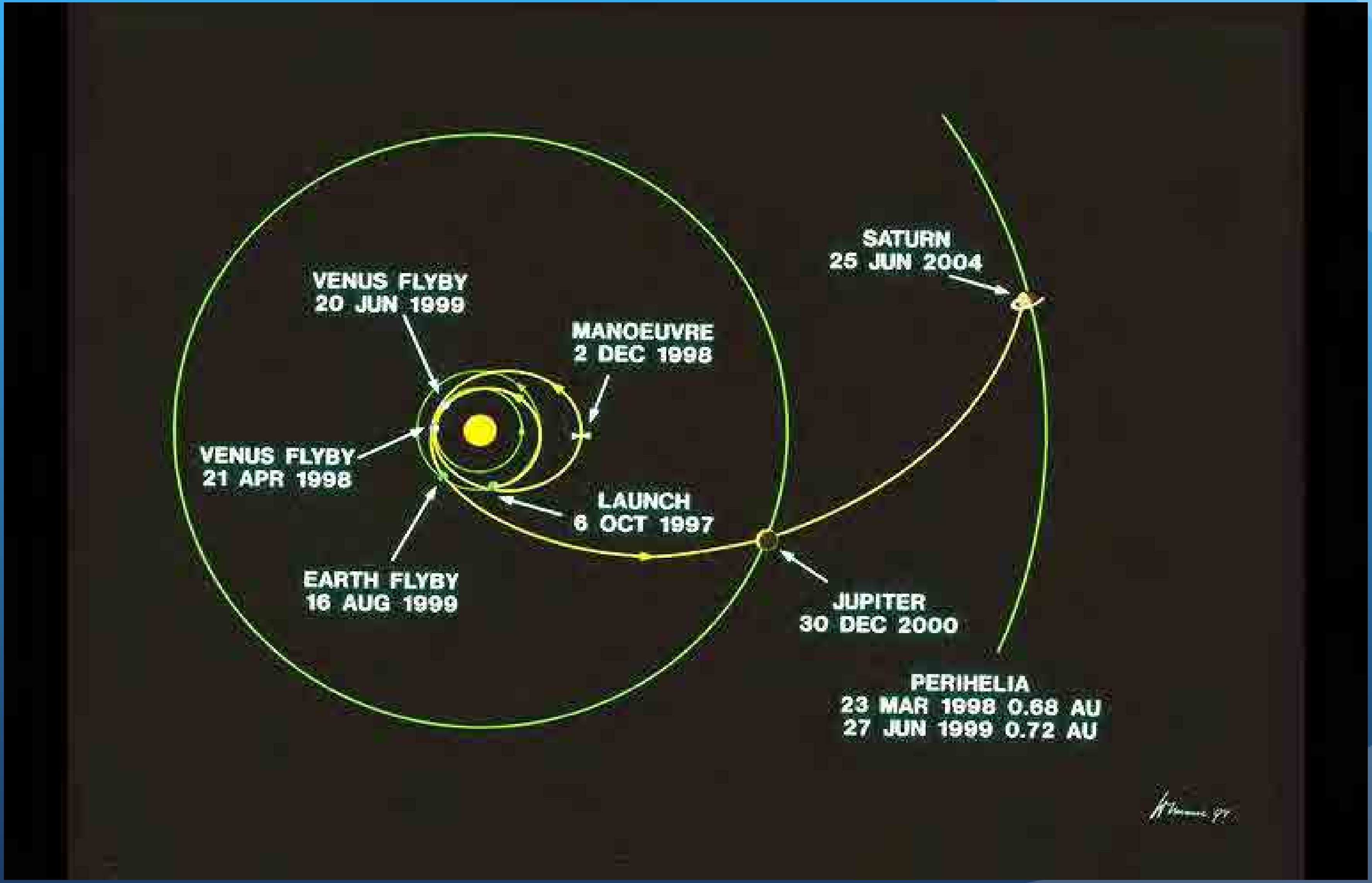


Cassini: Reaching Saturn

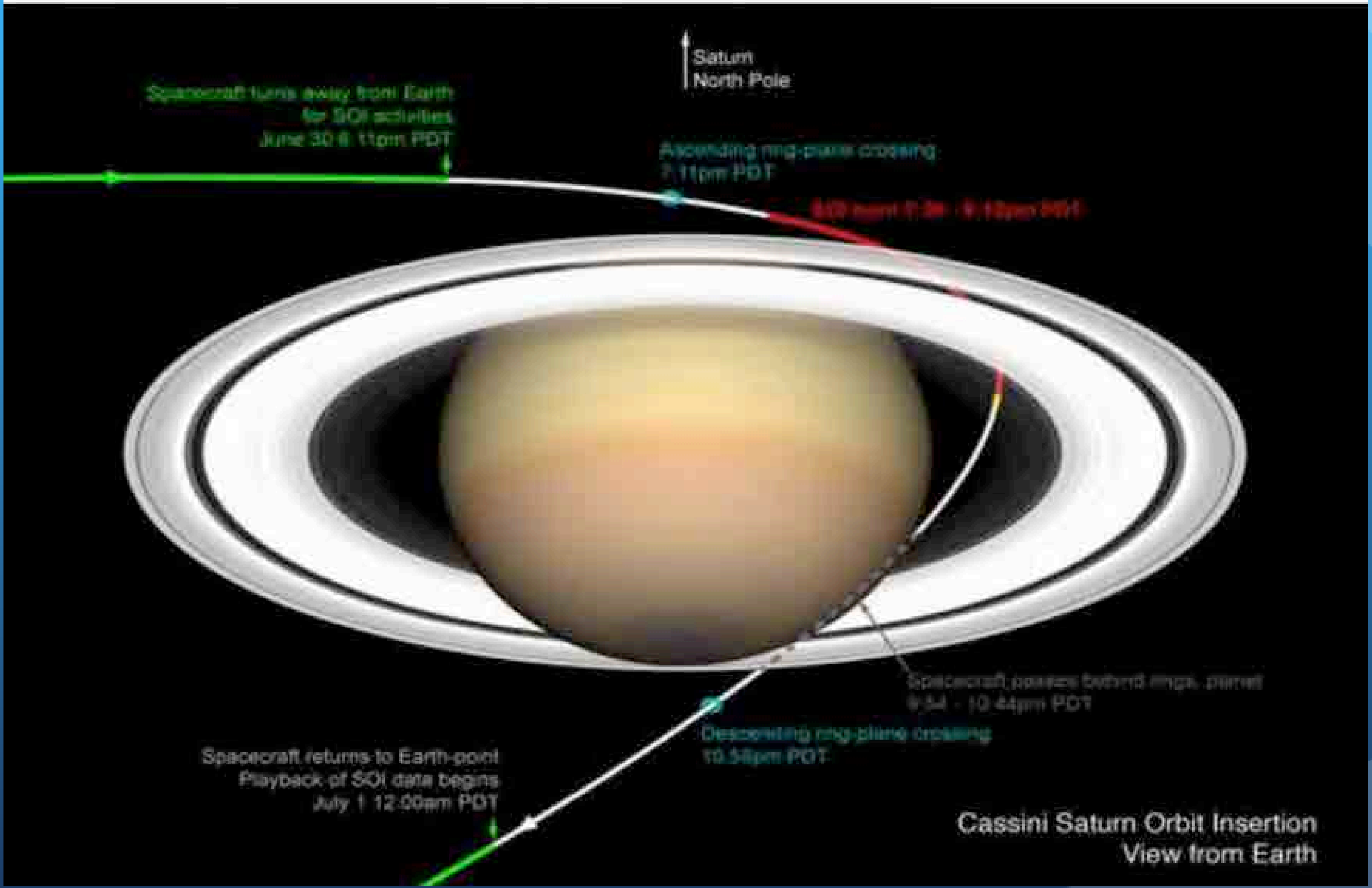
- Launched 1997 arrived at June 2004
- Orbited around Saturn and Titan until 2017
- Sent European Space Agency Huygens probe to surface of Titan



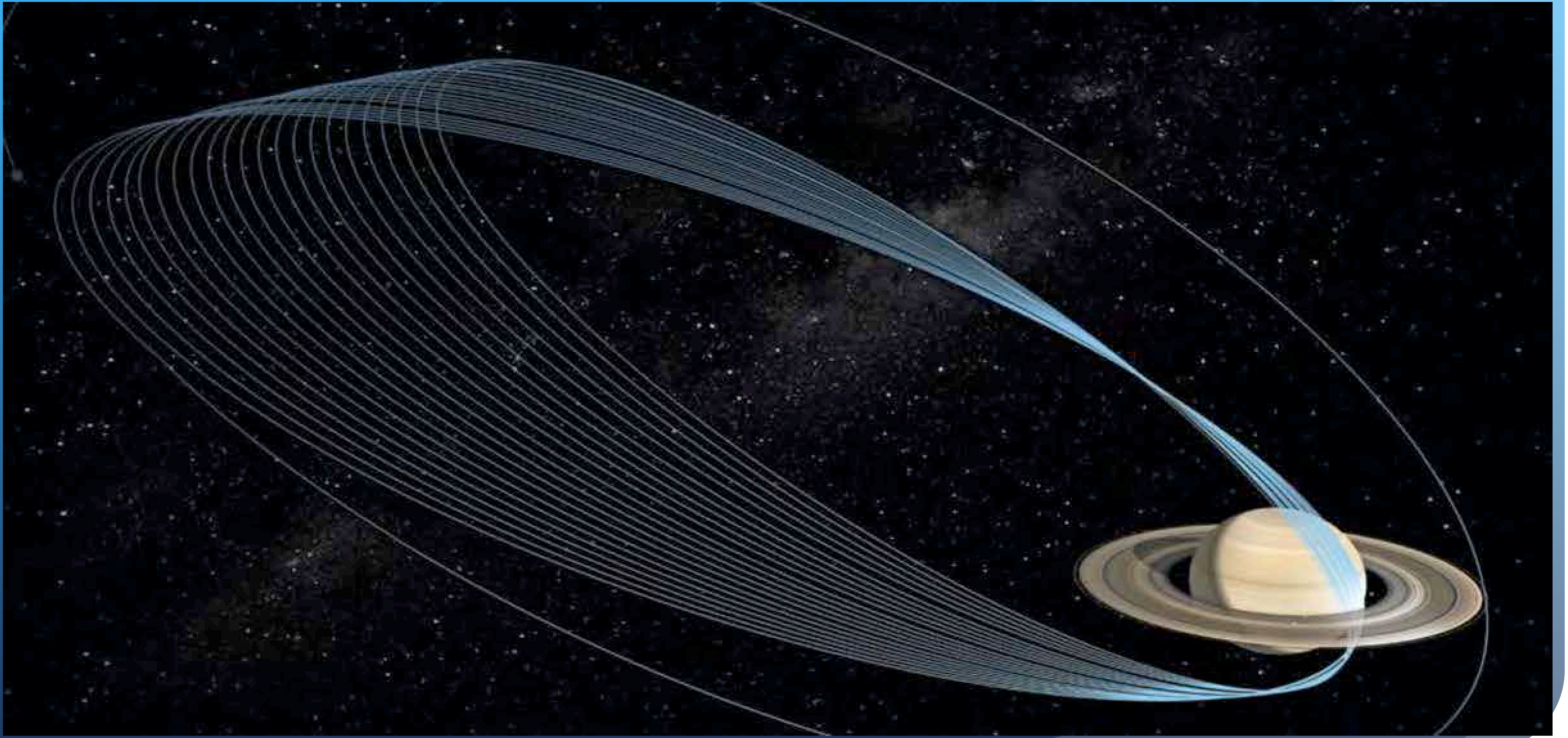
Cassini trajectory to Saturn



June 30, 2004: Saturn orbit insertion (SOI)



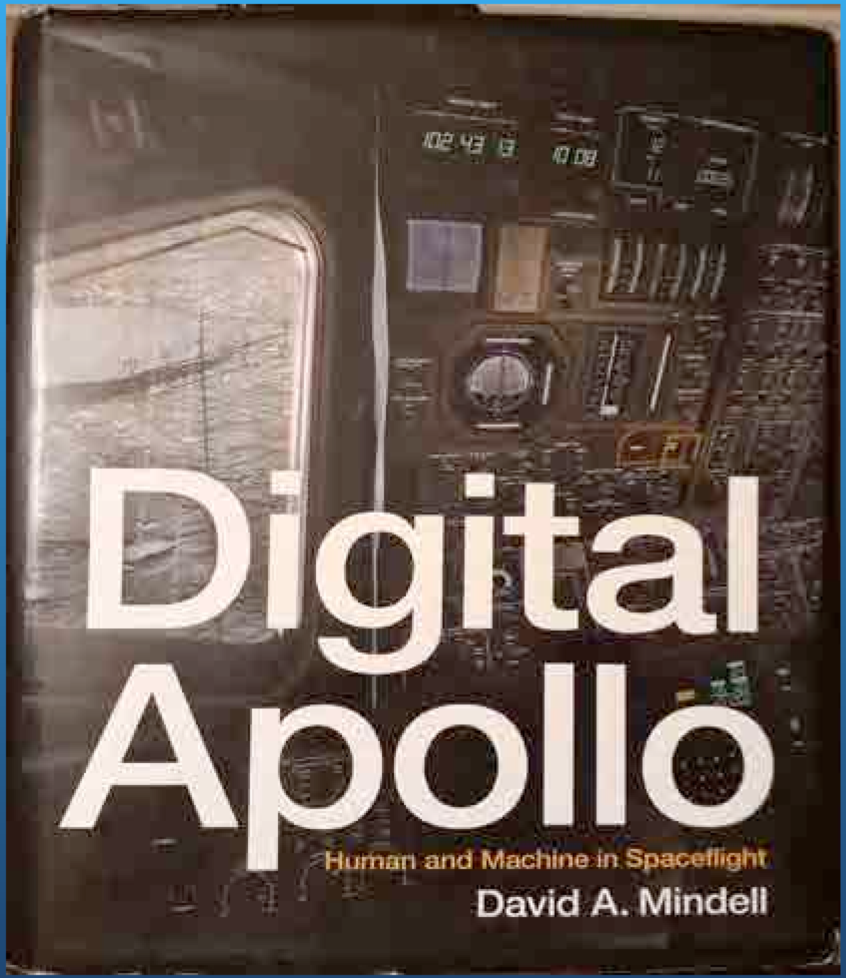
294 orbits of Saturn and Titan 2004-2017



Reaching an asteroid: Osiris Rex

- 7-year mission to survey asteroid Bennu, take sample in 2020, return to Earth
- To survey Bennu:
  - LIDAR: Light detection and ranging
  - Natural Feature Tracking (NFT) – added to supplement LIDAR
  - To reach asteroid: *inertial navigation (IMU)*

Recommendations: book and media



Peter Volante (front, center)

- B.S. Aeronautical Engineering 1958
- Learned to write software, early 1960s
- 1963: started at SDC > MIT-IL in 1966 (right place/time)
- Here: MIT group that supported Apollo 14 landing (*the longest-distance tech support call in history*)

What a great job!(?)

“Guess who’s coming to dinner?” “Daddy!”

Wife and child of Alex Kosmala, AGC Programmer

“Most of us ended up divorced” - Dan Lickly, Director, On-Board Flight Software Development, 1964-1968

Mid-1966:
NASA engineer to MIT software team

- *How can you possibly do this?*
- *Here you sit at the very center of the success or failure of this extremely important program.*
- *You're behind.*
- *Get it through your head you are ****ing this up!*

Coming up with solutions and new ideas was an adventure. Dedication and commitment were a given. Mutual respect was across the board. Because software was a mystery, a black box, upper management gave us total freedom and trust. We had to find a way and we did. Looking back, we were the luckiest people in the world; there was no choice but to be pioneers.

Margaret Hamilton, MIT News, July 2009

Cliché: My [device name] is [number] million times more powerful than the Apollo computer

- Simpsons clip of Meat Loaf Night (Wed.): Lisa is sick of Pork Chop Night (Fri.), wants “anything but pizza, hamburgers, or fried chicken!” Homer suggests Mars. Marge suggests a sushi restaurant.
- Bart heard on the playground that sushi is raw fish.
- Lisa: *As usual, the playground has the facts right, but missed the point entirely.*

Seven Apollo contributors few people have heard of

Phyllis Rye: Software Engineer

- Member of team that wrote AGC code
- Helped write multiple AGC programs, including COLOSSUS code for Command Module
- Later wrote flight software for Space Shuttle

Ramon Alonso: computer scientist

- Born in Argentina, learned English as 2nd language
- Late 1950s: worked on designing computer for Mars probe that never flew
- Apollo: conceived and designed Display Keyboard interface (DSKY)

Jane Goode: Software Engineer and Astronaut Trainer

- Helped write rendezvous targeting and navigation software for Command Module

Member of teams that:

- Provided mission support from MIT
- Trained astronaut crews

Richard Battin: Primary Guidance System Lead

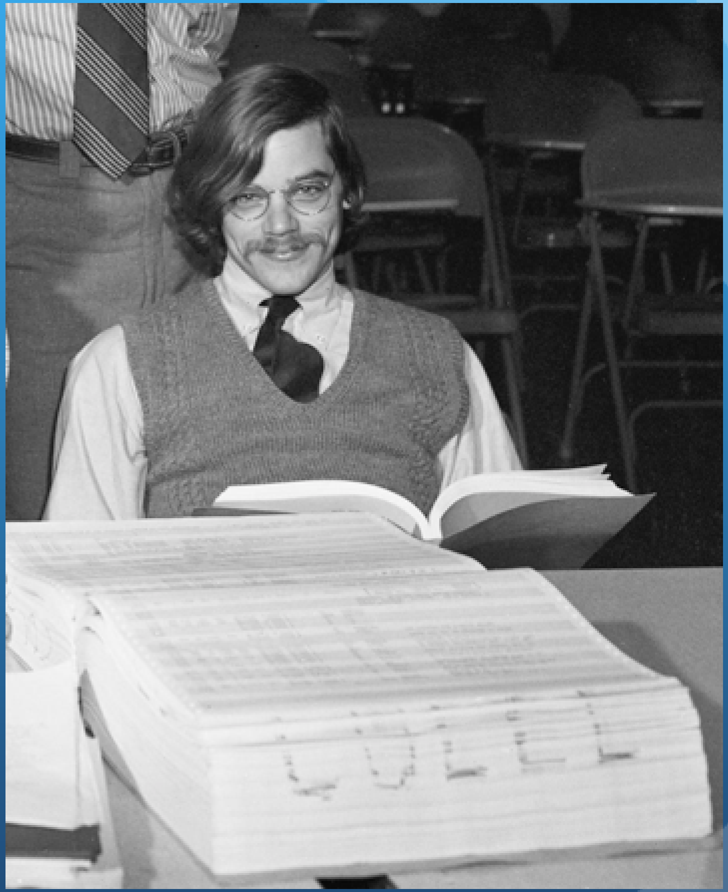
- Taught Astrodynamics at MIT 1960-2010 (students included 8 Apollo astronauts – his TA, Janice Voss, flew on the Space Shuttle 5 times)

From wehackthemoon.com:

- *“Played an integral role in making modern space exploration possible”*
- *“His work resulted in the basis of modern spaceflight guidance algorithms”*

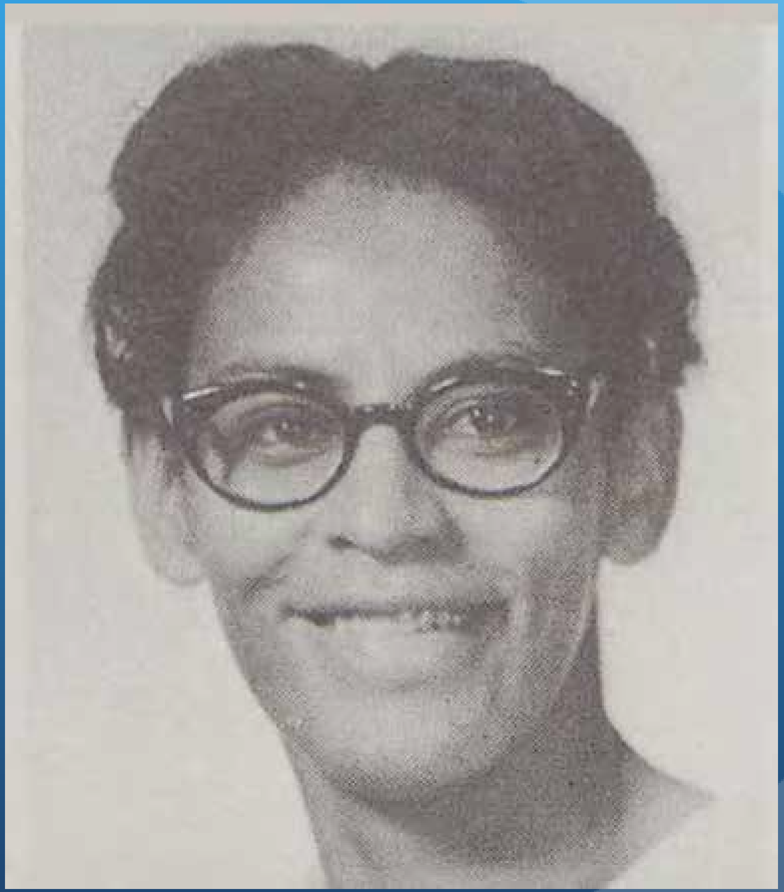
Don Eyles: Software Engineer

- Graduated from Boston Univ. in 1966, had never studied computer science, walked into MIT office, was hired same day
- Helped write AGC code for lunar landing sequence
- Wrote workaround code in 90 minutes when AGC erroneously aborted Apollo 14 lunar landing, saving mission from failure (*“an ingenious solution…an amazing feat…”* Peter Volante)



Elaine Denniston: Data Keypuncher and “Rope Mother”

- Went to MIT as a temp to do data keypunching for software testing
- Soon made herself invaluable checking code developers wrote for missing words, symbols, or punctuation (”Rope Mother”)
- Later went to law school, had a career as an attorney



Hal Laning: Computer Scientist

- Programmed some of the earliest computers in 1950s
- Conceived of asynchronous computer operations based on priority: the computer architecture that made landing on the Moon with 76 KB of memory possible

July 20, 2019