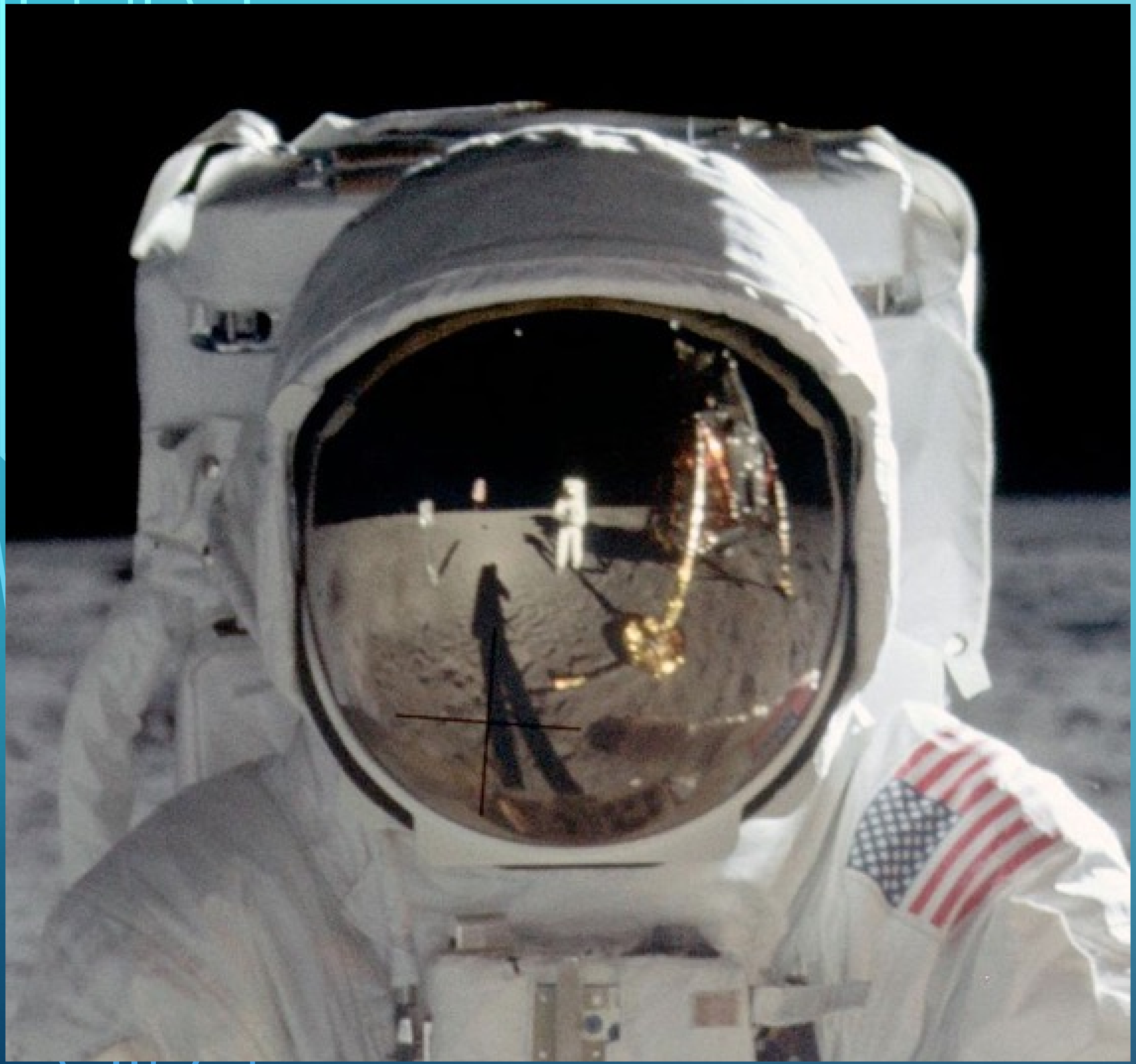


WHEN 100 FLOPS/WATT WAS A GIANT LEAP

THE APOLLO GUIDANCE COMPUTER HARDWARE, SOFTWARE AND APPLICATION IN MOON MISSIONS

Mark C Miller, LLNL (miller86@llnl.gov), July 17 2019

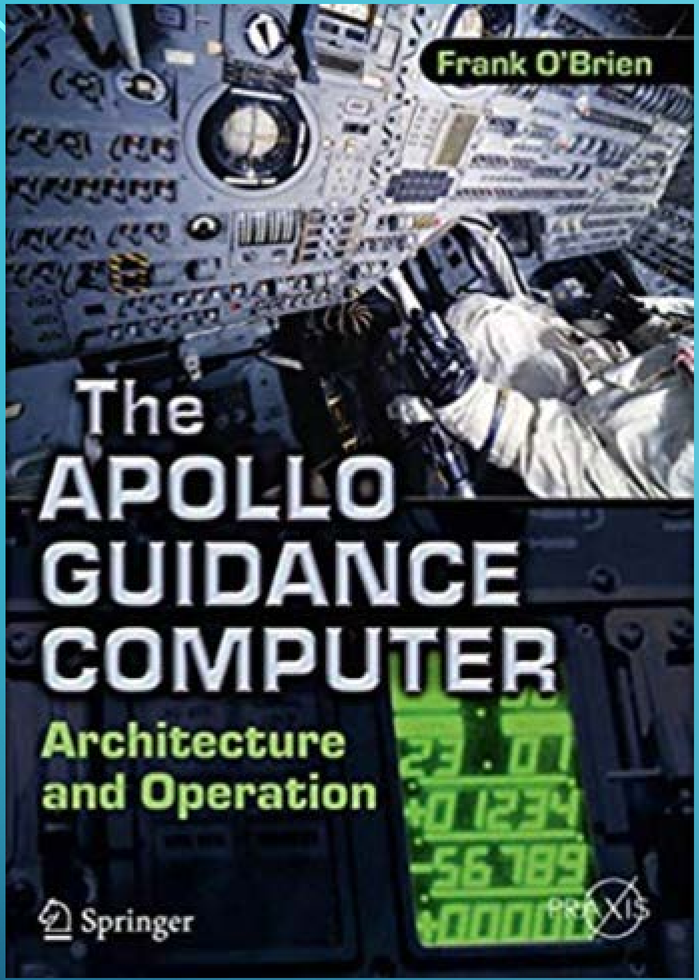
Presented in the webinar series
Best Practices for HPC Software Developers

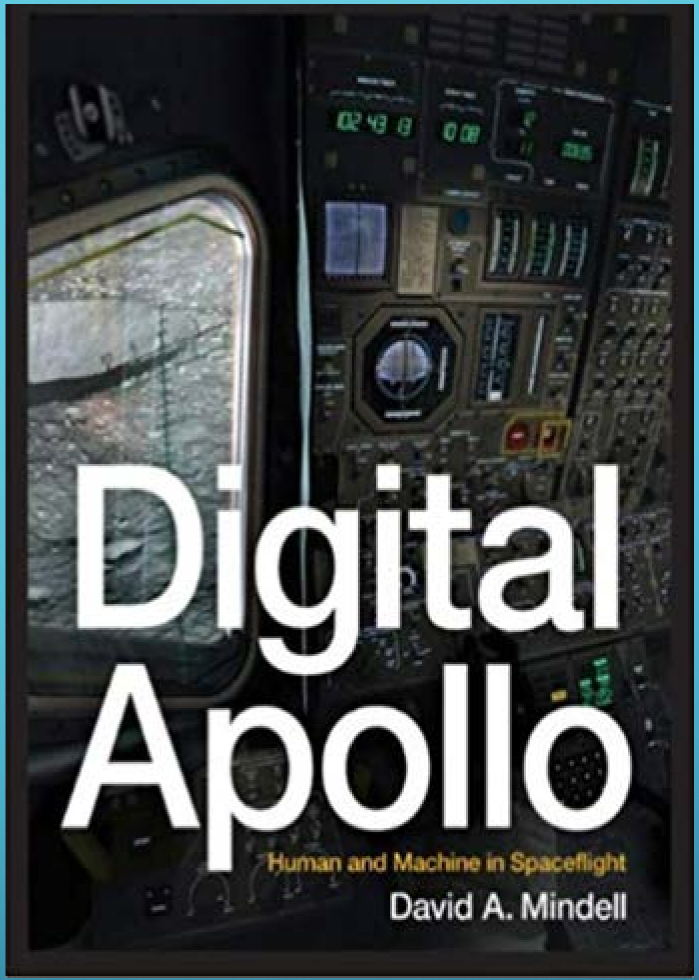
OUTLINE

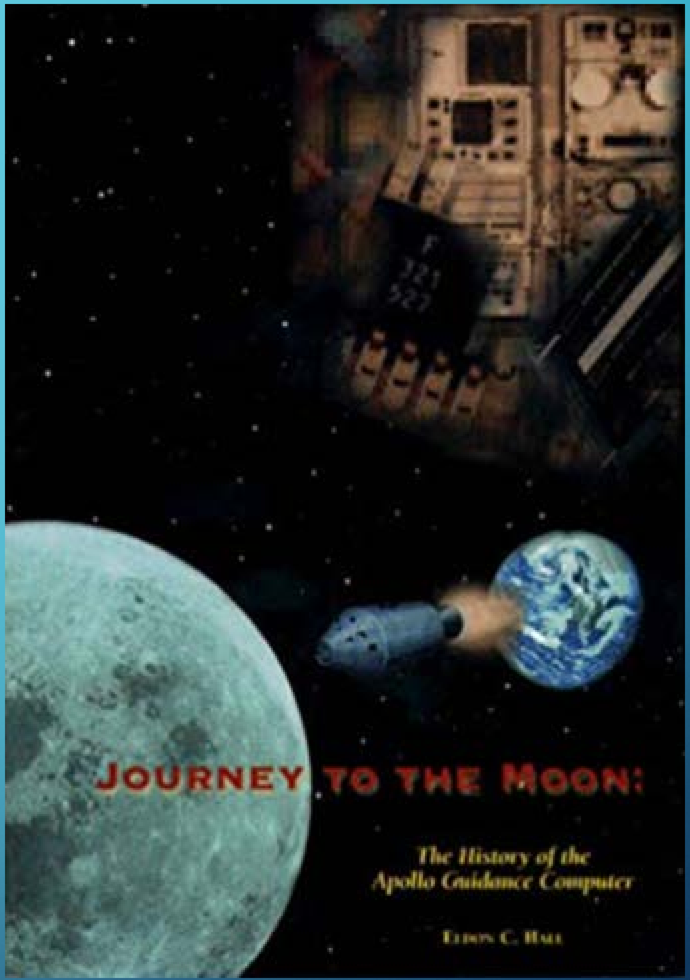
- Background
- Hardware Architecture
- The Software Effort
- Brief Detour
- Mission Applications

CURRENT GENERATION HPC/CSE SOFTWARE DEVELOPERS WILL RECOGNIZE MANY COMMON THEMES

- Flops/Watt power constraints
- Checkpoint/Restart
- Performance Portability
- Co-Design
- Sufficient Testing Resources
- Role and Impact of Software Development Processes







Virtual AGC Project: https://www.ibiblio.org/apollo/

3-Part Blog Series on Better Scientific Software Site (bssw.io)

Part 1 | Part 2 | Part 3

OUTLINE

- Background
- Hardware Architecture
- The Software Effort
- Brief Detour
- Mission Applications

WHAT WAS THE APOLLO PROGRAM?

- 10 year project, starting in 1961 to land people on the moon
  - 36 attempts 1958-1965; none survivable
- 7 Lunar Missions from Jul. 1969 – Dec. 1972
- The Apollo Guidance Computer (AGC) was instrumental in the success

Early Sixties State of the Art Computers

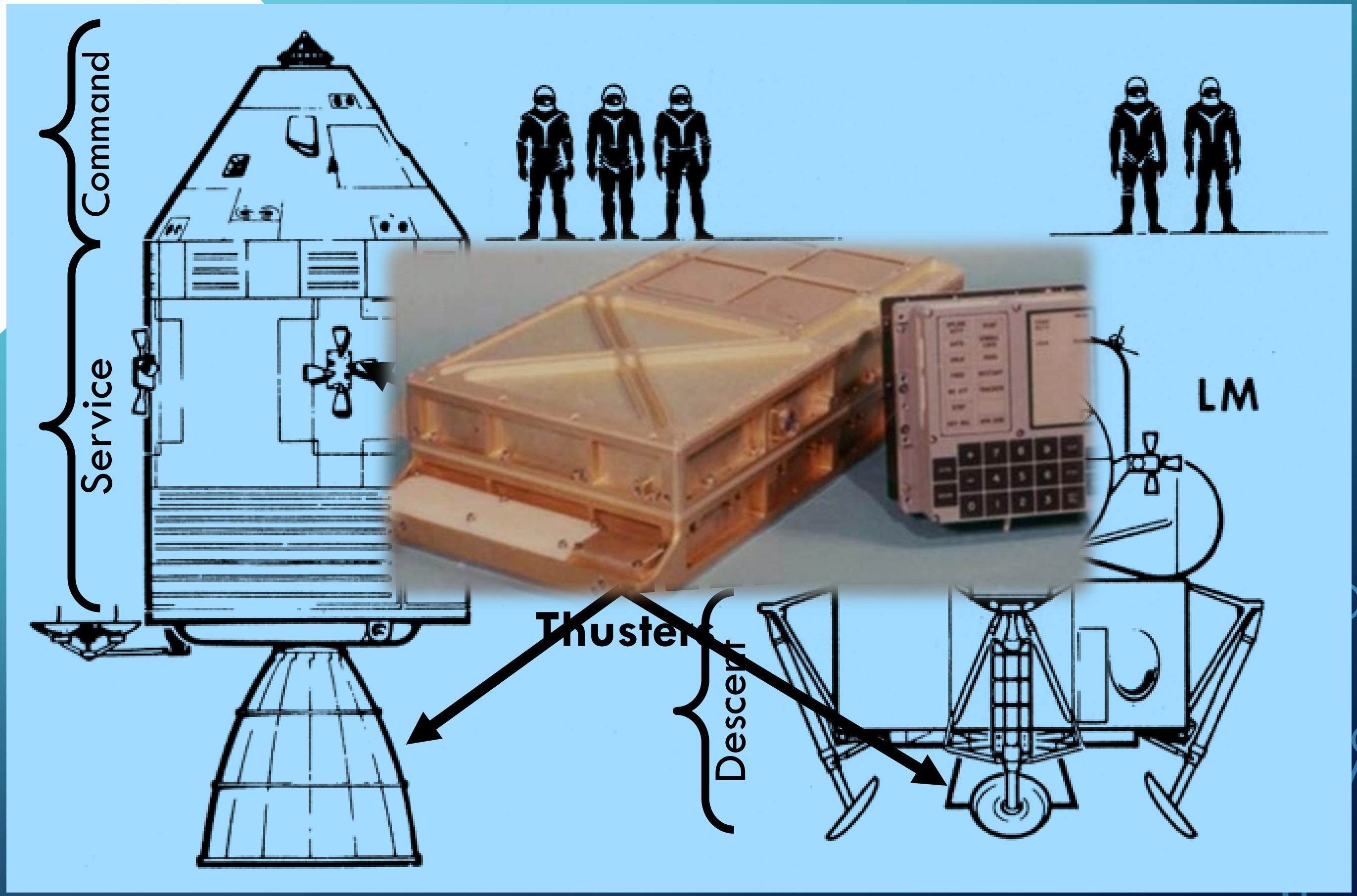
LLNL NARC RTC 2610/64B117

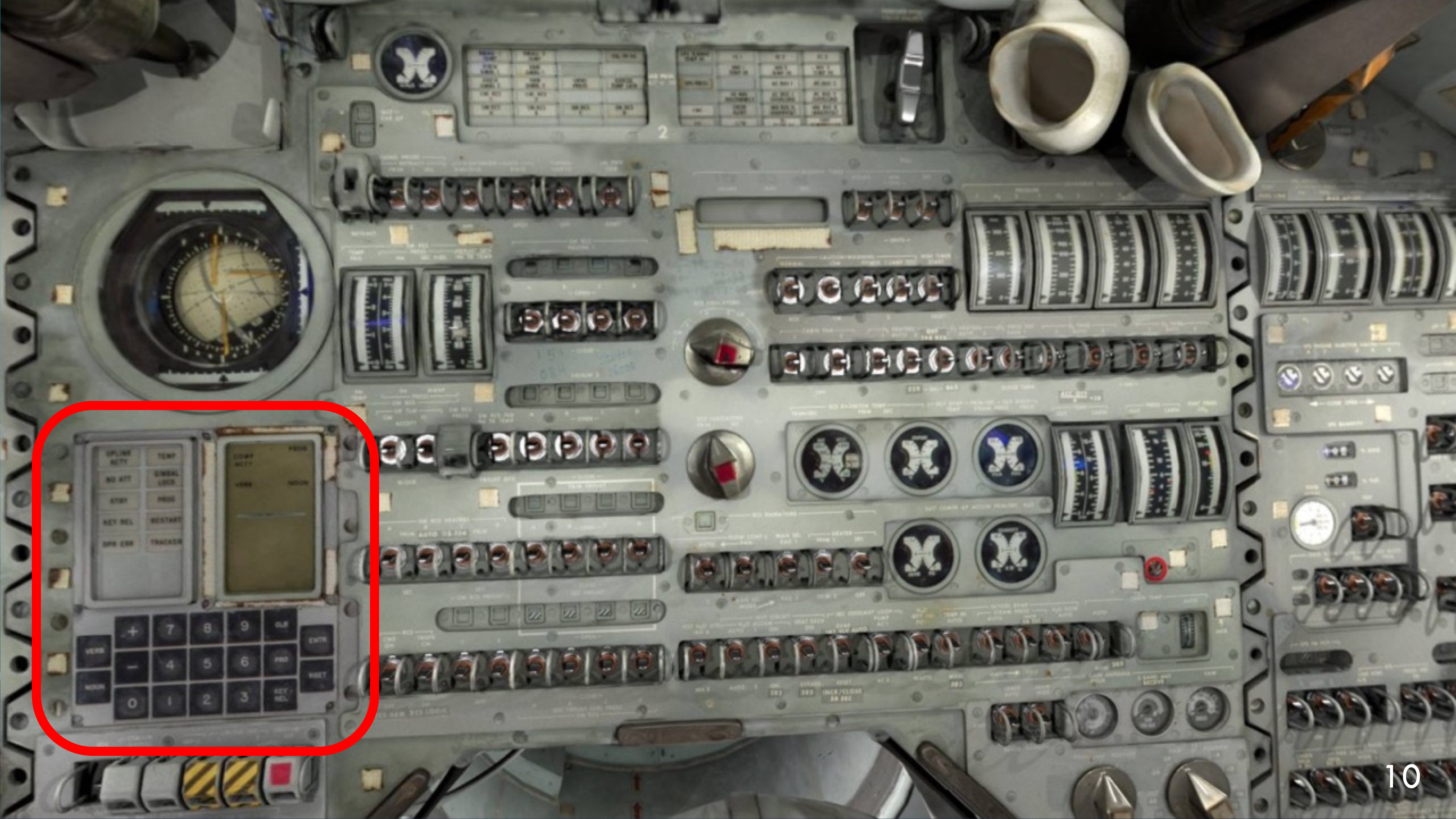
- 4,000 ft^3
- 8 tons
- 125 Kilowatts
- MTBF ≈ Days
- Reboot ≳ 30 mins
- UI = Punch Cards & Printouts
- Time slice multi-tasking
- ~1 Flops/Watt

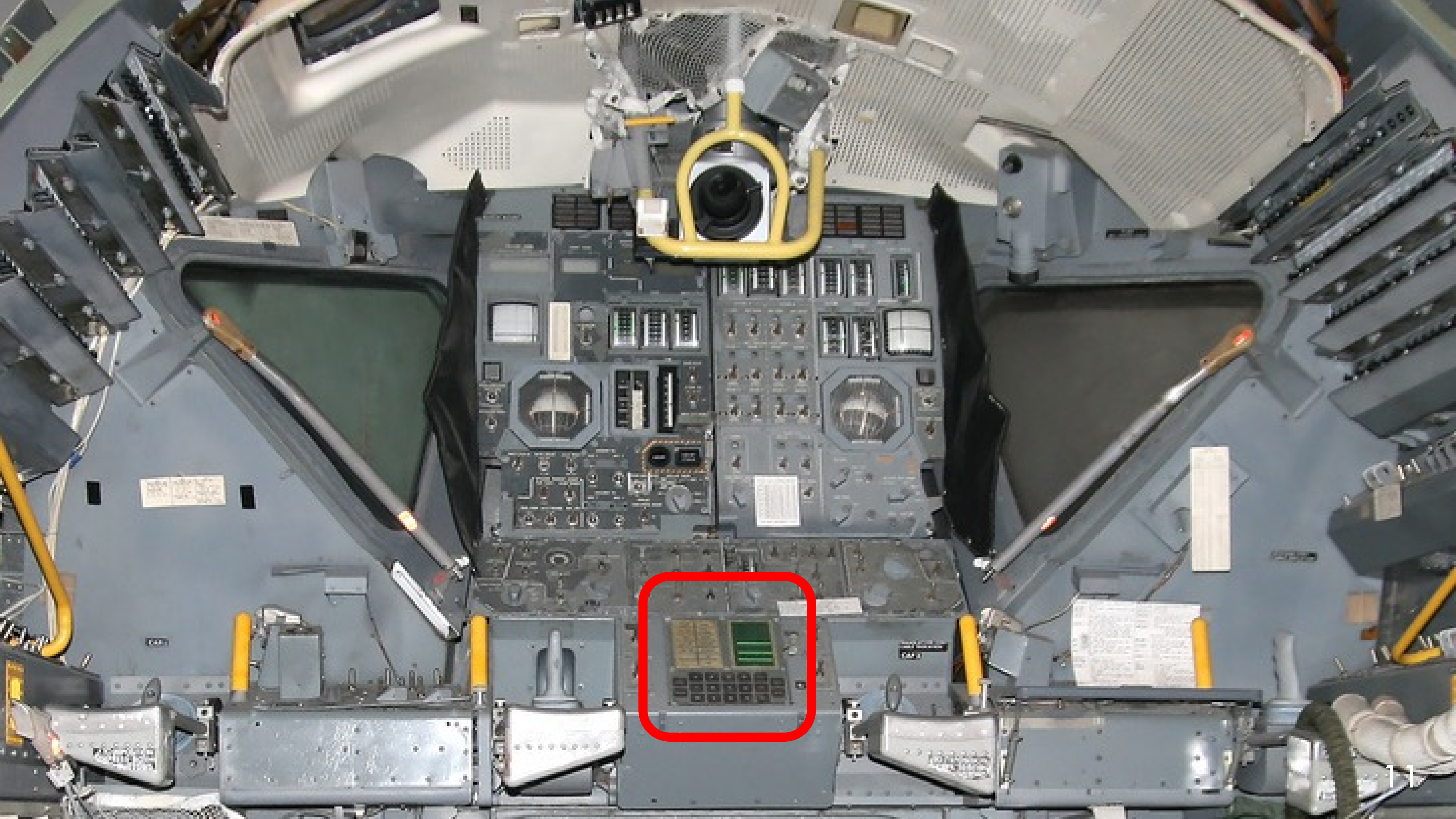
1966 BLOCK II AGC

- 1 cubic foot volume
- 70 lbs weight
- 55 Watts power
- MTBF $\gtrsim$ Months
- Reboot $\approx$ 7 seconds
- UI = Verb/Noun ELD (DSKY)
- Priority Based Multi-Tasking
- ~259 Flops/Watt

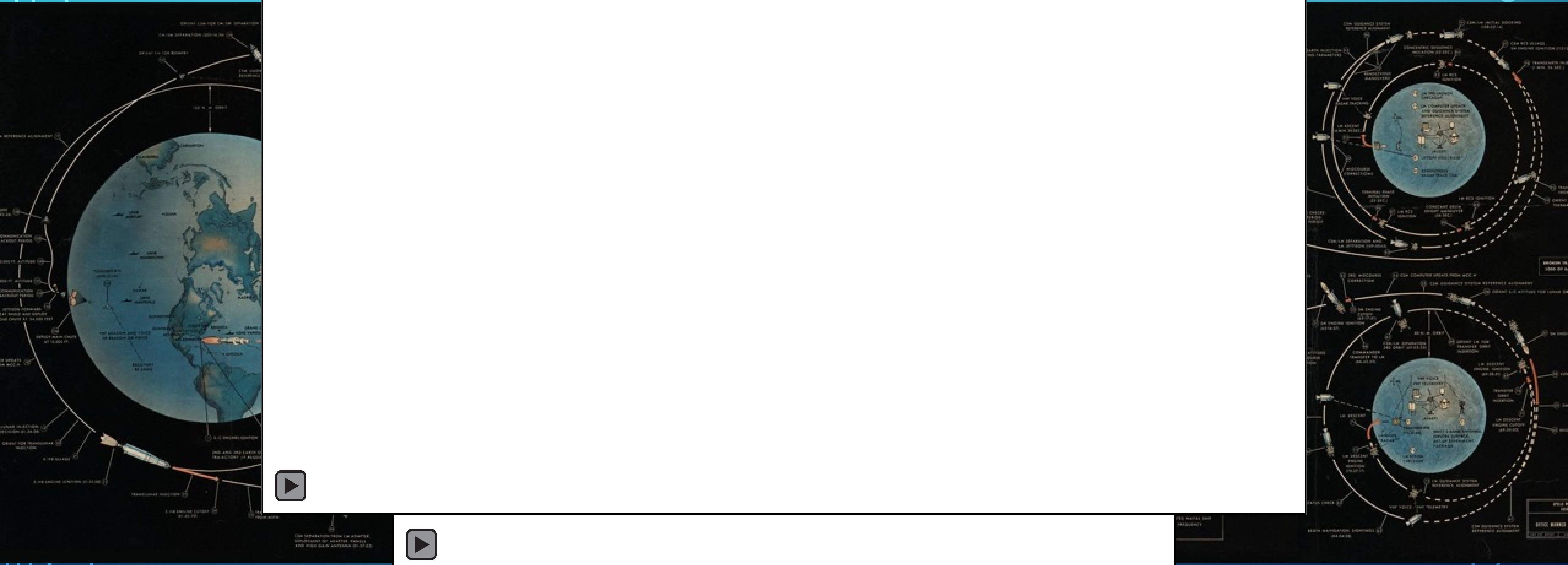
APOLLO SPACECRAFT



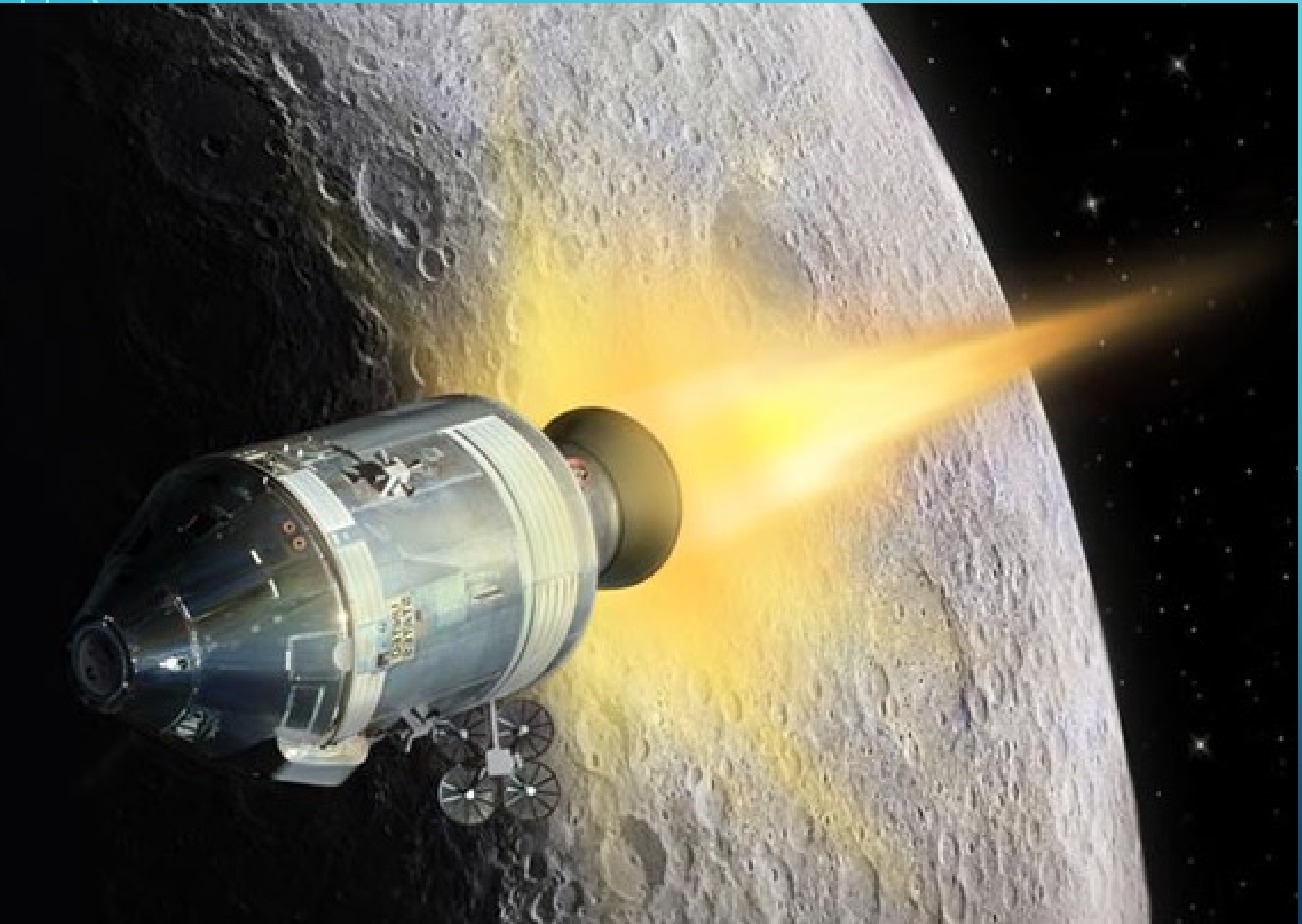

UPLINK ACTY
TEMP
NO ATT
GIMBAL LOCK
STBY
PROG
KEY REL
RESTART
OPR ERR
TRACKER
COMP ACTY
PROG
VERB
NOUN
VERB
NOUN
+
−
0
1
2
3
4
5
6
7
8
9
CLR
PRO
KEY REL
ENTR
RSET



ROLE OF THE COMPUTER



EXAMPLE MANEUVER:
LUNAR ORBIT INSERTION (LOI)



- Velocity = 2 miles/sec
- Distance from moon = 60 miles
- RT signal to Earth = 2.5 sec
- Insertion burn on far side

MIT INSTRUMENTATION LAB
PRIME CONTRACTOR ON APOLLO PGNCS

- Design the entire guidance system
- Draper Labs: Charles Stark “Doc” Draper (Apollo’s Iron Man)
- Designed the Polaris missile guidance system
- Massive r&D effort

OUTLINE

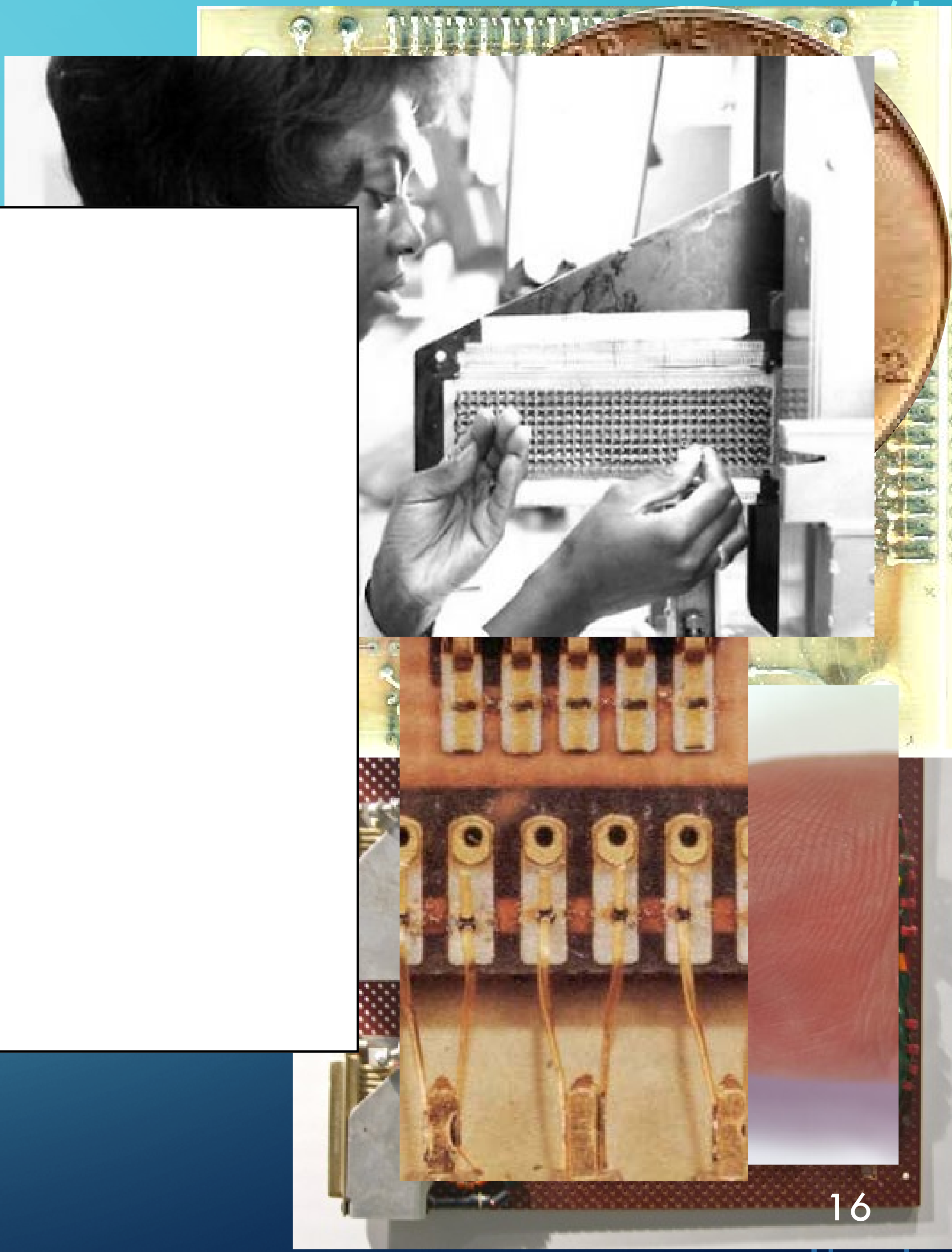
- Background
- Hardware Architecture
- Guidance Software
- Brief Detour
- Mission Applications

AGC HARDWARE OVERVIEW

SPECS

- 16 bit word size (1
- 1.024 MHz Clock
- 12-pulse micro-seq
- 4 central reg's + ~
- 2K words Erasable
- 36K words Fixed M

- **Both RAM and ROM were NVM**



CPU

Fixed / Erasable Memory
Timing, I/O

AGC ARCHITECTURE OVERVIEW

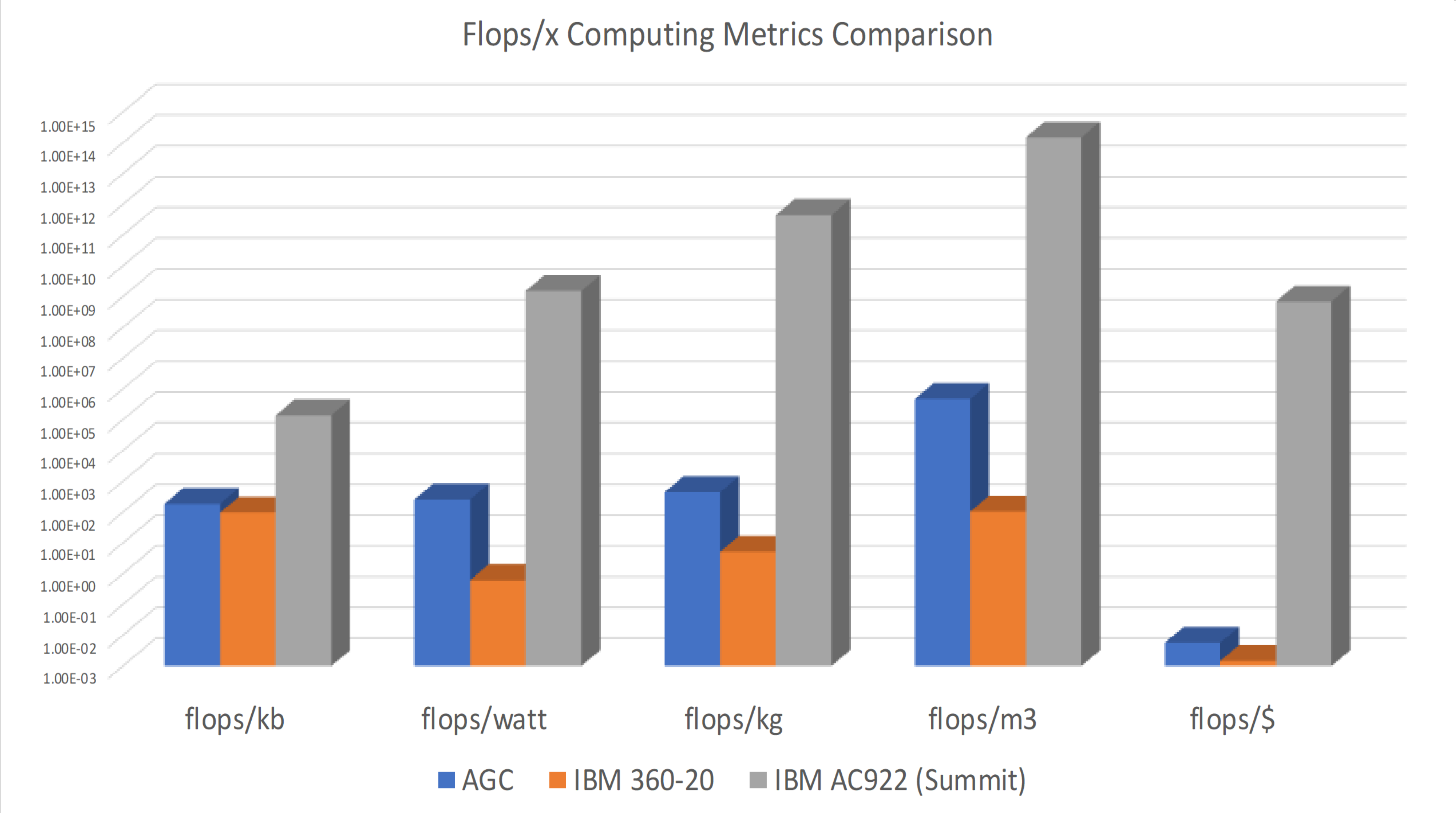
- 4 Central registers
  - A: accumulator w/overflow bit
  - Z: program counter
  - Q: div-remainder / return addr
  - L: lower-product
- Other special purposes registers
  - ROM / RAM memory banking
  - Editing (shift) registers
  - Zero / NEWJOB (00067_8)
  - Not directly programmable
- 8 basic + 33 extended instructions
  - Data Movement
  - Arithmetic & Logic
  - Flow Control
  - I/O & Interrupts
- Many exotic I/O devices
- Programmed in Assembly Language

ACTUAL MACHINE INSTRUCTION SET

- **CS**: clear and subtract
- **TS**: transfer to storage w/ overflow ha
- **XCH**: exchange A w/ storage
- **AD**: add
- **MASK**: bit-wise and
- **TC**: Transfer control
- **CCS**: count, compare & Skip
- **INDEX**: add (+/-) offset to next instru
- **MP**: multiply
- **DV**: divide
- **Others**…

```
LOOPRATE        EXTEND
                INDEX   DAPTEMP6
                MP      NO.PJETS
                CA      L
                INDEX   DAPTEMP6
                TS      DAPTEMP1        # SIGNED TORQUE AT 1 JET-SEC FOR FILTER
                EXTEND
                MP      BIT10           # RESCALE TO 32; ONE BIT ABOUT 2 JET-MSEC
                EXTEND
                BZMF    NEGTORK
STORTORK        INDEX   Q               # INCREMENT DOWNLIST REGISTER.
                ADS     DOWNTORK        #   NOTE: NOT INITIALIZED; OVERFLOWS.

                CCS     DAPTEMP6
                TCF     RATELOOP +1
                TCF     ROTORQUE
SMALLTJU        CA      ZERO
                INDEX   DAPTEMP6
                XCH     TJP
                EXTEND
## Page 1466
                MP      ELEVEN          # 10.24 PLUS
                CA      L
                TCF     LOOPRATE
ROTORQUE        CA      DAPTEMP2
                AD      DAPTEMP3
```


Flops/x Computing Metrics Comparison
1.00E+15
1.00E+14
1.00E+13
1.00E+12
1.00E+11
1.00E+10
1.00E+09
1.00E+08
1.00E+07
1.00E+06
1.00E+05
1.00E+04
1.00E+03
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
flops/kb
flops/watt
flops/kg
flops/m3
flops/$
AGC
IBM 360-20
IBM AC922 (Summit)

THE AGC EXECUTIVE (OPERATING SYSTEM)

"TASKS"

- Short, finely tuned
  - < 5 ms (150-200 instructions)
- Scheduled by time (in the future)
- Some tasks only schedule a "job"
- Waitlist data structure to manage
  - List of tasks sorted by time to run

"JOBS"

- Priority Scheduled
- 12 words of state (4 regs + MPAC)
- 43 words for vector accumulator
  - only for Interpreted jobs
- Jobs adjust their own priority up/down
- New Job checked every 20 ms
  - Two basic instructions (CCS/TC)
  - At end of every interpreted instruction

WAYPOINTS AND RESTART

- Critical routines were restart protected
- Restart phase tables maintained in fixed memory
- Waypoints (phase table pointers) periodically updated in erasable memory

- Consumed 4% of fixed memory, additional coding and testing complexity

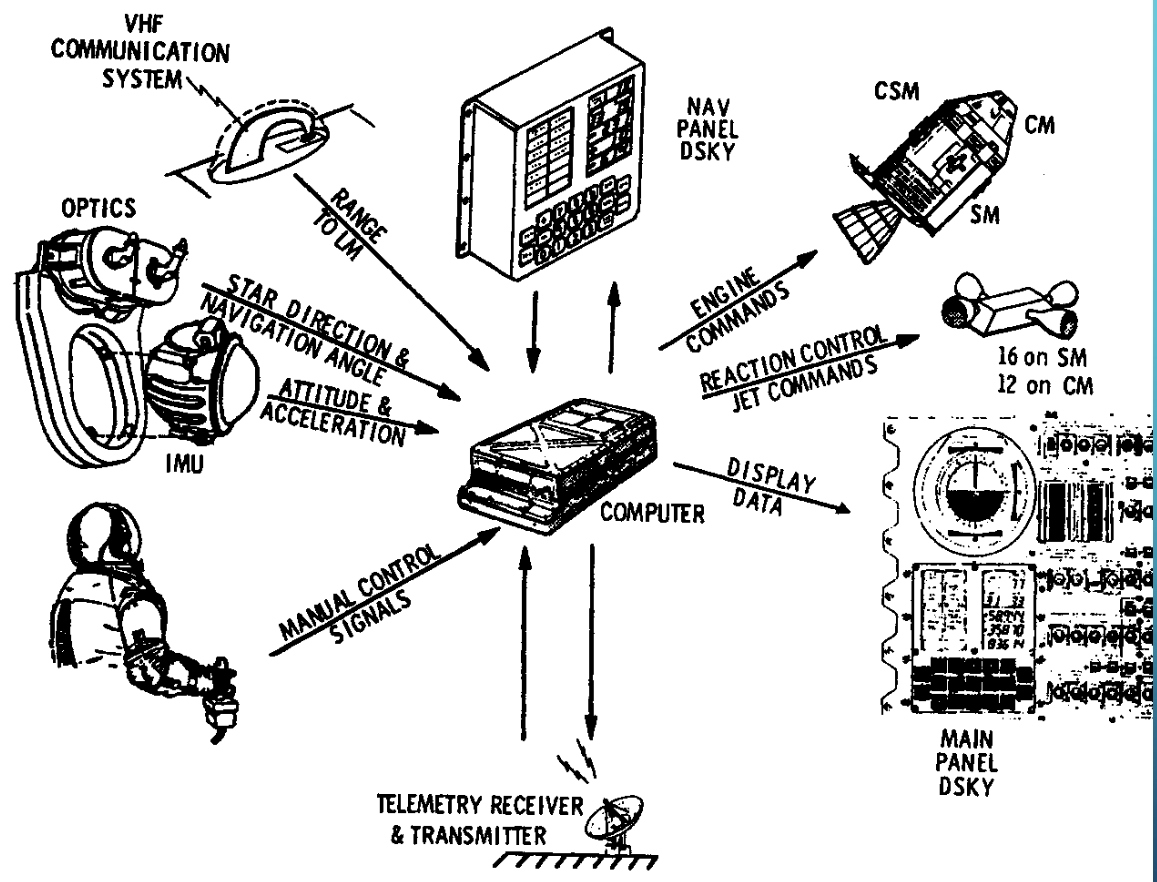
Problem: Compute $\underline{z} = aM(\underline{x} + \underline{y})$
where a is a scalar and M a 3 x 3 matrix

Program (requires 7 words of storage)			Explanation
VXSC	MXV	Operation Codes	1) The first address of an equation is used to load an accumulator; VAD requests a vector load.
	X Y M A	Operand Adresses	2) Each op code results in a subroutine call with the corresponding address left in a standard location.
STORE	Z	Left-over address used to store result	3) After all op codes have been "executed," the remaining address is used to store the result. Since the result of the last operation is a vector, a vector will be stored in Z.

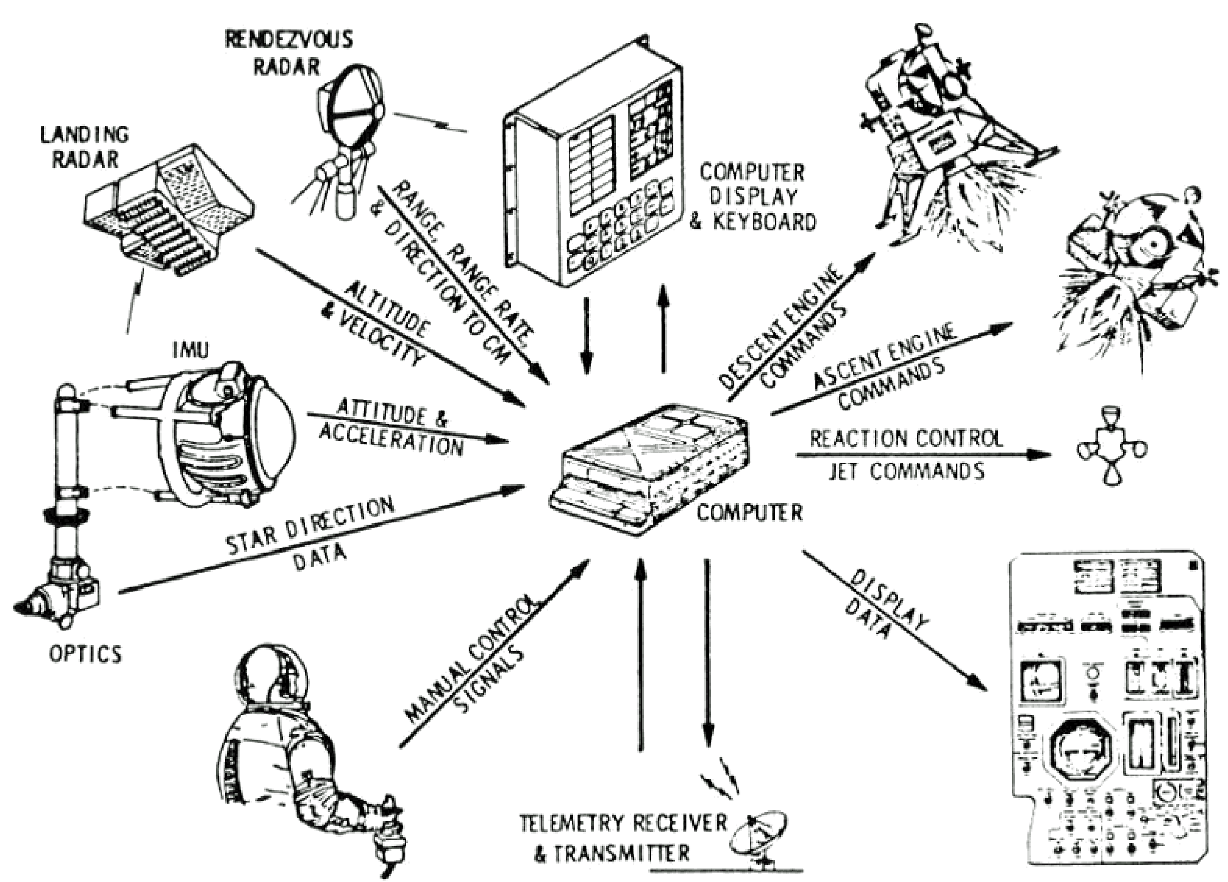
A form of compression to tradeoff memory space for time

I/O DEVICES

Command Module



Lunar Module



I/O PROCESSING (MEMORY MAPPED)

“CHANNELS”

- Very low update rate
- Keystrokes & ELDs on DSKY
- Caution & Warning lights
- RCS Thruster firing
- Switch Statuses
- Managed via interrupt routines

“COUNTERS”

- Pulses from fine-grained state devices
  - IMU gimbals
  - Main engine gimbals
  - Optics & Radar gimbals
- PINC/MINC “instructions”
- Not managed by software
- Cycle stealing

FAULT TOLERANT COMPUTING WAS CRITICAL

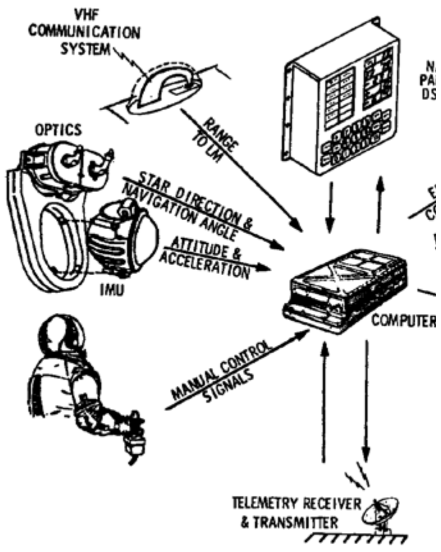
- Hardware level power checks
- Parity check every memory ref
- NEWJOB word night watchman
- Program Alarms (e.g. radar turned off)
- P00DOO (program aborts)
- System Restarts (< 7 seconds)
  - Key data downlinked to Huston

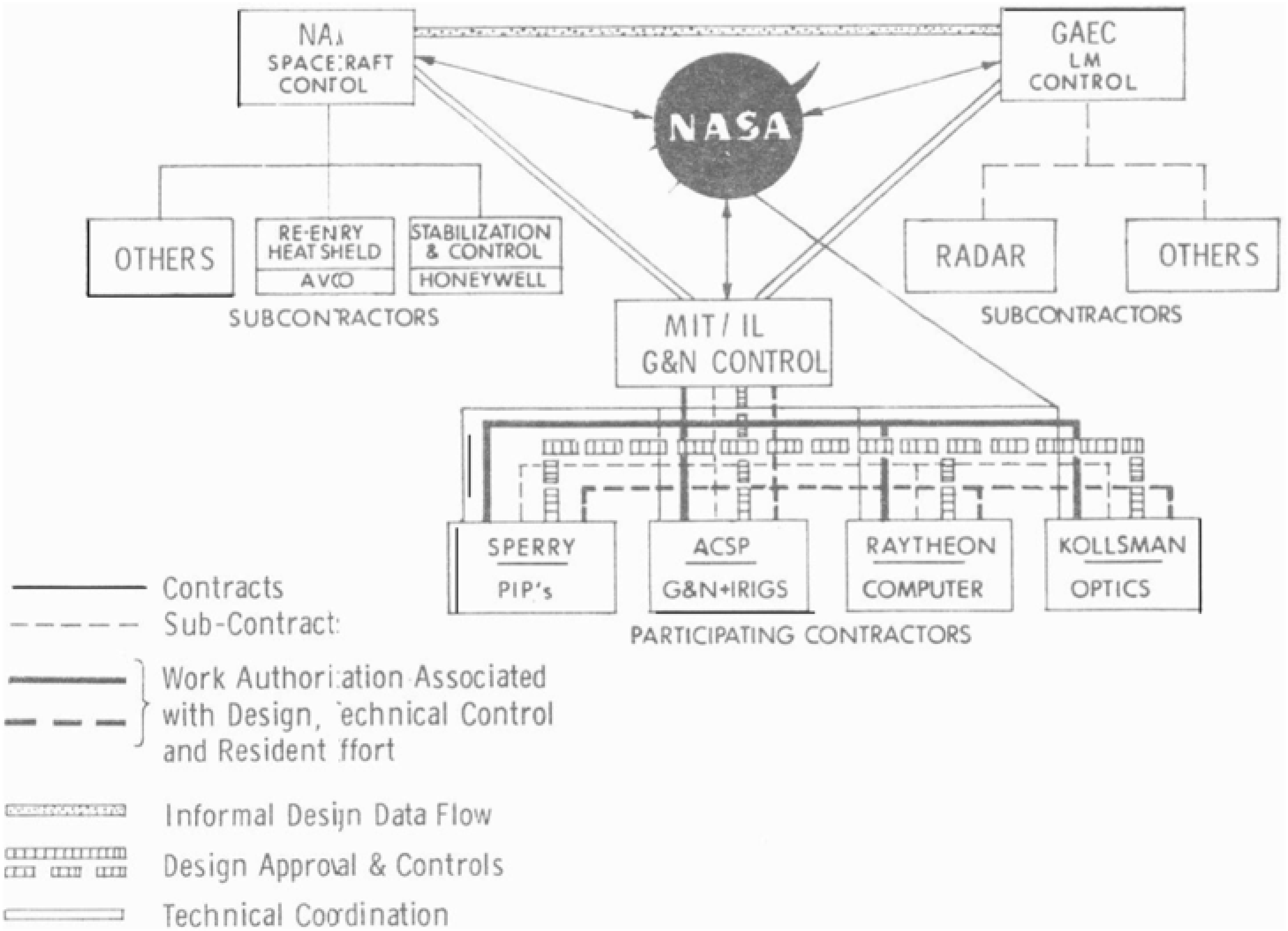
- Extreme Reliability Achieved
  - 42 Units
  - 11,000 hours of vibration and heat/cold
  - 32,000 hours normal operation
  - Only 4 faults observed
  - MTBF → 40,000 hours

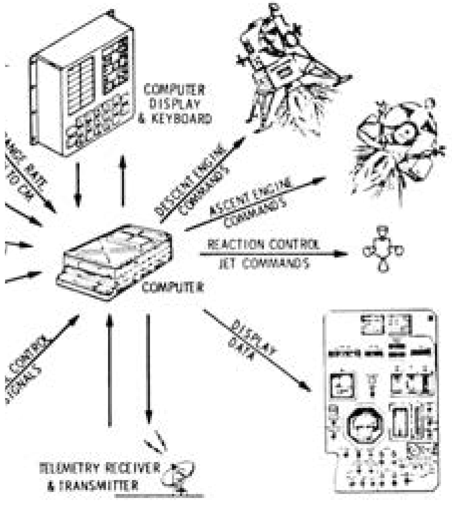
OUTLINE

- Background
- Hardware Architecture
- Guidance Software
- Brief Detour
- Mission Applications

EXTREME CO-DESIGN







Lunar Module Computer Interfaces

- None of
  were kn
- Everythi

WESTERN UNION
TELEGRAM

W. P. MARSHALL, PRESIDENT

435P EDT AUG 9 61 BB257 PB375

W NFA084 GOVT PD NF WASHINGTON DC 9 405P EDT

DR STARK DRAPER, DIR

INSTRUMENTAL LABORATORY MASSACHUSETTS INST OF TECHNOLOGY

CAMBRIDGE MASS

PLEASED TO ADVISE THAT THE NATIONAL AERONAUTICS AND SPACE ADMINISTRATION TODAY ANNOUNCED THAT MIT'S INSTRUMENTATION LABORATORY HAS BEEN SELECTED TO DEVELOP THE GIDANCE NAVIGATION SYSTEM OF THE PROJECT APOLLO SPACECRAFT. APOLLO IS CAPABLE OF CARRYING THREE MEN TO THE MOON AND BACK. MIT IS THE FIRST MEMBER OF THE APOLLO TEAM TO BE CHOSEN. BIDS ARE NOW UNDERWAY FOR THE PRIME CONTRACTOR'S JOB. IN ADDITION TO APOLLO THE INSTRUMENTATION LABORATORY WILL ALSO DEVELOP THE GROUND SUPPORT AND CHECKOUT EQUIPMENT. CONTRACT COVERING THE FIRST YEAR IS AN ESTIMATED $4 MILLION

LEVERETT SALTONSTALL UNITED STATES SENATOR.

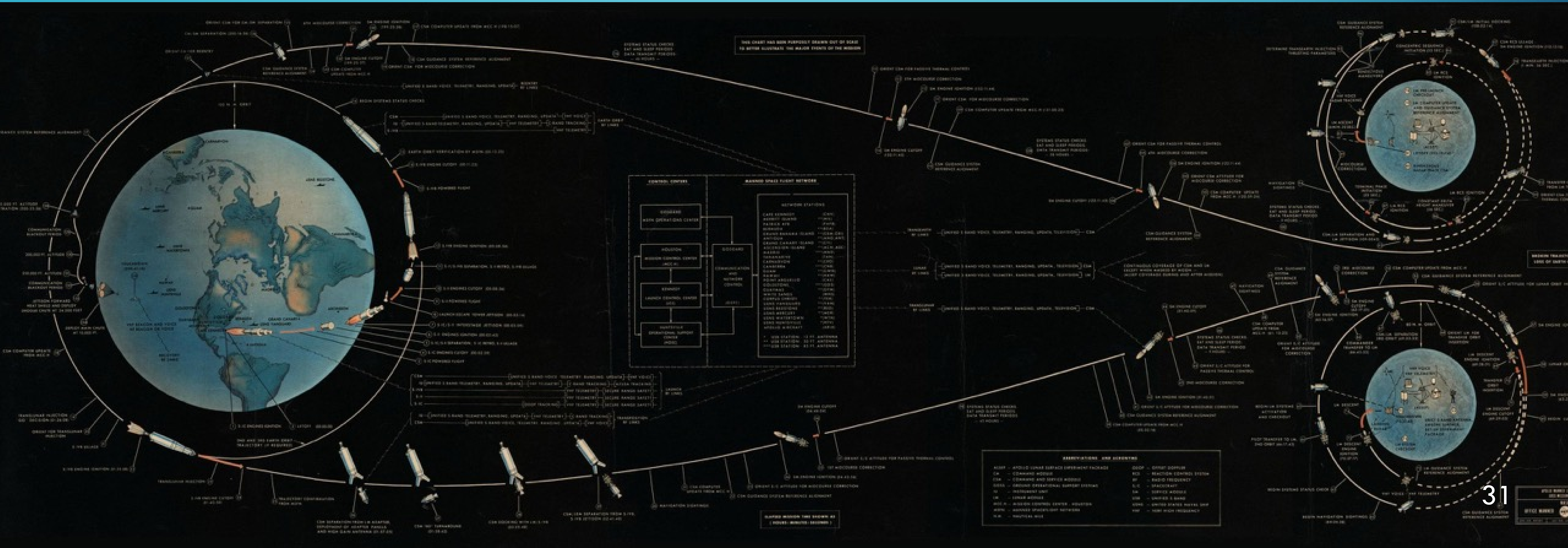
THE ESSENTIAL STEP
MIT SOFTWARE ENGINEERS NEEDED TO PERFORM

- Assemble a “flight program” & release it to Raytheon for rope core weaving
  - 2 months to weave the ropes; 2 months to install, test, run crew rehearsals, etc.
  - Lead engineer for an assembled flight program was called a “rope mother”

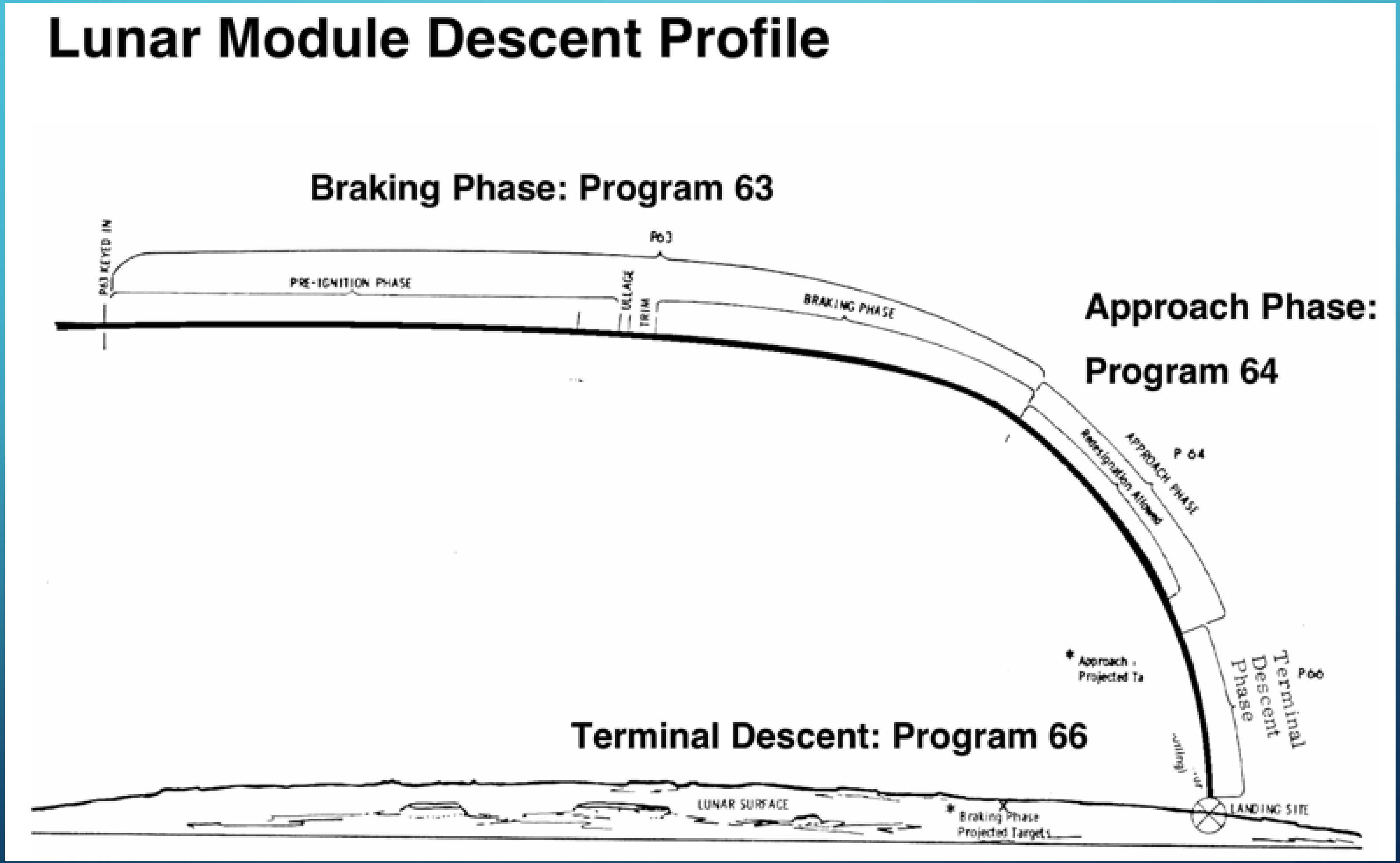
- For ~30 flights (uncrewed and crewed) each with unique guidance requirements

THE AGC HAD AN "APP" FOR THAT

- A lunar mission involved ~11 "burns" of the main engines
- For each unique maneuver, there was a *major mode program* to handle it



LUNAR LANDING MAJOR MODES



EXAMPLE OF GUIDANCE ROUTINE SOFTWARE DEVELOPMENT WORKFLOW – EPHEMERIS ROUTINES

- Knowing the position of the moon at any moment
  - Accurately (within a fraction of a mile)
  - Over a sufficiently long time period (2 weeks)
  - Minimizing time and space resource usage

- Where do you get the “ground truth” data to test validity?
  - Classically studied problem (Newton, Euler, Lagrange, Laplace, Delaunay…)
  - Observational data from astronomers (approx. distance in Earth radii)
  - Brown’s Lunar Theory (1897) and Tables of Motion of the Moon (1919)
  - Data from main-frame codes using a 1,600 term Fourier series approximation

POSSIBLE APPROACHES TO EPHEMERIS FOR AGC

- Store tabulated data and interpolate → too much memory
- Use truncated Fourier series → not accurate enough
- Solve 2-body problem (Earth – Moon system) → not accurate enough
  - 3-body problem (Earth, Moon and Sun) is likely accurate enough → not enough compute
-

  Polynomial fit to X, Y, Z positional data → Accurate and memory efficient

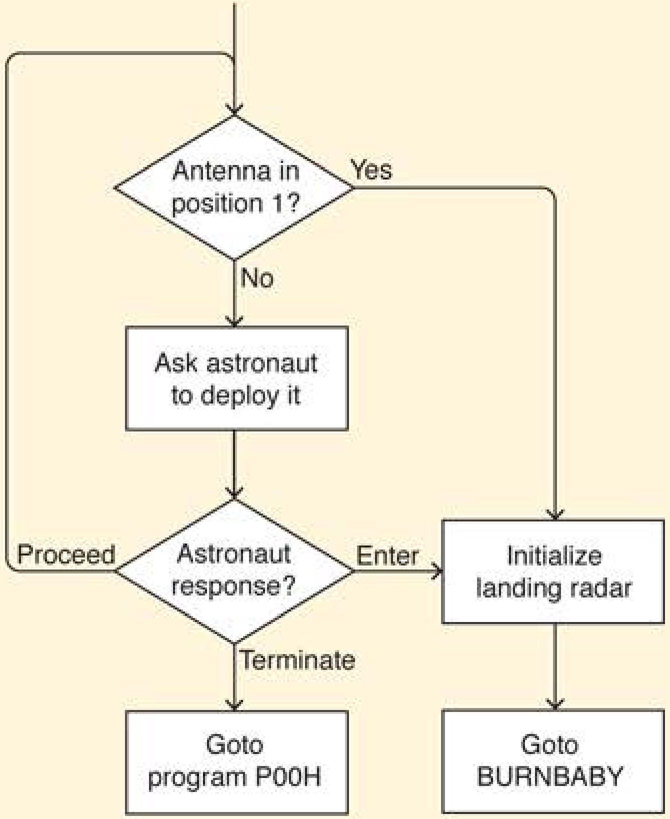
POLYNOMIAL FIT OF EPHEMERIS DATA

- Accuracy:
  - Position to ~1 mile and velocity ~0.5 mph
  - Over a 2-week long period
- 8 double precision coefficients for each of X, Y and Z → **48 words**
  - Did this go into fixed or erasable?
  - Raytheon manufactured contingency ropes for delays in launch
- How implemented…
  - Initially on Honeywell 1800 using MAC language
  - Accuracy, performance and coding confirmed
  - Re-coded in AGC Interpreter Language → **86 words**
  - Tested on AGC all-digital simulator, then test-lab AGC unit
- Became a part of all assembled flight "ropes"

A SNIPIT OF AGC SOURCE CODE

Reading an AGC Program

line	label	opcode	address	comments
0184	P63SPOT3	CA	BIT6	IS THE LR ANTENNA IN POSITION 1 YET
0185		EXTEND		
0186		RAND	CHAN33	
0187		EXTEND		
0188		BZF	P63SPOT4	BRANCH IF ANTENNA ALREADY IN POSITION 1
0189		CAF	CODE500	ASTRONAUT: PLEASE CRANK THE
0190		TC	BANKCALL	SILLY THING AROUND
0191		CADR	GOPERF1	
0192		TCF	GOTOP00H	TERMINATE
0193		TCF	P63SPOT3	PROCEED SEE IF HE'S LYING
0194	P63SPOT4	TC	BANKCALL	ENTER INITIALIZE LANDING RADAR
0195		CADR	SETPOS1	
0196		TC	POSTJUMP	OFF TO SEE THE WIZARD...
0197		CADR	BURNBABY	

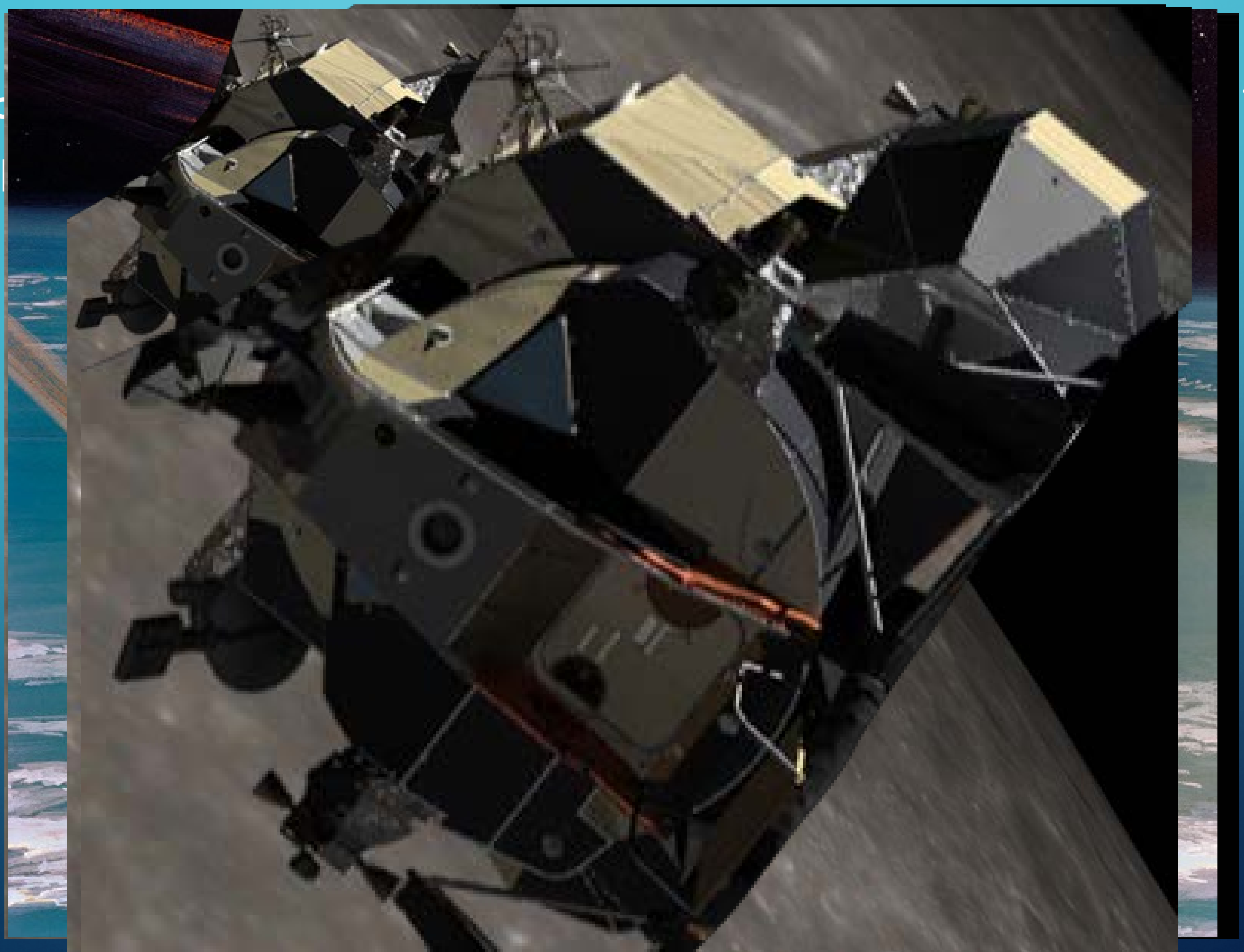


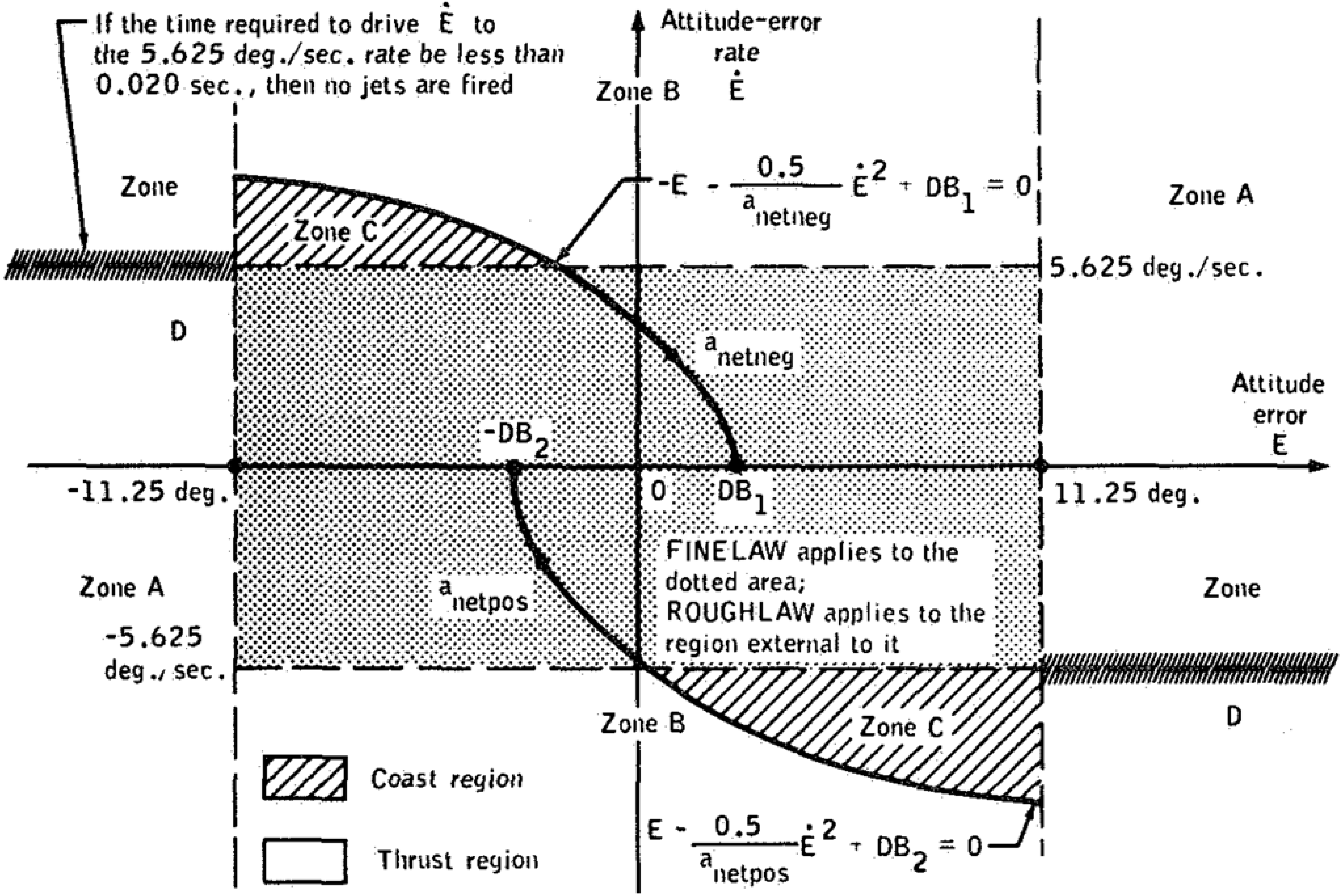
INFRASTRUCTURE SOFTWARE

Program Name	Purpose	Size (AGC words)
Executive[25]	Priority-driven large/long-running process manager	~350
Waitlist[26]	Time-sequenced small/short-running process manager	~300
Down-Telemetry[29]	Transmit system data to ground	~200
Restart[30,31,32]	Error recovery and restart protection	~1225
Interpreter[27]	Space guidance domain-specific programming language interpreter	~2200
DSKY I/O[28]	Cockpit displays and keypad	~3500
Combined Total	22% of fixed memory	~7775

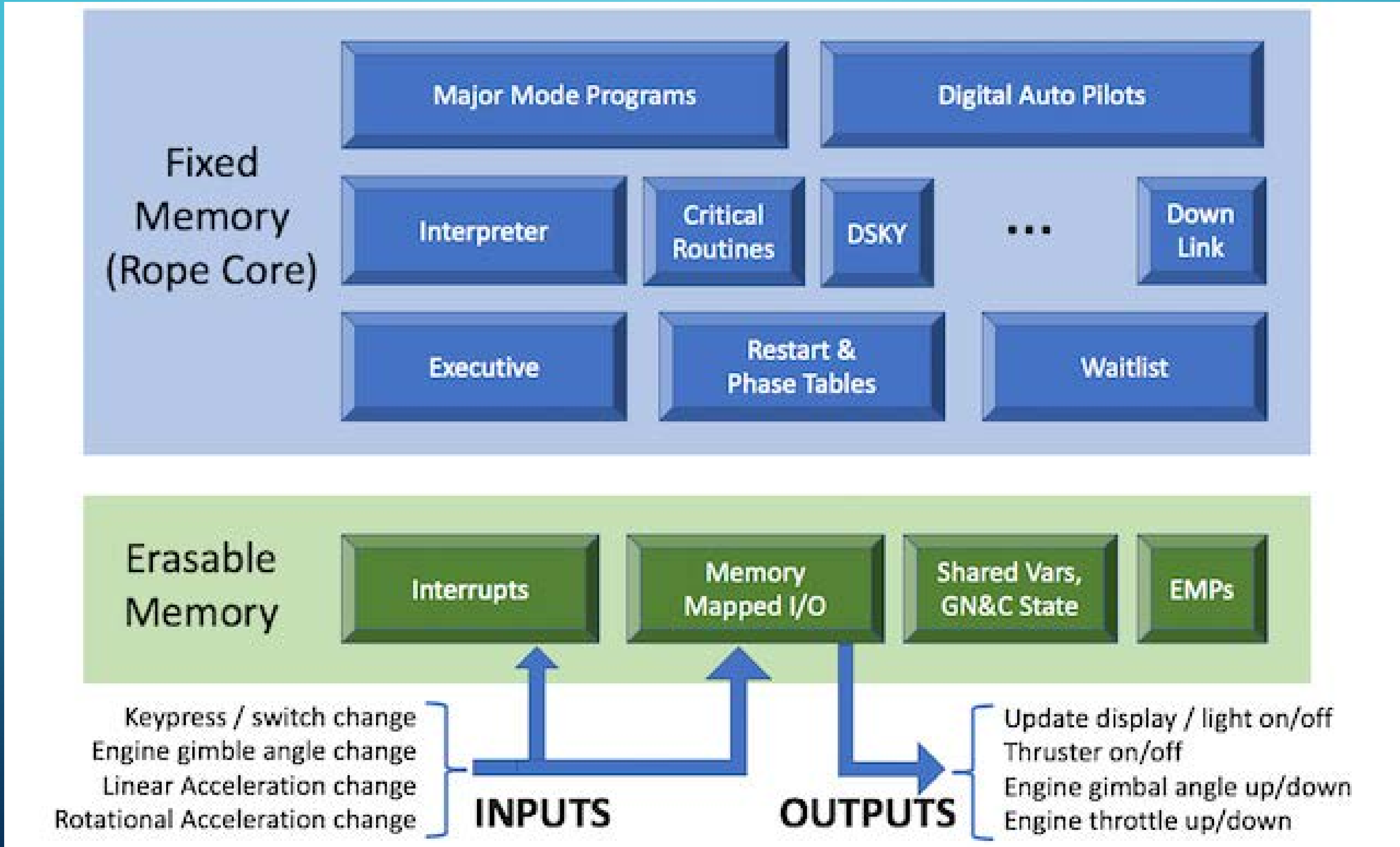
A PERFORMANCE PORTABILITY CHALLENGE

- Same c… …igurations in whicl…




If the time required to drive Ė to the 5.625 deg./sec. rate be less than 0.020 sec., then no jets are fired
Attitude-error rate Ė
Zone B
Zone
Zone C
D
Zone A
-E - 0.5/a netneg Ė^2 + DB1 = 0
5.625 deg./sec.
a netneg
Attitude error E
-DB2
-11.25 deg.
0
DB1
11.25 deg.
FINELAW applies to the dotted area; ROUGHLAW applies to the region external to it
Zone A
a netpos
-5.625 deg./sec.
Zone
Zone B
Zone C
D
Coast region
Thrust region
E - 0.5/a netpos Ė^2 + DB2 = 0

AGC SOFTWARE "STACK"



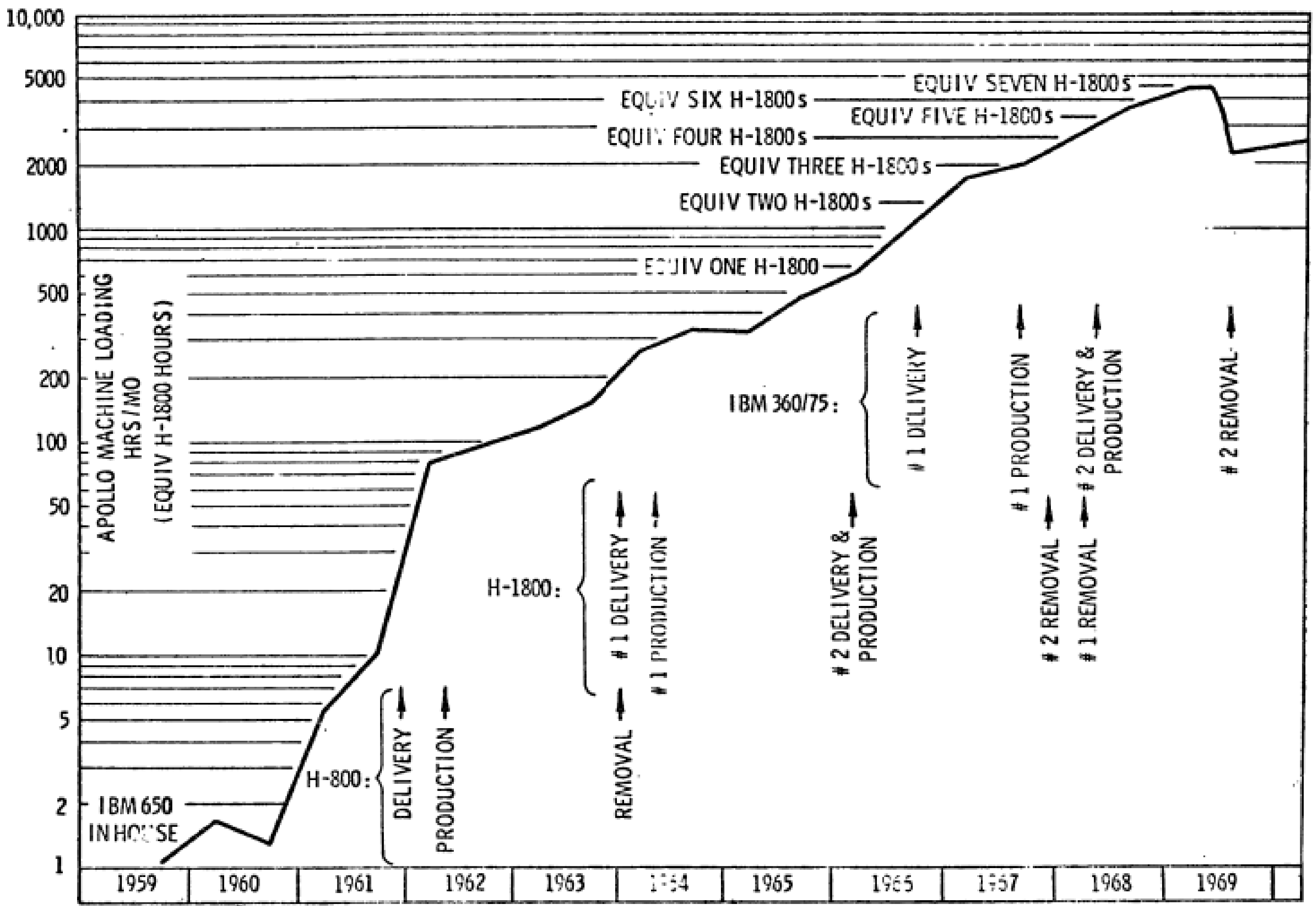
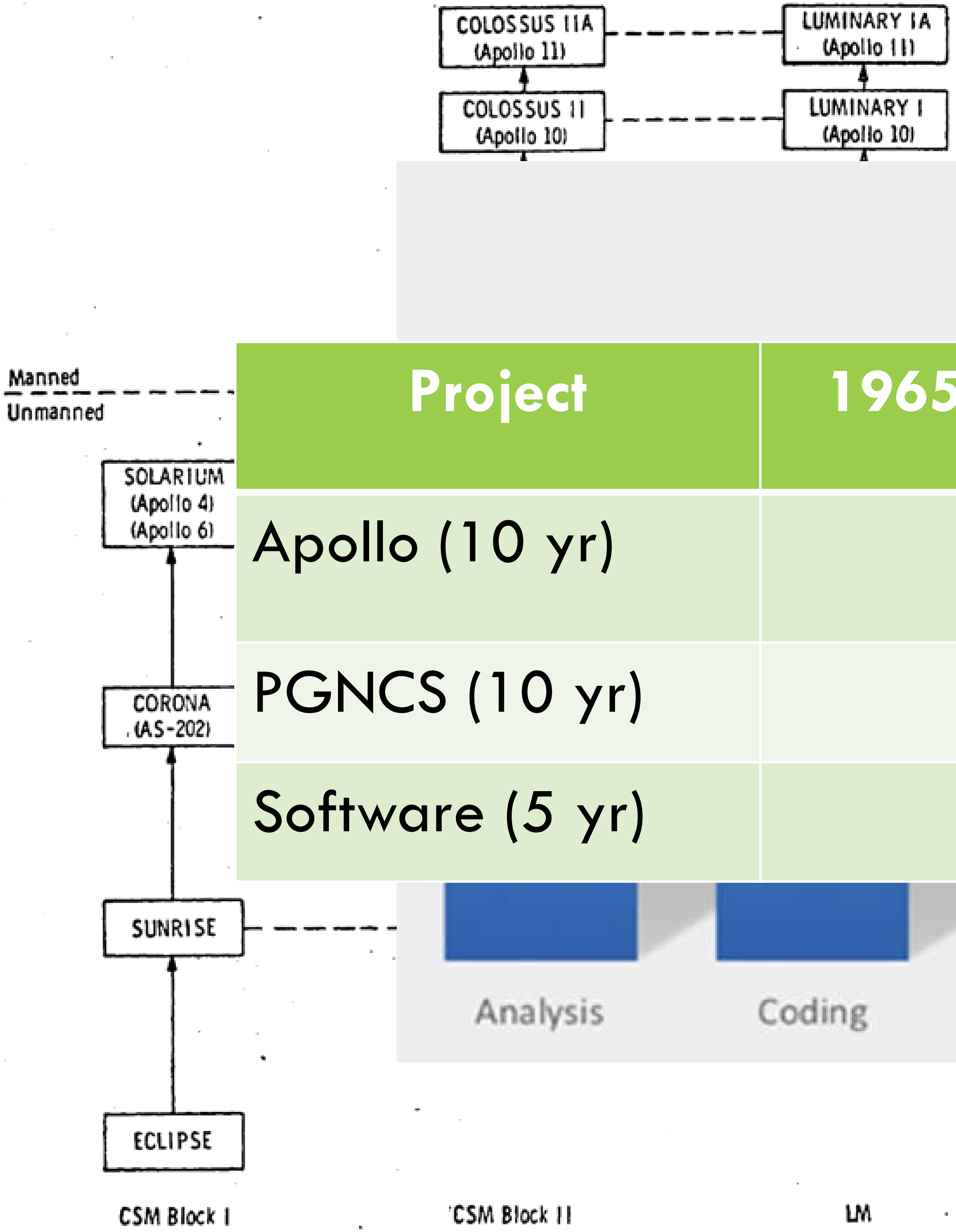
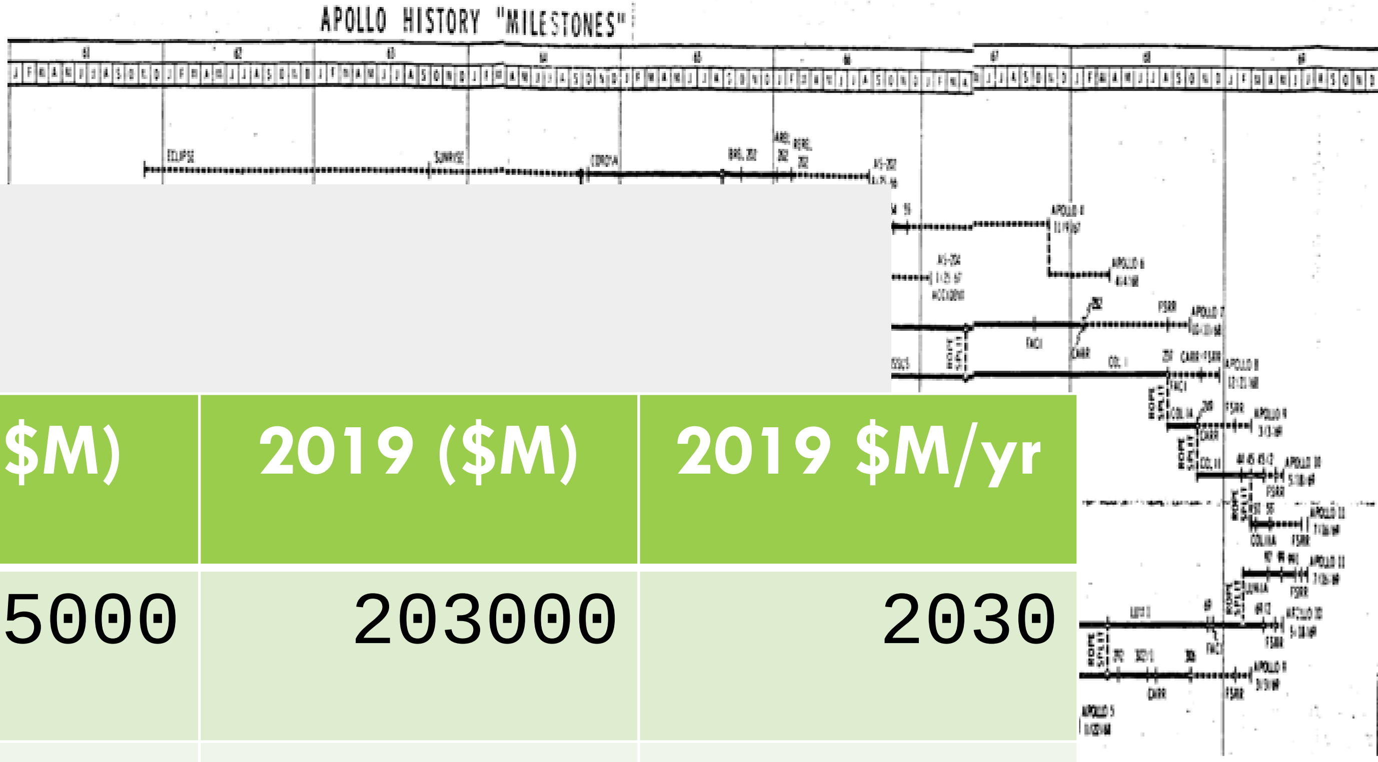
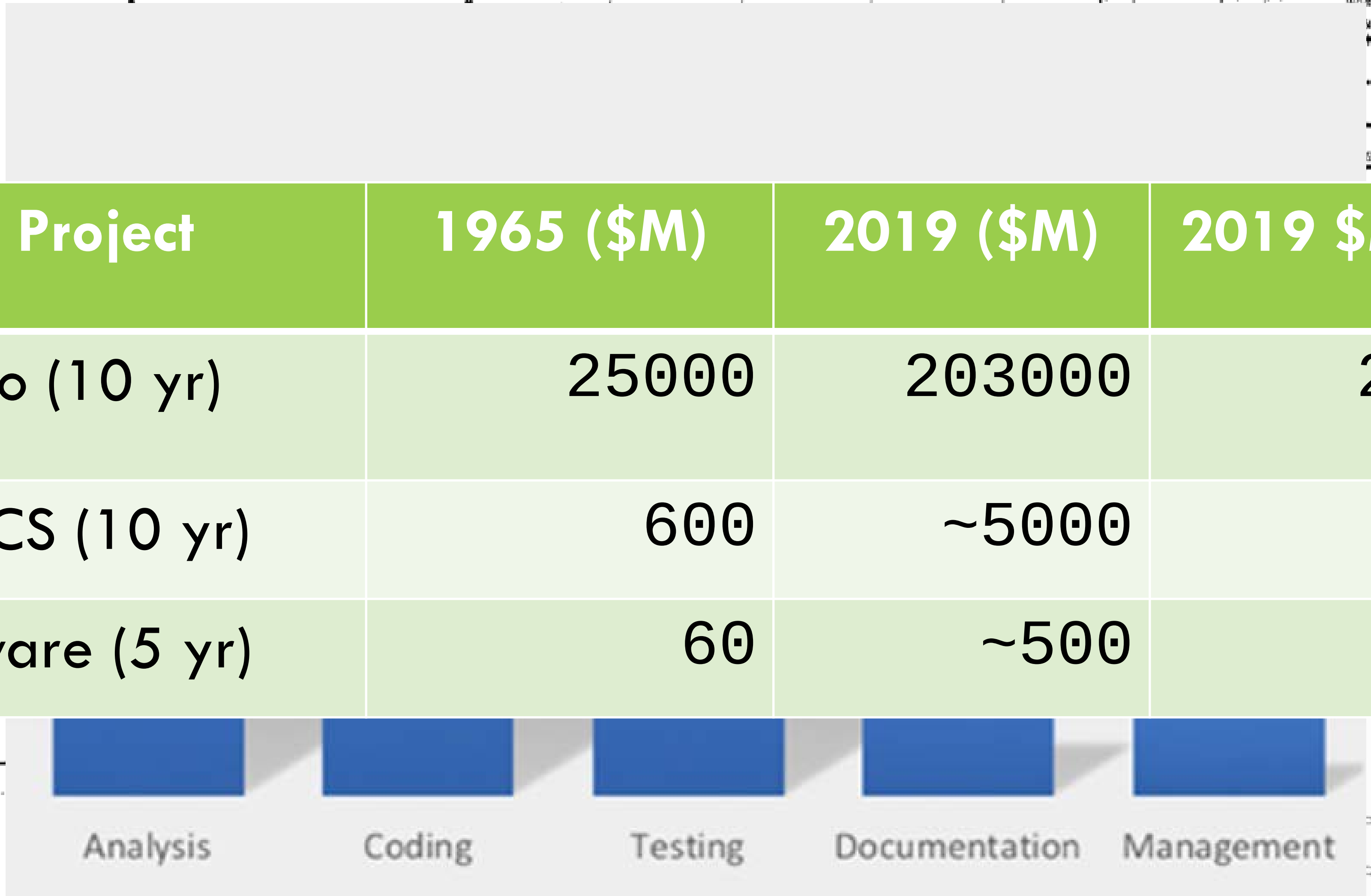


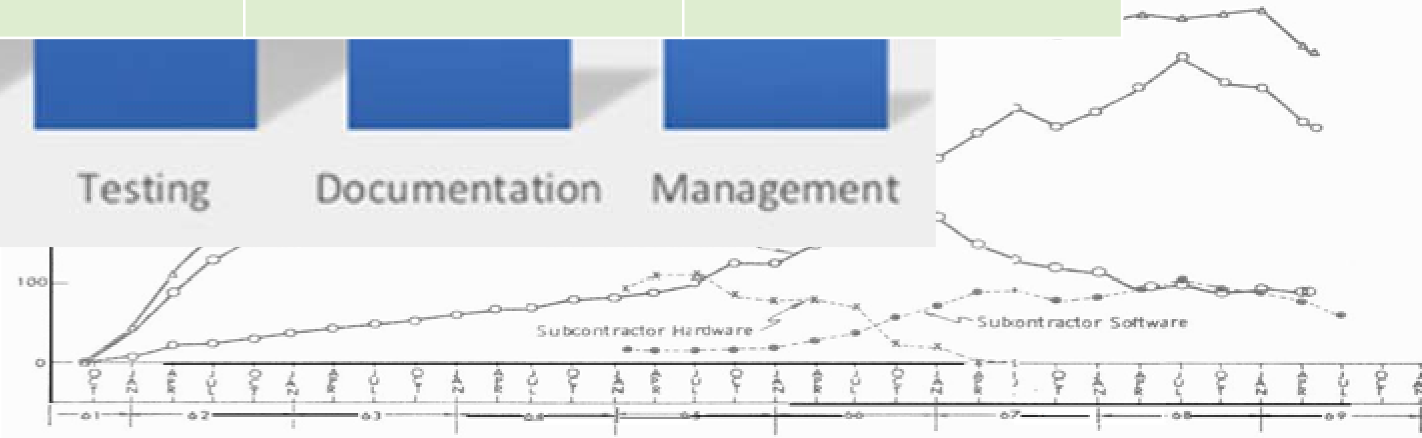
Figure 8. Computer usage by the Draper Laboratory Apollo effort.

-53-









Project	1965 ($M)	2019 ($M)	2019 $M/yr
Apollo (10 yr)	25000	203000	2030
PGNCS (10 yr)	600	~5000	500
Software (5 yr)	60	~500	100

"The need for formal validation rose with the size of the software. Programs of 2,000 words took between 50 and 100 test runs to be fully debugged, and full-size flight program took from 1,000 to 1,200 runs."

"In the early stages, there were no *programmers*. Instead, engineers and scientists learned the techniques of programming.

It was believed that competent engineers could learn programming more easily than programmers could learn engineering."

"Throughout much of the Apollo effort, MIT experienced difficulty in estimating the time and effort required to design, test and verify successive mission programs."

"SOFTWARE ENGINEERING"

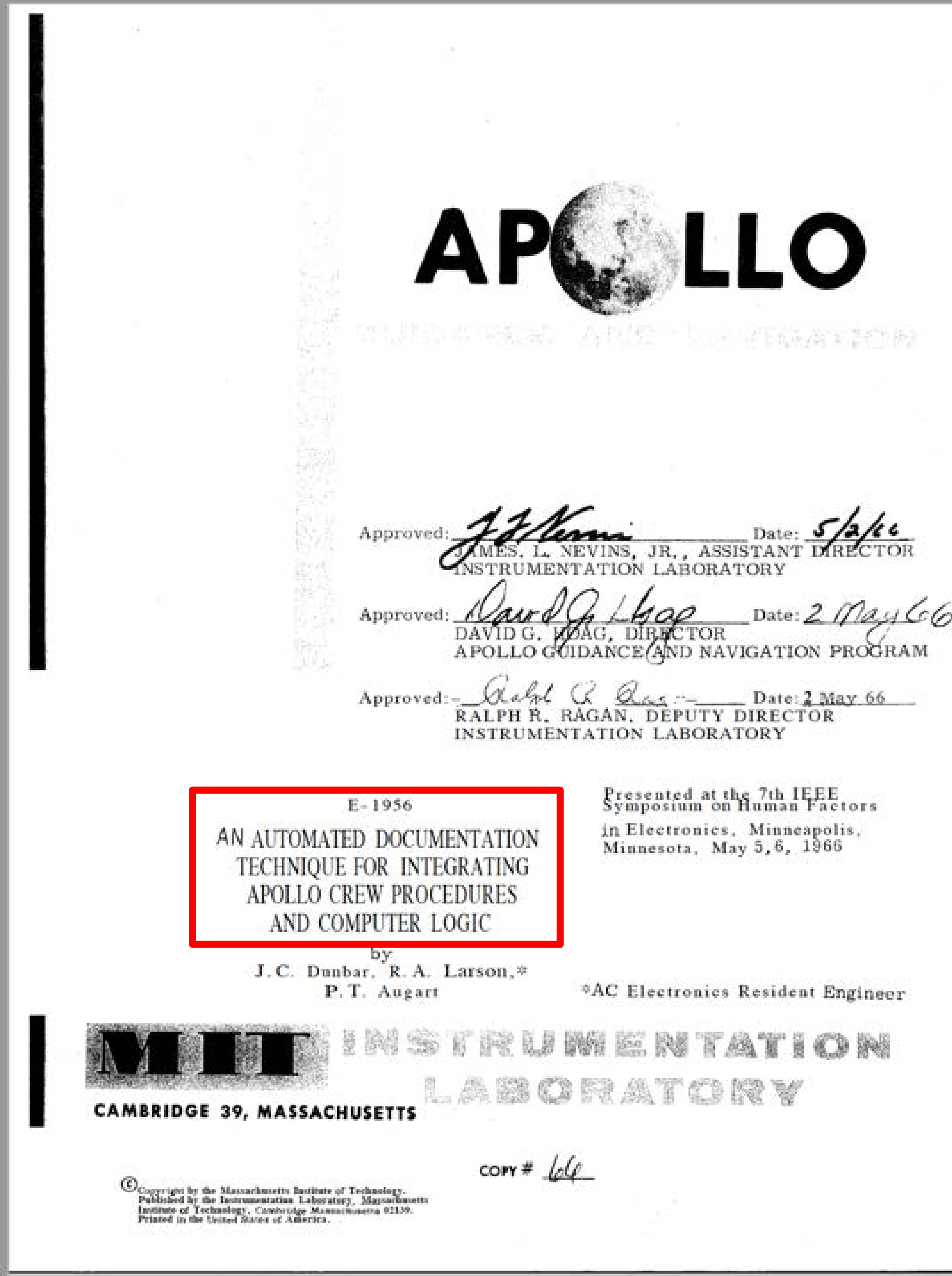
- Margaret Hamilton, lead developer of Lunar Module flight program introduced this term…

"…to bring the software [effort] legitimacy so that it and those building it would be given due respect"

"No one doubted the quality of the s
process used in development that ca

Five lessons were identified:

1. up-to-date documentation is crucia
2. verification must proceed through s
3. requirements must be clearly defin
4. good development plans should be
5. more programmers do not mean fa



APOLLO

Approved: _______ Date: 5/2/66
JAMES L. NEVINS, JR., ASSISTANT DIRECTOR
INSTRUMENTATION LABORATORY

Approved: _______ Date: 2 May 66
DAVID G. HOAG, DIRECTOR
APOLLO GUIDANCE AND NAVIGATION PROGRAM

Approved: _______ Date: 2 May 66
RALPH R. RAGAN, DEPUTY DIRECTOR
INSTRUMENTATION LABORATORY

E-1956

AN AUTOMATED DOCUMENTATION TECHNIQUE FOR INTEGRATING APOLLO CREW PROCEDURES AND COMPUTER LOGIC

by

J.C. Dunbar, R.A. Larson,*
P.T. Augart

Presented at the 7th IEEE Symposium on Human Factors in Electronics, Minneapolis, Minnesota, May 5,6, 1966

*AC Electronics Resident Engineer

MIT INSTRUMENTATION LABORATORY

CAMBRIDGE 39, MASSACHUSETTS

COPY # 66

OUTLINE

- Background
- Hardware Architecture
- Guidance Software
- Brief Detour
- Mission Applications

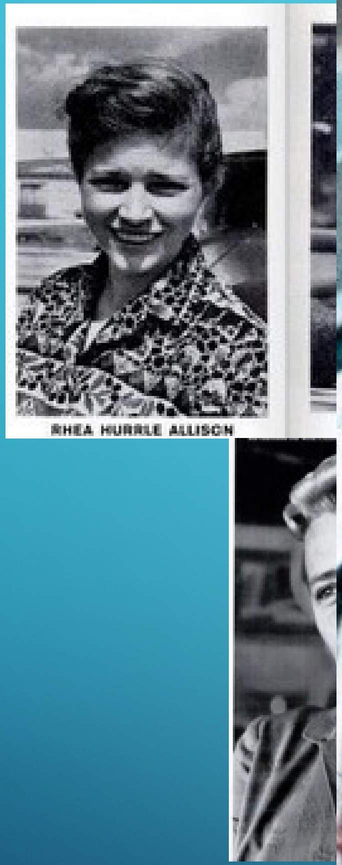
A BRIEF DETOUR

HISTORICAL CONTEXT

OFTEN GLOSSED OVER

OR TOTALLY IGNORED

Apollo Moonwalkers
Apollo G&N System (MIT-IL)
Apollo Astronauts
Apollo Mission Control
Gemini Astronauts

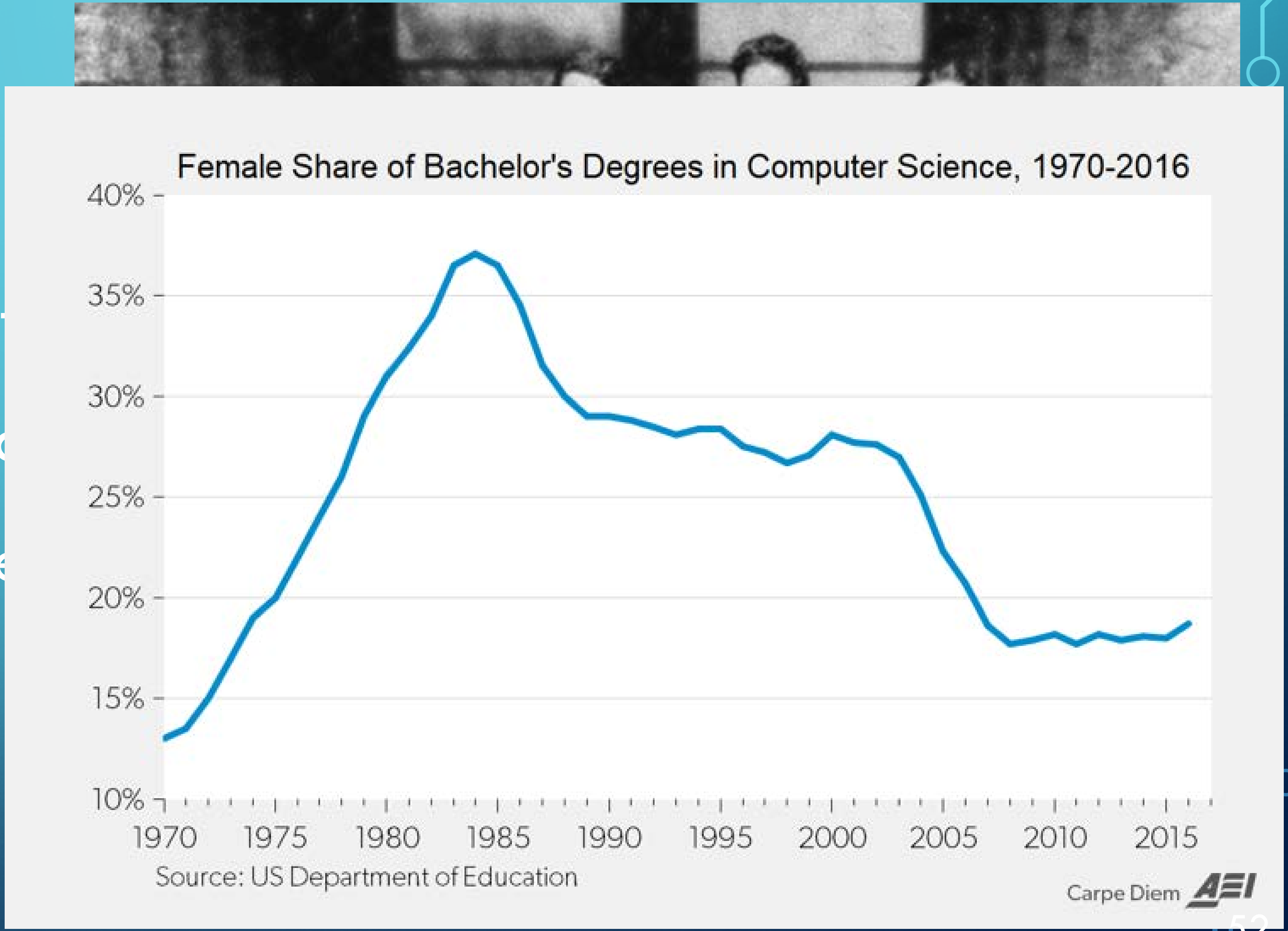


Mercury 13

Valentina Tereshkova – 3 days in orbit, 1963

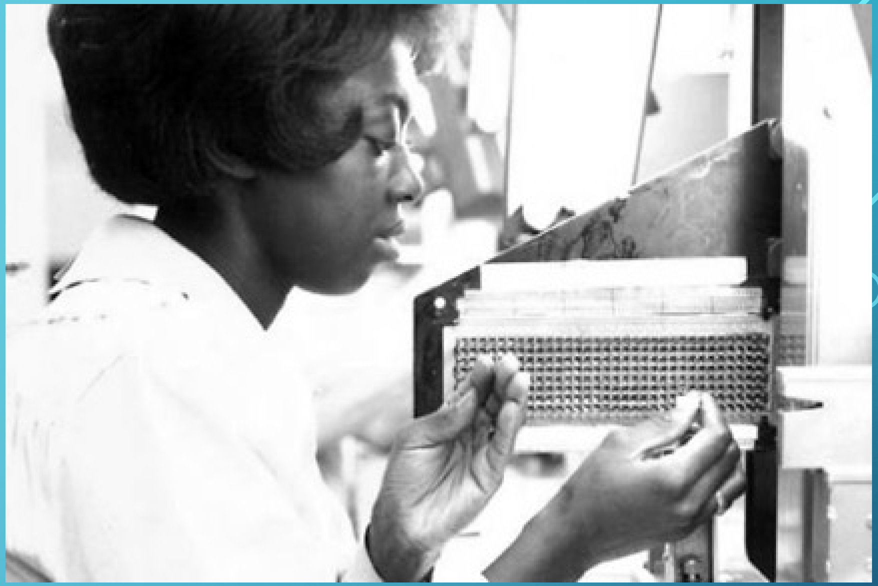
A BRIEF DETOUR: WOMEN AND COMPUTERS

- 1640-1950: "Comput
- Tedious calculation wo
- 1950-1960: Compute

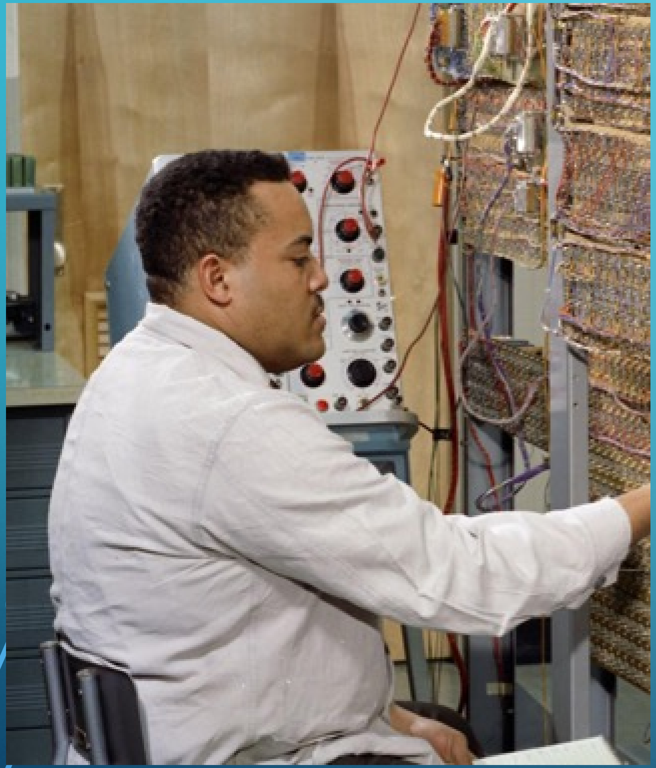


A BRIEF DETOUR
WOMEN
IN THE AGC PROJECT

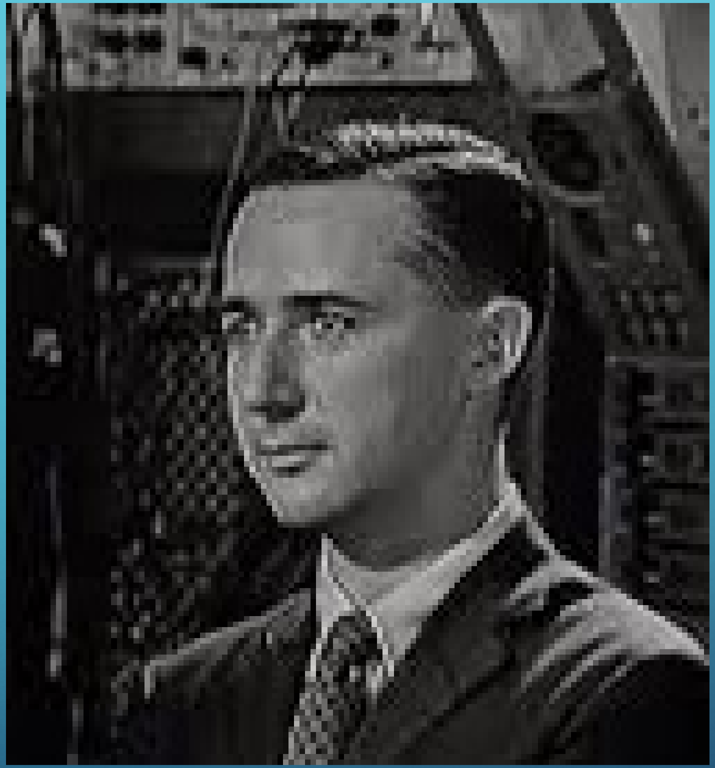
Margaret Hamilton, Phyllis Rye,
Saydean Zeldin, Elain Denniston



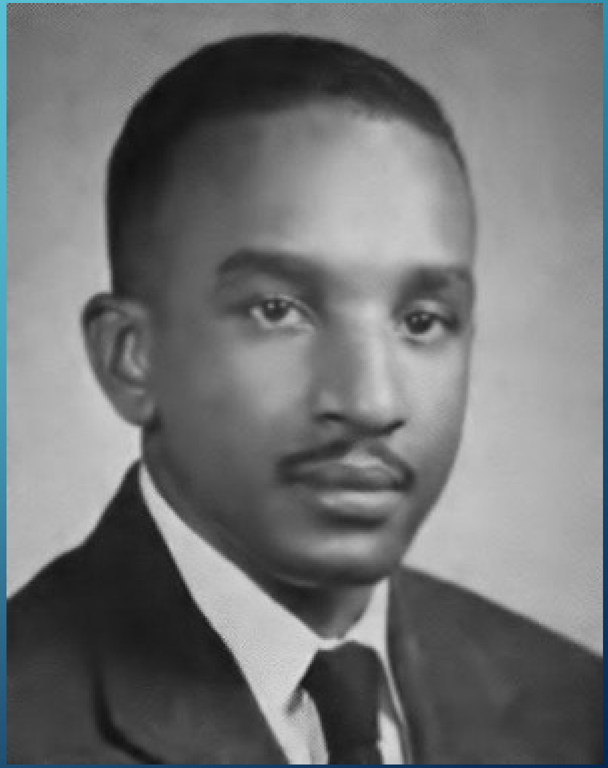
A BRIEF DETOUR
PEOPLE OF COLOR IN THE AGC PROJECT



William Mallory



Ramon Alonso



Robert Pinckney

A BRIEF DETOUR: WERNHER VON BRAUN

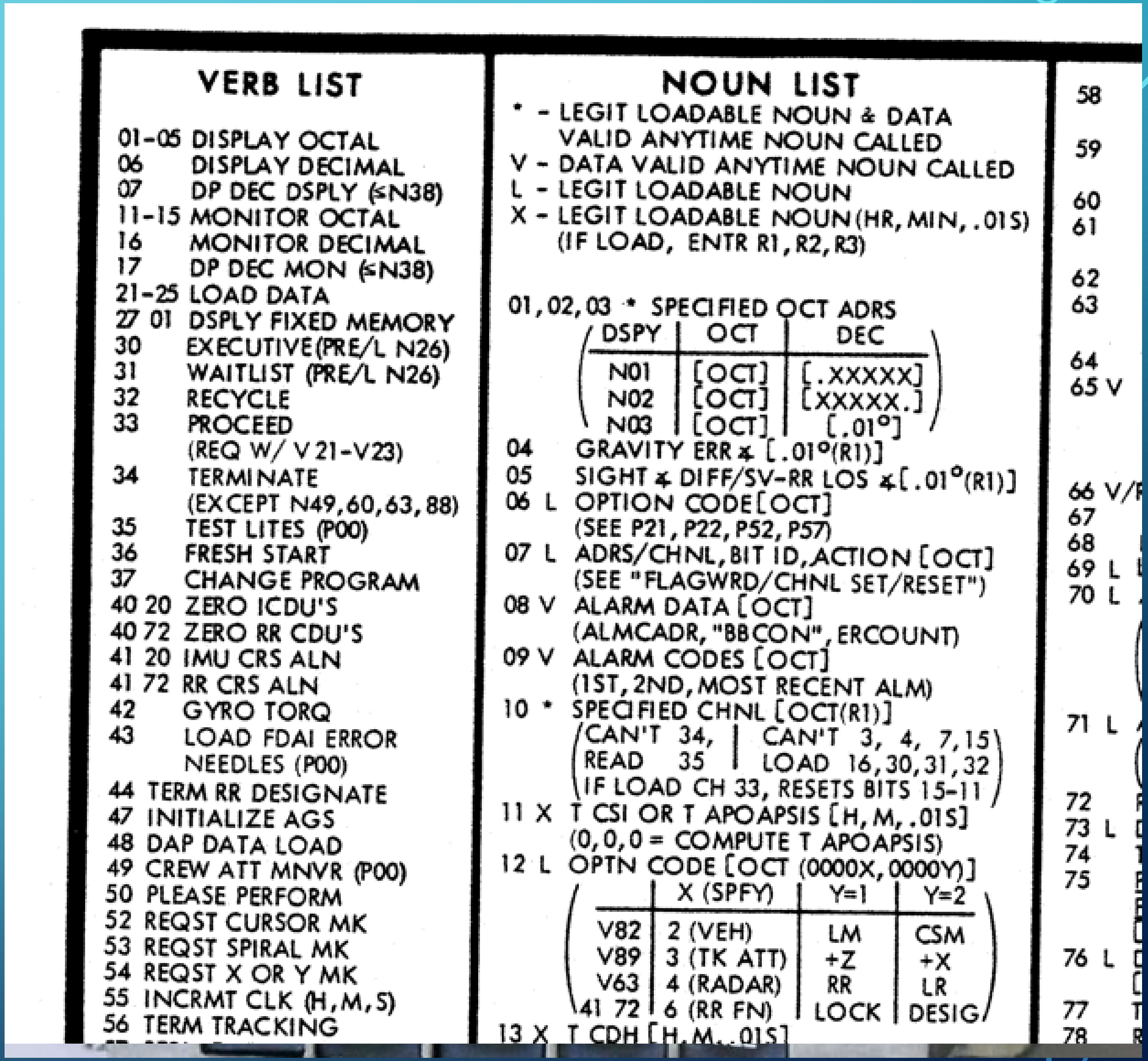
- Creator of V2 Rocket
- Member of NAZI Party; arrested for suspici
- Captured and brought to US with ~1,600 others in 1945
- Led development of F1 engine and Saturn booster
- Championed racial integration in Wallace's Alabama

OUTLINE

- Background
- Hardware Architecture
- Guidance Software
- Brief Detour
- Mission Applications

USER INTERFACE

- VERB – NOUN
- 3, 5 char line display
- Indicator Lights
- Two in CM, one in LM one in Huston



VERB LIST

01-05 DISPLAY OCTAL
06 DISPLAY DECIMAL
07 DP DEC DSPLY (≤N38)
11-15 MONITOR OCTAL
16 MONITOR DECIMAL
17 DP DEC MON (≤N38)
21-25 LOAD DATA
27 01 DSPLY FIXED MEMORY
30 EXECUTIVE (PRE/L N26)
31 WAITLIST (PRE/L N26)
32 RECYCLE
33 PROCEED (REQ W/ V21-V23)
34 TERMINATE (EXCEPT N49,60,63,88)
35 TEST LITES (P00)
36 FRESH START
37 CHANGE PROGRAM
40 20 ZERO ICDU'S
40 72 ZERO RR CDU'S
41 20 IMU CRS ALN
41 72 RR CRS ALN
42 GYRO TORQ
43 LOAD FDAI ERROR NEEDLES (P00)
44 TERM RR DESIGNATE
47 INITIALIZE AGS
48 DAP DATA LOAD
49 CREW ATT MNVR (P00)
50 PLEASE PERFORM
52 REQST CURSOR MK
53 REQST SPIRAL MK
54 REQST X OR Y MK
55 INCRMT CLK (H,M,S)
56 TERM TRACKING

NOUN LIST

* - LEGIT LOADABLE NOUN & DATA VALID ANYTIME NOUN CALLED
V - DATA VALID ANYTIME NOUN CALLED
L - LEGIT LOADABLE NOUN
X - LEGIT LOADABLE NOUN (HR, MIN, .01S) (IF LOAD, ENTR R1, R2, R3)

01,02,03 * SPECIFIED OCT ADRS

DSPY	OCT	DEC
N01	[OCT]	[.XXXXX]
N02	[OCT]	[XXXXX.]
N03	[OCT]	[.01°]

04 GRAVITY ERR ∡ [.01°(R1)]
05 SIGHT ∡ DIFF/SV-RR LOS ∡[.01°(R1)]
06 L OPTION CODE [OCT] (SEE P21, P22, P52, P57)
07 L ADRS/CHNL, BIT ID, ACTION [OCT] (SEE "FLAGWRD/CHNL SET/RESET")
08 V ALARM DATA [OCT] (ALMCADR, "BBCON", ERCOUNT)
09 V ALARM CODES [OCT] (1ST, 2ND, MOST RECENT ALM)
10 * SPECIFIED CHNL [OCT(R1)]
(CAN'T READ 34, 35 | CAN'T LOAD 3, 4, 7, 15, 16, 30, 31, 32 IF LOAD CH 33, RESETS BITS 15-11)
11 X T CSI OR T APOAPSIS [H, M, .01S] (0,0,0 = COMPUTE T APOAPSIS)
12 L OPTN CODE [OCT (0000X, 0000Y)]

	X (SPFY)	Y=1	Y=2
V82	2 (VEH)	LM	CSM
V89	3 (TK ATT)	+Z	+X
V63	4 (RADAR)	RR	LR
41 72	6 (RR FN)	LOCK	DESIG

13 X T CDH [H, M, .01S]

58
59
60
61
62
63
64
65 V
66 V/
67
68
69 L
70 L
71 L
72
73 L
74
75
76 L
77
78

AP
CR

- 1
- G
- D
- H
  A

MAY 1962 EDITION
GSA FPMR (41 CFR) 101-11.6

UNITED STATES GOVERNMENT

Memorandum

	Action	Info
R. RAGAN		✓
D. HOAG		✓
L. LARSON		✓
CENTRAL FILES		✓
E. Copps		✓
John Miller		✓
J. Dahlen		✓
[illegible]		✓
[illegible]		✓
R. Battin		✓
Alden Hall		✓
DUE DATE		

TO : See list

DATE: OCT 1 2 1966

FROM : FM/Deputy Chief, Mission Planning and Analysis Division

66-FM1-130

SUBJECT: Altitude and velocity limits imposed by the spacecraft computer program on the AS-503 mission

As you know, we are currently figuring on using the AS-278 spacecraft computer programs for AS-503. Ed Copps called me the other day to state that the orbital integration routines in the AS-278 program are scaled such that they will only work for altitudes less than about 5,400 nautical miles above the surface of the earth and velocities no greater than about 32,700 feet per second. (I am told the maximum values to be encountered in a nominal mission are about 3,900 nautical miles and 29,500 feet per second). He was looking for reassurance that this scaling would not present a constraint on the AS-503 mission, and I told him that I didn't think it would but I would check here at MSC. In the meantime, MIT is proceeding, assuming that these limits are not unacceptably restrictive for the AS-503 mission. If anyone knows a reason why this is not satisfactory, please let me know immediately.

Bill

Howard W. Tindall, Jr.

APOLLO 11

- Russian Luna 15
- Bad communications
- Program alarms & restarts
- Boulder Field
- Ascent engine arm CB
- Gimbal lock at rendezvous and switch to AGS

APOLLO 14

- Abort switch hack

APOLLO 8

- Entering a pre-launch program, P01 while in flight can crash the AGC
- Lovell practices using the space sextant, Accidentally enters P01 instead of star 01.
- Corrupts some guidance parameters in AGC erasable memory.

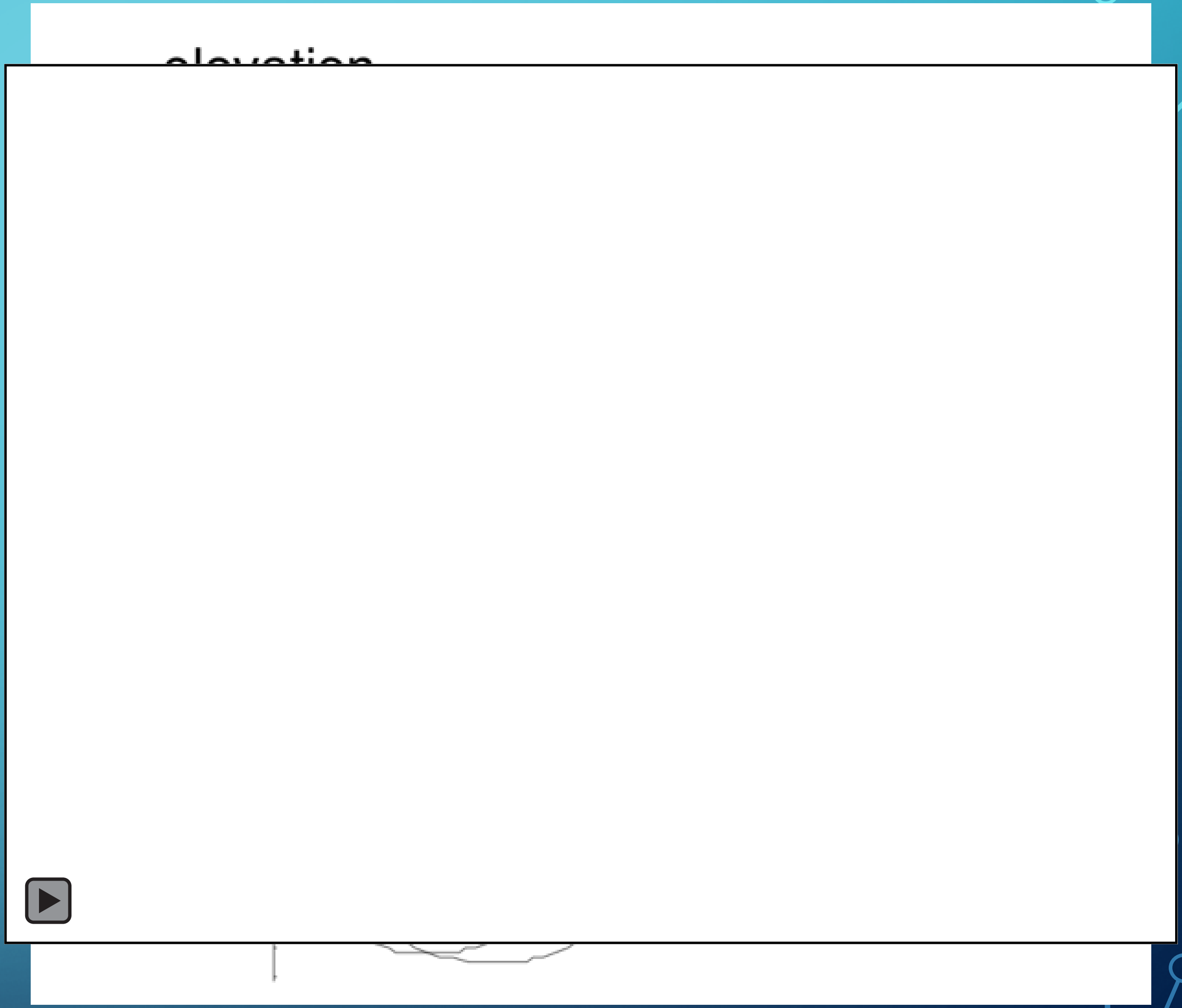
APOLLO 9

- First use of Erasable Memory Program (EMP) in crewed flight

- LM descent engine test configuration

APOLLO 10

- Barbeque mode troubles
- Full-up lunar descent abort test
- AGS in “AUTO” not “ATT-HOLD” (video)

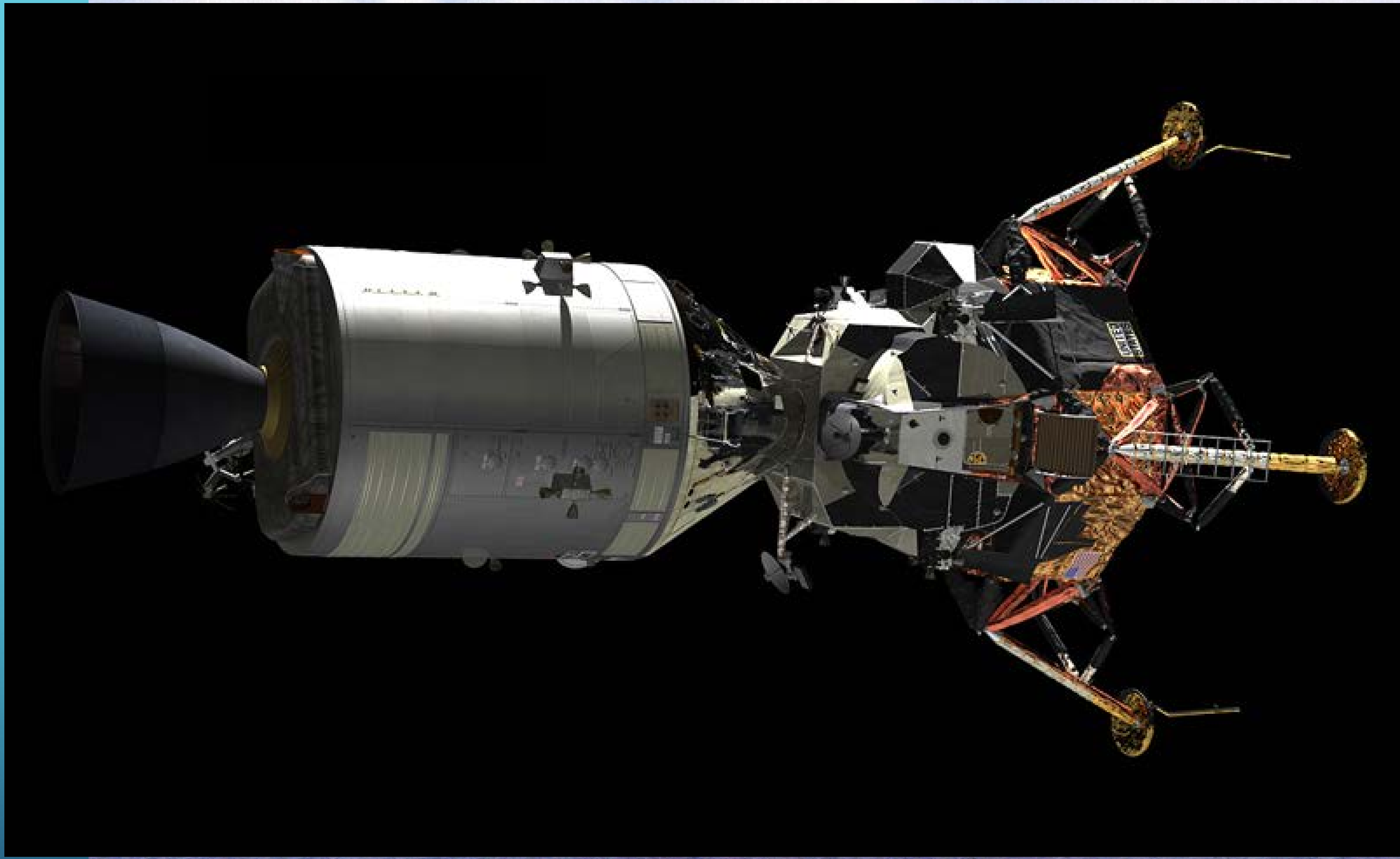


APOLLO 12

- Lightning strike
- Landing accuracy

APOLLO 13

- What-if thinking
- Three burns to get home



APOLLO 15

- Landing over lunar mountain range

- Added a simple terrain model for landing radar

COMPUTING AND SPACEFLIG

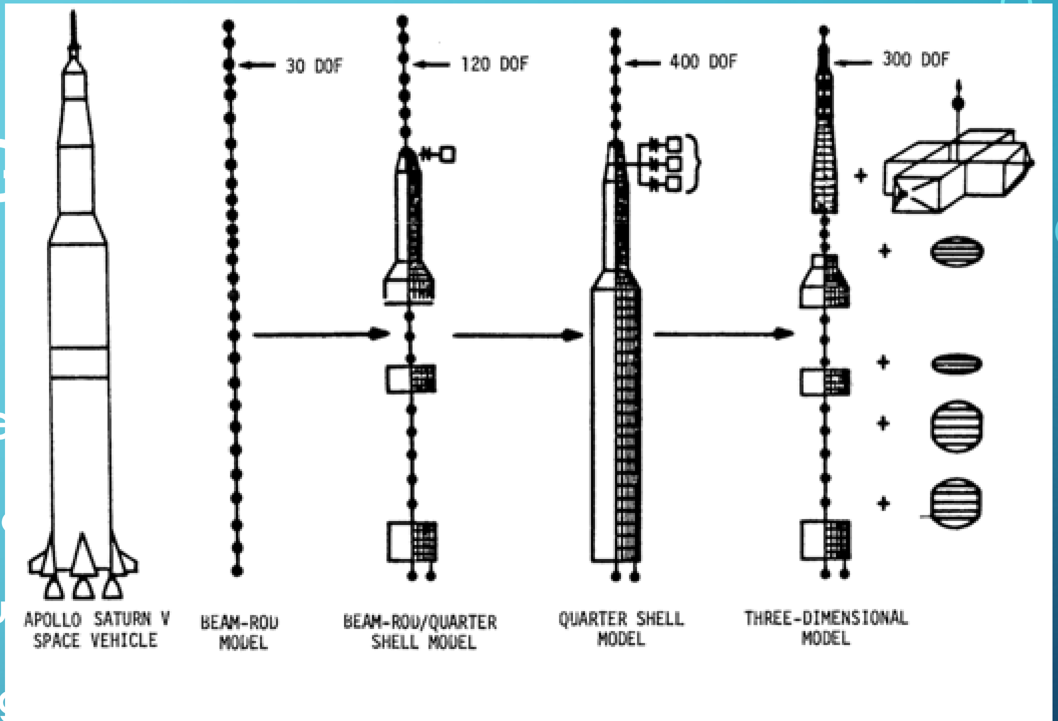
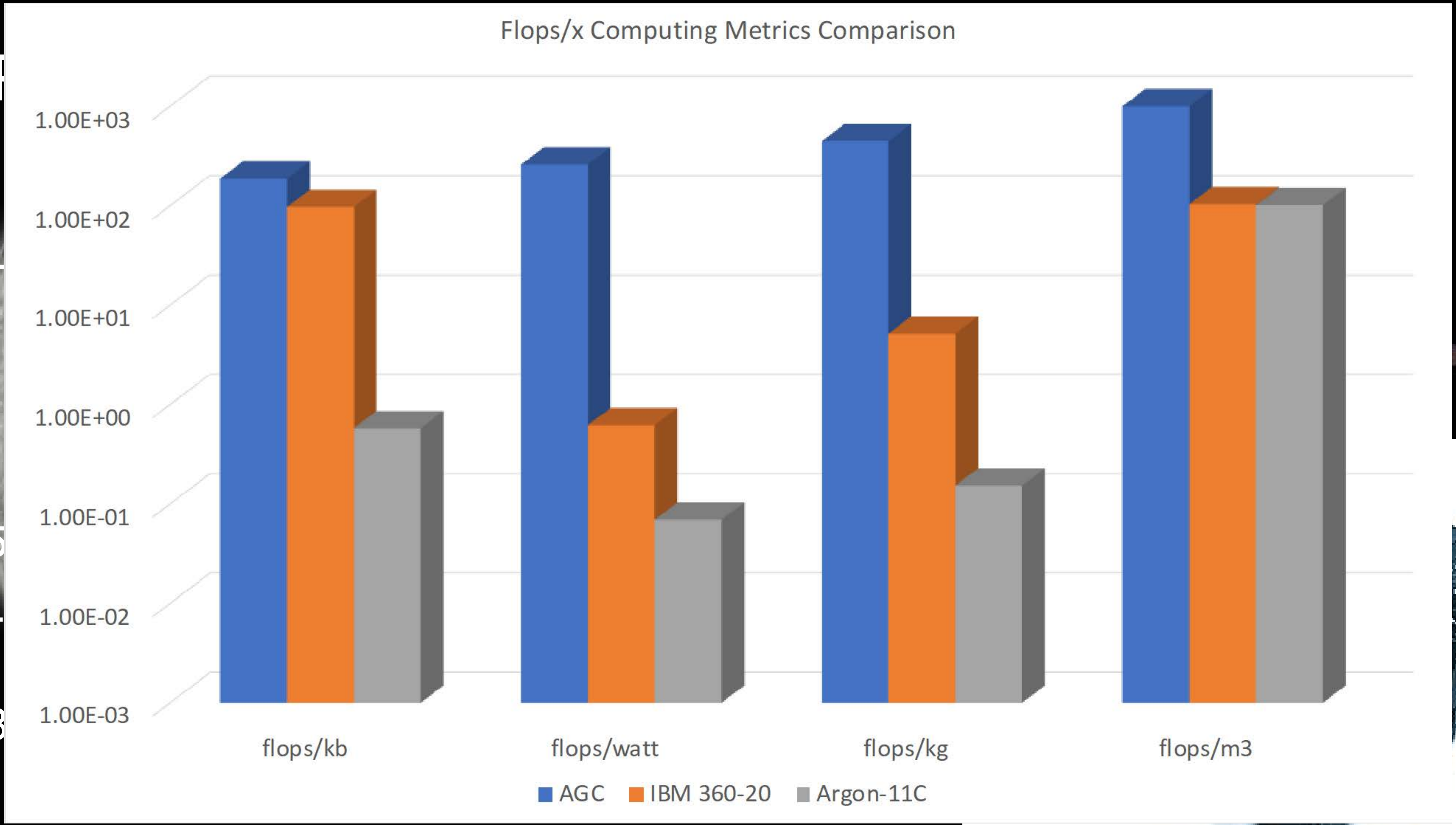


FIGURE 1-7 MATH MODEL EVOLUTION

- Computing was an essential tool in all aspe
  - Simulation and modeling used in all major vehi
  - Digital and Analog computers for Training simu
  - Real time computing complex (RTCC) for missin
- Computing deficiencies are a key reason Russia was unable to match US
- Apollo both drove innovations in computing and benefited from them


Flops/x Computing Metrics Comparison
1.00E+03
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
flops/kb
flops/watt
flops/kg
flops/m3
AGC
IBM 360-20
Argon-11C

RESOURCE LINKS

- bssw.io blog post
- Mercury 13 (Netfilx doc)
- AGC Restoration
- AGC Source Code on GitHub
- Virtual AGC Project
- Ultimate AGC Talk
- Spaceflight Computing History
- AGC Software Cost Model
- Hidden Figures (the book)