KING ETHELRED'S
COLLEGE OF ENGINEERING

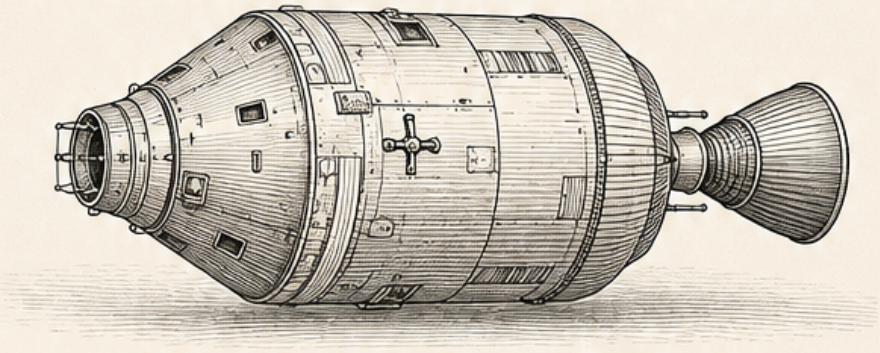
Established in 978 (ish)

DEPARTMENT OF AERONAUTICAL ENGINEERING
AND APPLIED MATHEMATICS

A COMPARATIVE ANALYSIS OF COMPUTATIONAL CAPABILITY:

THE APOLLO GUIDANCE COMPUTER, MODERN SMARTPHONES, AND THE APOLLO PROGRAM COMPUTING COMPLEX

into raw computational resources, memory, storage and I/O



A dissertation submitted in partial fulfilment of the requirements for the degree of

Bachelor of Engineering (Honours)

in Aeronautical Engineering

BY

H. A. FOTHERINGHAM

B.Sc. (Hons.), King Ethelred's

UPPER PIDDLINGTON - SEPTEMBER 2026

Fortitudine.
Ingenio.
Tea and biscuits.

EDITORIAL NOTE

King Ethelred's College of Engineering is a fictional institution. The institutional styling, author and departmental flourishes are parody; the technical analysis, numerical values and cited source material in this dissertation are intended to be accurate. Historical data are drawn primarily from NASA technical reports and mission documentation. Contemporary smartphone data are taken from manufacturer specifications.

The comparison is deliberately framed around three distinct claims that are commonly collapsed into one another: (1) raw computational capability, memory and storage; (2) suitability for deterministic, safety-critical spacecraft control; and (3) the much broader assertion that a single phone exceeds the computing power of the entire Apollo programme. The first can be quantified with reasonable confidence, the second is an architectural question, and the third is undefined unless a boundary and metric are specified.

Departmental motto: Fortitudine. Ingenio. Tea and biscuits.

ABSTRACT

This dissertation examines the claim that a modern smartphone possesses vastly more raw computational capability, memory and storage than the Apollo Guidance Computer (AGC), together with the counterclaim that such comparisons are a false equivalence because the AGC possessed superior input/output (I/O) processing. It also considers the broader statement that a phone has more computing power than the entire Apollo programme. The Block II AGC contained 2,048 words of erasable memory and 36,864 words of fixed core-rope memory; it used a 2.048 MHz master oscillator, an 11.72 microsecond memory cycle and an ADD time of approximately 23.4 microseconds [1-3]. By comparison, the representative contemporary platform used here - a 12 GB, up-to-1 TB Galaxy S25 Ultra using the Snapdragon 8 Elite family - combines multi-gigahertz CPU cores with dedicated GPU, NPU, image-signal, sensing and modem subsystems, UFS 4.0 storage, Wi-Fi 7 and cellular links whose peak rates are measured in gigabits per second [10-12].

The analysis finds that the raw-resource claim is correct by very large margins. A 12 GB phone has approximately three million times the erasable-memory capacity of one AGC, while a 1 TB device has about 13.6 million times the physical byte capacity of the AGC's fixed core-rope memory. Clock frequency alone is more than two thousand times higher and greatly understates the architectural gap because modern processors are multicore, deeply pipelined, superscalar and supplemented by heterogeneous accelerators. Nevertheless, a phone is not a drop-in replacement for an AGC. The AGC's I/O, power, timing model, software executive, electrical interfaces and qualification were built specifically around deterministic spacecraft guidance and control.

The claim that the AGC had far more I/O processing ability than a phone is not supported when 'ability' means aggregate data throughput or total I/O processing capacity. The Apollo 11 1201/1202 alarms in fact demonstrate the opposite limitation: rendezvous-radar counter interrupts could consume approximately 15% of all available processor time, on top of a nominal peak descent workload of about 90%, because those I/O events were serviced by the central processor [3]. What the AGC possessed was specialised and predictable control I/O, not greater aggregate I/O horsepower. The wider 'entire Apollo programme' statement is directionally plausible if interpreted as instantaneous raw throughput, but technically indeterminate without a defined inventory, time boundary and performance metric. Apollo relied on a distributed computing estate including two spacecraft AGCs, the Saturn V Launch Vehicle Digital Computer, five IBM System/360 Model 75 systems in Houston's Real Time Computer Complex, Goddard mainframes, remote tracking-site processors, simulation systems and development computers [5-9].

Principal conclusion: the modern phone wins overwhelmingly on raw resources and throughput; the AGC wins only in the sense that it was engineered, verified and integrated for a specific real-time spacecraft-control role. The achievement of Apollo computing is therefore not hidden superiority over modern hardware, but exceptional engineering under severe constraints.

Keywords: Apollo Guidance Computer; AGC; Apollo 11; 1201 alarm; 1202 alarm; real-time computing; input/output; smartphones; embedded systems; Mission Control.

SUMMARY OF FINDINGS

Claim audit - technical verdicts

A modern phone has vastly more raw CPU capability, memory and storage than one AGC	SUPPORTED
An AGC can be replaced by an unmodified consumer phone in a lunar module	NOT SUPPORTED
The AGC had far more aggregate I/O processing capability than a modern phone	NOT SUPPORTED
The AGC had better purpose-built deterministic spacecraft I/O and control interfaces	SUPPORTED
A phone has more compute than “the entire Apollo Program”	PLAUSIBLE BUT UNDEFINED

The key distinction is raw throughput versus fitness for a safety-critical real-time control role.

Figure S.1 - Summary claim audit. “Supported” does not mean equivalent in purpose; it means the proposition is technically defensible under the stated metric.

The social-media disagreement is best understood as a category error. One side is comparing resources: memory capacity, storage, processor throughput and general I/O bandwidth. The other side is implicitly comparing fitness for purpose: deterministic scheduling, direct interfaces to inertial sensors and actuators, safety behaviour and flight qualification. Both dimensions matter, but they are not interchangeable.

A modern smartphone can contain millions of times more memory and storage than an AGC while still being unsuitable, in unmodified form, for direct control of a lunar module. Conversely, an AGC can be an exquisitely appropriate flight computer without possessing more raw I/O processing capacity than a modern mobile SoC. The Apollo 11 overload episode is particularly useful because it provides a measured historical case: two radar coupling-data units could generate up to 12,800 counter pulses per second; each pulse consumed one 11.72 microsecond memory cycle, so the unwanted I/O load could consume about 15% of the computer [3].

The study therefore rejects the phrase false equivalence when it is applied to the narrow statement that a phone has more raw compute, memory and storage than an AGC. That statement is simply a resource comparison and is true. It accepts, however, that the comparison becomes misleading if listeners infer that a consumer phone and the AGC are equivalent systems, or that raw processor throughput alone measures the engineering difficulty of landing on the Moon.

CONTENTS

EDITORIAL NOTE 2

ABSTRACT 3

SUMMARY OF FINDINGS 4

NOMENCLATURE AND CONVENTIONS 8

Bit and byte convention 8

1 INTRODUCTION 9

1.1 Background 9

1.2 Research questions 9

1.3 Scope and method 9

2 THE APOLLO GUIDANCE COMPUTER 11

2.1 Design objective 11

2.2 Memory architecture 11

2.3 Processor and executive 11

2.4 Input/output and counter interrupts 13

2.5 Reliability and qualification 13

3 REPRESENTATIVE MODERN SMARTPHONE ARCHITECTURE 14

3.1 Reference platform 14

3.2 Heterogeneous system-on-chip design 14

3.3 Memory and storage 15

4 METHODOLOGY OF COMPARISON 16

4.1 Metrics that can be compared directly 16

4.2 Metrics that require interpretation 16

4.3 Three comparison layers 16

4.4 Uncertainty and source hierarchy 16

5 QUANTITATIVE RESOURCE COMPARISON 17

5.1 Core numerical comparison 17

5.2 Clock frequency is a conservative indicator 17

5.3 Memory as an operational constraint 17

5.4 "Raw compute" without a universal benchmark 18

5.5 Physical efficiency 18

6 INPUT/OUTPUT: WHERE THE REBUTTAL GOES WRONG 19

6.1 Defining I/O processing ability 19

6.2 The AGC counter mechanism 19

6.3 Determinism versus throughput 19

7 APOLLO 11 CASE STUDY: 1201 AND 1202 20

7.1 What actually happened 20

7.2 Why the spacecraft did not crash 20

7.3 The lesson for modern systems 20

8 THE WIDER APOLLO COMPUTING COMPLEX 21

8.1 The phrase "entire Apollo Program" 21

8.2 Houston Real Time Computer Complex 21

8.3 Goddard and the tracking network 21

8.4 Saturn V onboard computing 21

8.5 Can the whole mission estate be beaten by one phone? 22

9 DISCUSSION 23

9.1 Why "false equivalence" is too strong 23

9.2 Dave's strongest point 23

9.3 Dave's weakest point 23

9.4 Why the AGC is more impressive when described accurately 23

10 CONCLUSIONS 24

10.1 Findings 24

10.2 Final answer to the disputed claim 24

REFERENCES 25

APPENDIX A WORKED CALCULATIONS 26

A.1 AGC physical memory capacity 26

A.2 Apollo 11 radar-interrupt arithmetic 26

APPENDIX B CLAIM-BY-CLAIM AUDIT 27

APPENDIX C ENGINEERING PERSPECTIVE 28

NOMENCLATURE AND CONVENTIONS

Symbol / term	Meaning
AGC	Apollo Guidance Computer
CMC	Command Module Computer (an AGC)
LGC	Lunar Guidance Computer (an AGC)
DSKY	Display and Keyboard
CDU	Coupling Data Unit
IMU	Inertial Measurement Unit
RTCC	Real Time Computer Complex, Mission Control Houston
LVDC	Launch Vehicle Digital Computer, Saturn V Instrument Unit
I/O	Input/output
CPU	Central Processing Unit
SoC	System on Chip
NPU	Neural Processing Unit
ISP	Image Signal Processor
UFS	Universal Flash Storage
KiB	2^10 bytes
GiB	2^30 bytes
GB / TB	Decimal manufacturer capacity unless stated otherwise

Bit and byte convention

Apollo sources usually describe memory in words, not modern bytes. A Block II AGC word physically occupied 16 bits including parity; operationally, the word comprised sign/data or instruction fields plus parity. For simple storage-capacity comparisons this thesis reports the physical bit capacity: 2,048 x 16 bits = 4,096 bytes of erasable memory and 36,864 x 16 bits = 73,728 bytes (72 KiB) of fixed memory. This is intentionally explicit because popular accounts variously quote “2K words”, “4 KB”, “36K words” and “72 KB”, all referring to the same hardware from different conventions [1,4].

1 INTRODUCTION

1.1 Background

Comparisons between Apollo-era computing and modern consumer electronics have become a recurring shorthand for technological progress. The attraction is obvious: the Apollo programme achieved crewed lunar landings using computers that appear, by modern measures, extraordinarily small and slow. Yet the shorthand often combines several different propositions. “A phone has more computing power than the AGC” can mean raw arithmetic throughput, memory capacity, storage capacity, data movement, functional capability, mission reliability, or simply that the phone is newer and more complicated. Each interpretation produces a different answer.

The narrow claim examined first in this dissertation is unambiguous: a current smartphone possesses vastly more raw computational resources, memory and storage than one Apollo Guidance Computer. That can be tested directly against documented AGC specifications and contemporary manufacturer data. A second claim - that the AGC had far more I/O processing ability than a phone - requires more care because “I/O processing ability” is not a standard scalar metric. It may refer to aggregate bandwidth, number of interfaces, deterministic response, electrical compatibility with actuators, or the ability to continue controlling a spacecraft under overload.

1.2 Research questions

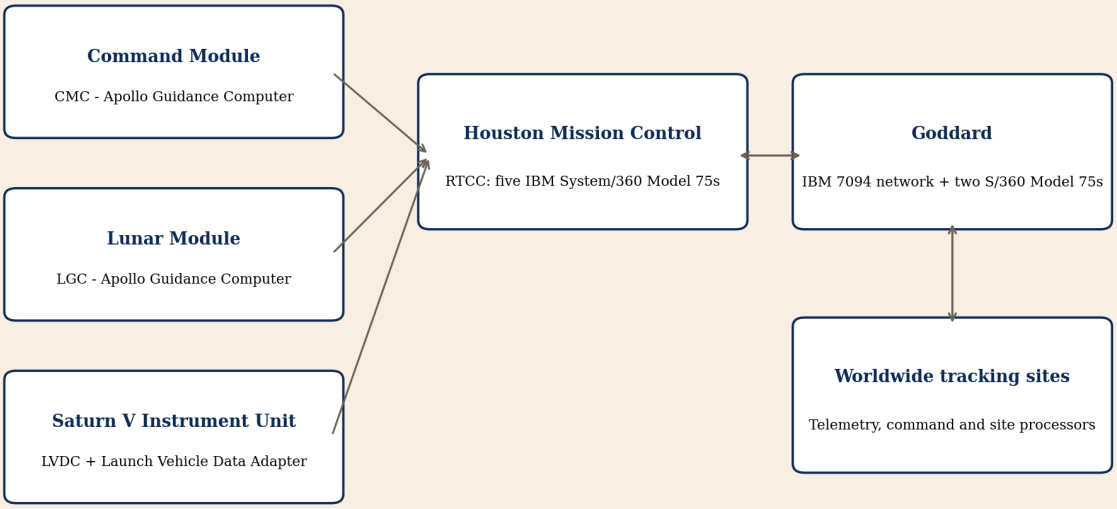
1. How large is the raw resource gap between a Block II AGC and a representative modern smartphone?
2. Does the AGC's specialised I/O architecture imply greater aggregate I/O processing capacity than a phone?
3. What do the Apollo 11 1201/1202 alarms reveal about the AGC's real-time strengths and throughput limitations?
4. How should the phrase 'more compute than the entire Apollo Program' be interpreted, and can it be defended quantitatively?

1.3 Scope and method

The principal historical sources are NASA mission reports, NASA Technical Reports Server documents, Apollo programme histories and IBM material describing the ground computing estate. Contemporary data are limited to manufacturer-published specifications for the Snapdragon 8 Elite mobile platform and Samsung Galaxy S25 Ultra. The intention is not to construct a synthetic benchmark between incompatible instruction sets; instead, the study compares documented capacities and timing scales, then analyses architecture and workload behaviour.

This distinction is essential. MIPS, TOPS, clock frequency and link bandwidth are useful within appropriate contexts but cannot be collapsed into one universal number. The AGC used fixed-point arithmetic, one’s-complement representation, a memory-cycle-dominated architecture and a priority executive. A modern phone uses 64-bit out-of-order cores, deep cache hierarchies, SIMD/vector execution and dedicated accelerators. Any single-number comparison of “compute” therefore contains assumptions. Where an exact cross-generation metric would be misleading, this dissertation states the limitation rather than manufacturing precision.

Apollo was a distributed computing system, not a single AGC



Any claim about "the entire Apollo Program" must define which machines, which instant, and which metric.

Figure 1.1 - The Apollo mission computing environment was distributed across spacecraft, launch vehicle, Mission Control, Goddard and remote sites. The diagram is conceptual rather than an exhaustive inventory.

2 THE APOLLO GUIDANCE COMPUTER

2.1 Design objective

The AGC was not designed as a general-purpose laboratory mainframe miniaturised for space. It was an embedded guidance and control computer whose defining constraints were mass, power, reliability, deterministic interaction with sensors and actuators, and the need for meaningful autonomy at lunar distance. NASA histories note that the round-trip communication delay and the demands of lunar landing made an onboard digital computer strategically important [13]. The Command Module and Lunar Module each carried an AGC using essentially the same hardware but different flight software [2].

The Block II computer weighed about 70 lb and occupied roughly a cubic foot. NASA's Lunar Module technical reference gives a 2.048 MHz basic clock oscillator, a 12 microsecond memory cycle and a 24 microsecond add time; the more detailed mission documentation quotes 11.72 and 23.4 microseconds respectively [2-4]. These figures do not make the AGC "weak" in context. They describe a machine carefully matched to navigation equations, control loops, crew interaction and the manufacturing technology of the mid-1960s.

2.2 Memory architecture

The Block II AGC divided memory into 2,048 words of erasable magnetic-core memory and 36,864 words of fixed core-rope memory [1]. The fixed memory physically encoded programme bits by routing conductors through or around magnetic cores. This made the flight software unusually resistant to accidental modification, but it also meant that software changes late in the programme carried substantial manufacturing and verification consequences.

Erasable memory held changing navigation state, counters, temporary data and the working storage required by the executive. The scarcity of writable memory shaped software practice. Apollo programmers reused locations at different mission phases, overlaid data structures and compressed functionality aggressively. Such practices would be undesirable in a contemporary safety-critical system if abundant memory were available, but they were rational responses to a severe resource budget.

2.3 Processor and executive

The AGC's processor and memory were tightly coupled. Many operations required one or more 11.72 microsecond memory cycles, and an assembly-language ADD took about two cycles. Rather than treating the AGC as a simple sequential calculator, the software presented a multiprogrammed environment with a priority/event-driven asynchronous executive [4]. Jobs were scheduled by priority, while higher-priority interrupts could pre-empt running work.

This priority model is central to the Apollo 11 story. The executive did not attempt to preserve every scheduled activity under overload. It protected high-priority navigation and control functions and allowed less essential work to be dropped or restarted. In modern terminology, this is graceful degradation by criticality. The conceptual resemblance to contemporary real-time systems is stronger than any resemblance to a desktop timesharing operating system.

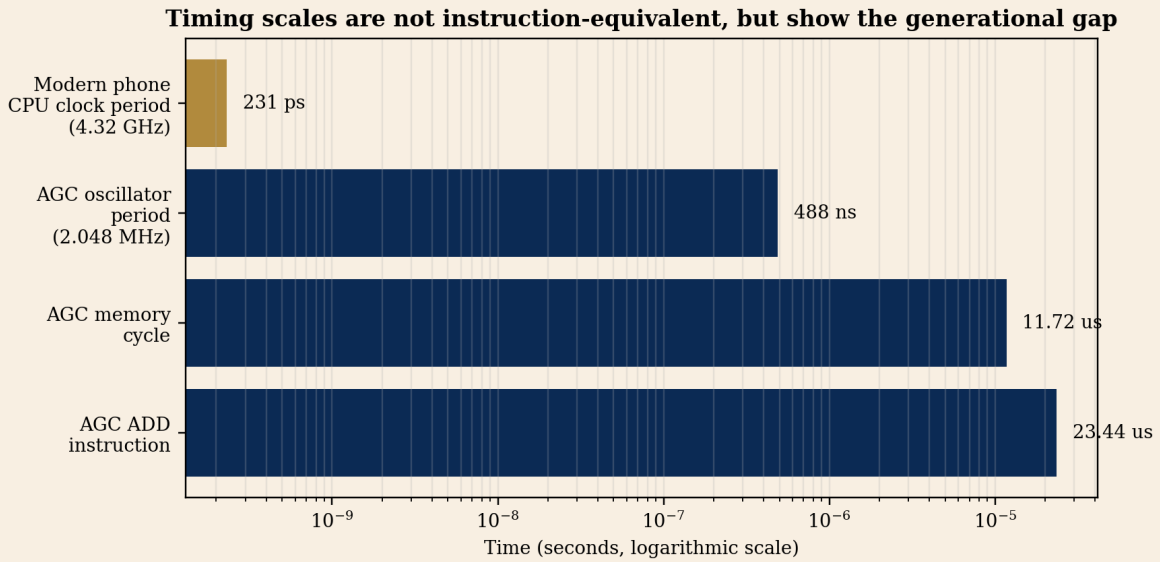


Figure 2.1 - AGC timing versus a current mobile CPU clock period. The plot is intentionally labelled "not instruction-equivalent": clock periods alone are not a benchmark, but they show why a modern device has a vast timing margin.

2.4 Input/output and counter interrupts

The AGC communicated with the guidance system through a mixture of ordinary I/O channels, discrete signals and hardware-assisted counters. High-frequency inputs such as accelerometer pulses and coupling-data-unit increments were treated as counter interrupts. NASA's Apollo 11 mission report explains that such an input temporarily suspended lower-priority computation for one memory cycle while the pulse was processed [3]. This mechanism was elegant because it offered predictable handling of physical events without a separate modern-style peripheral processor. It also meant that sufficiently intense I/O activity consumed central-processor time directly.

That last point is important when evaluating claims of superior AGC I/O processing. The architecture was excellent for the rate and criticality of the expected spacecraft signals. It was not a high-bandwidth I/O engine in the contemporary sense. Its strength lay in determinism, simplicity and integration with the executive. An event could arrive, pre-empt ordinary work, update a counter and return control in a known time. This is qualitatively different from streaming multi-gigabit radio or camera data through dedicated DMA engines and accelerators.

2.5 Reliability and qualification

Reliability is another dimension that raw performance comparisons ignore. Hall's reliability history documents an exceptionally controlled design, production and maintenance programme, including 17 computers that flew missions and additional flight-type units used in prelaunch activity [14]. Contemporary consumer phones are reliable products, but they are not ordinarily qualified as single-string guidance computers for radiation, spacecraft power quality, mission thermal environment, launch vibration and certified actuator control.

This does not give the AGC more computational capability. It gives it a different assurance case. A hypothetical spacecraft computer built around modern silicon would normally use rad-hard or carefully screened components, redundant channels, fault containment and safety-certified software rather than an off-the-shelf handset. The correct lesson is therefore that performance and mission suitability are orthogonal engineering dimensions.

3 REPRESENTATIVE MODERN SMARTPHONE ARCHITECTURE

3.1 Reference platform

For a concrete comparison this thesis uses a representative 2025-2026 flagship Android configuration: Samsung's Galaxy S25 Ultra, available with 12 GB of memory and up to 1 TB of storage, coupled with the Snapdragon 8 Elite family [10-12]. The exact handset is less important than the order of magnitude. Even mid-range devices exceed the AGC by enormous margins in memory and storage; selecting a flagship merely makes contemporary I/O architecture easier to document because Qualcomm publishes detailed subsystem specifications.

3.2 Heterogeneous system-on-chip design

A modern phone is not simply "a faster CPU". It is a heterogeneous computing platform containing CPU cores, GPU, neural-processing hardware, image-signal processors, audio engines, a sensing hub, security processors, storage controllers, GNSS, cellular modem and Wi-Fi/Bluetooth/UWB connectivity. The Snapdragon 8 Elite product brief describes prime CPU frequencies up to 4.32-4.47 GHz, performance cores up to 3.53 GHz, LPDDR5x support, UFS 4.0 storage, triple 18-bit image signal processors, a sensing hub with dedicated micro-NPUs, peak cellular downlink up to 10 Gbit/s and Wi-Fi 7 up to 5.8 Gbit/s [10,11].

This distributed structure matters more than the headline clock. Camera pixels need not be handled one interrupt at a time by the main CPU. Modem baseband processing, image pipelines, sensor fusion and graphics are heavily offloaded. Direct-memory-access mechanisms move large blocks without executing a central-CPU instruction per byte. Consequently, aggregate phone I/O is the product of many concurrent subsystems, not a single processor servicing every event serially.

Different design objectives: deterministic control versus heterogeneous high-throughput computing

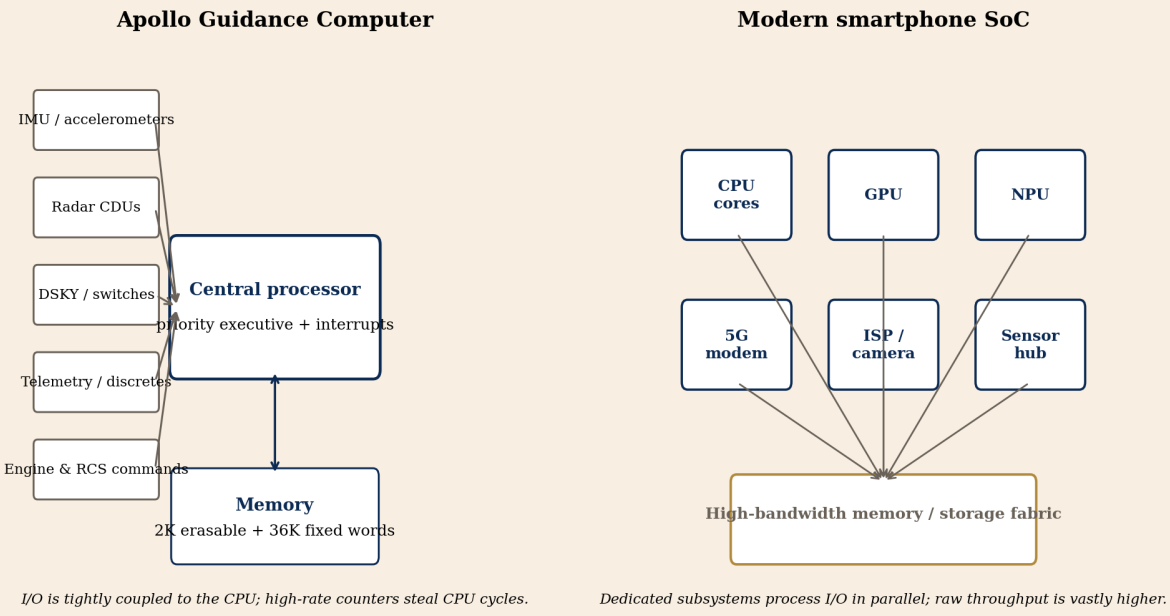


Figure 3.1 - Conceptual architecture contrast. The AGC centralises much time-critical I/O through processor-serviced channels and counters; the phone distributes high-volume I/O across specialised blocks.

3.3 Memory and storage

Samsung specifies 12 GB of memory for the Galaxy S25 Ultra and storage options up to 1 TB [12]. Using decimal manufacturer units, 12 GB divided by the AGC's 4,096 bytes of physical erasable-memory capacity is approximately 2.93 million. Using binary GiB conventions yields approximately 3.15 million. The result is so large that the choice of convention does not affect the conclusion.

For storage, a 1 TB phone divided by 73,728 physical bytes of AGC fixed memory gives approximately 13.56 million. The comparison is imperfect because core-rope is executable read-only memory whereas phone flash is writable mass storage, but the claim at issue is simply capacity. A modern handset can also hold caches and on-chip memories that individually exceed the total writable storage of the AGC.

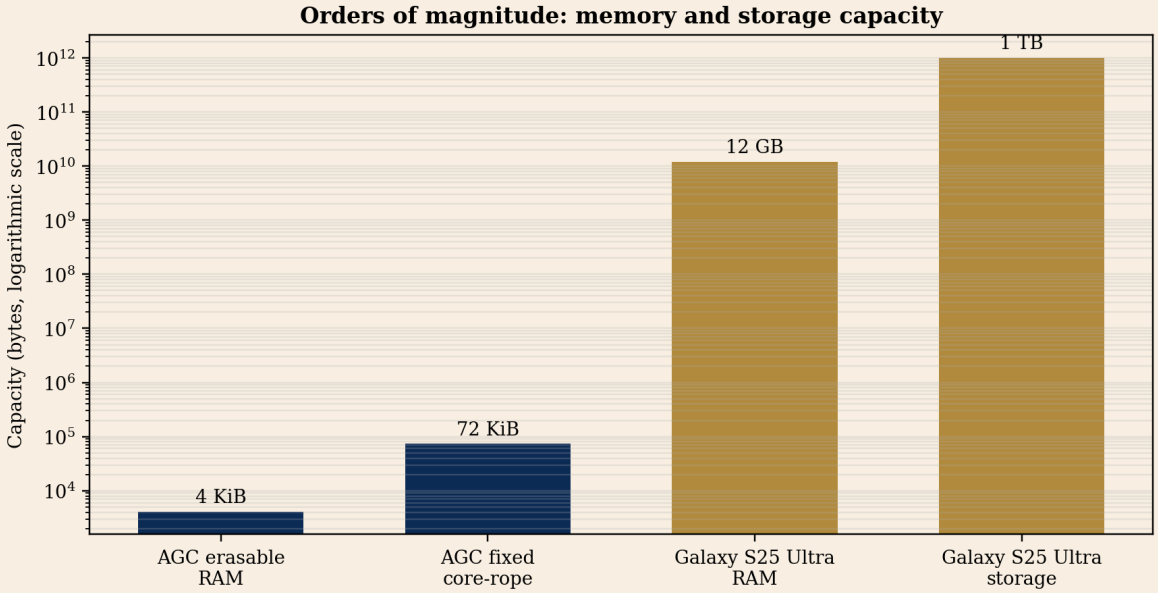


Figure 3.2 - Capacity comparison on a logarithmic scale. The modern device is separated from the AGC by millions of times, not by a modest generational improvement.

4 METHODOLOGY OF COMPARISON

4.1 Metrics that can be compared directly

Memory capacity, fixed-program storage capacity, oscillator/clock frequency and documented link rates are physical quantities and can be compared directly provided the units are stated. Such comparisons are useful for establishing scale. They do not by themselves establish application performance.

4.2 Metrics that require interpretation

Instructions per second are architecture-dependent. An AGC instruction may embody a very different amount of work from an AArch64 instruction, while a modern core may issue several micro-operations per cycle, speculate, execute out of order and operate on wide vectors. Likewise, "I/O processing ability" can describe bytes per second, interrupt latency, number of independent channels, jitter, control-loop deadline assurance or electrical drive capability. The thesis therefore avoids producing a single synthetic "phone is X times faster" number where no common benchmark exists.

4.3 Three comparison layers

Layer	Question	Appropriate evidence
Resource capacity	How much memory/storage and what timing scale?	Words/bytes, clock, documented cycles
Architecture	How is work scheduled and I/O serviced?	Block diagrams, interrupts, dedicated accelerators
Mission fitness	Can the system safely control the spacecraft?	Interfaces, assurance, timing guarantees, qualification

The social-media dispute occurs because the first participant is usually speaking about layer one while the rebuttal jumps to layers two and three. Calling the original resource comparison a "false equivalence" is only justified if the original statement claimed equivalent mission suitability. If it merely says "more raw compute, memory and storage", then layer one is exactly the intended comparison.

4.4 Uncertainty and source hierarchy

Where NASA sources differ slightly in rounded values, the mission-era figures are retained with the range made explicit. For example, the LM news reference rounds the memory cycle to 12 microseconds and ADD to 24 microseconds, while detailed Apollo 11 documentation gives 11.72 and 23.4 microseconds. Modern platform data are taken from official Qualcomm and Samsung specifications rather than third-party benchmark databases. No attempt is made to infer RAM bandwidth for the exact handset where the manufacturer does not publish it.

5 QUANTITATIVE RESOURCE COMPARISON

5.1 Core numerical comparison

Metric	Apollo Guidance Computer	Representative modern phone	Interpretation
Writable memory	2,048 words = 4,096 physical bytes	12 GB	~2.9-3.1 million x capacity
Fixed/program memory	36,864 words = 73,728 physical bytes	Up to 1 TB flash storage	~13.6 million x capacity
Basic oscillator / peak CPU clock	2.048 MHz	4.32-4.47 GHz prime core	~2,100-2,180 x clock frequency
AGC memory cycle	11.72 us	Not directly analogous	AGC handles ~85,300 memory cycles/s
AGC ADD	~23.4 us	Modern integer add commonly cycle-scale	Instruction-equivalence not asserted
Cellular / Wi-Fi link	No analogous packet radio subsystem	Up to 10 Gbit/s cellular; 5.8 Gbit/s Wi-Fi	Phone aggregate I/O is vastly higher

5.2 Clock frequency is a conservative indicator

A 4.32 GHz clock is about 2,109 times the AGC's 2.048 MHz oscillator. That ratio substantially understates the raw compute difference because the AGC's useful operations span multiple oscillator periods and memory cycles, whereas a modern CPU uses multiple cores, wide execution machinery, deep pipelines, caches and speculative issue. A single modern clock period is about 0.231 nanoseconds; an AGC ADD takes about 23.4 microseconds, a ratio of roughly 101,000 in time. That ratio must not be misreported as "101,000 times faster" because the operations and architectural context are not equivalent, but it shows why treating both machines as one-instruction-per-clock processors is inappropriate.

5.3 Memory as an operational constraint

The memory ratio has practical meaning beyond headline capacity. The AGC executive, navigation state, temporary variables and mission-specific workloads all competed for roughly four physical kilobytes of erasable memory. Modern mobile operating systems routinely allocate more memory to a single graphical application than thousands of AGCs combined. This permits abstractions, redundancy, caches, complex drivers and large software stacks that Apollo programmers simply could not afford.

Yet abundant memory can reduce predictability. Dynamic allocation, caches and complex virtual-memory behaviour are acceptable in a consumer phone because occasional latency variation is tolerable. A safety-critical control computer is normally designed to bound those behaviours. Therefore the modern capacity advantage should not be confused with automatically better worst-case timing.

5.4 Raw compute without a universal benchmark

Popular discussions often convert AGC performance into a small MIPS number. Such conversions can be useful pedagogically but are fragile. The AGC's instruction timings vary, its interpreted mathematical language adds another layer, and modern processors execute instructions with very different semantics and degrees of parallelism. This study therefore concludes "many orders of magnitude" rather than assigning a false six-decimal multiplier. The independently observable evidence - clock scale, multicore architecture, vector/accelerator hardware and memory hierarchy - is sufficient to establish the direction with overwhelming margin.

The most conservative imaginable modern comparison would still favour the phone. Even if one ignored the GPU, NPU and all peripheral accelerators, treated modern execution as only one useful instruction per CPU cycle and considered only a subset of cores, billions of instruction opportunities per second remain available. By contrast, a 23.4 microsecond AGC ADD corresponds to roughly 42,700 additions per second if executed continuously with no other work. The purpose of this illustration is scale, not benchmark equivalence.

5.5 Physical efficiency

The AGC's approximately 70 lb mass and roughly 55 W flight power are themselves impressive for the era [2,15]. A modern smartphone system performs far more computation in a few watts of SoC power and a few hundred grams of total device mass. Semiconductor process scaling, integration and architectural evolution therefore improve not only throughput but also compute per unit mass, volume and energy by enormous factors. The AGC nevertheless met a far harsher assurance objective than a consumer handset.

6 INPUT/OUTPUT: WHERE THE REBUTTAL GOES WRONG

6.1 Defining I/O processing ability

The statement that the AGC had "far more I/O processing ability than a phone" sounds plausible because the AGC directly connected to inertial sensors, radar interfaces, crew controls and propulsion commands. But direct connection to mission equipment is not the same as aggregate I/O processing capacity. A modern phone simultaneously services image sensors, display pipelines, multi-gigabit radios, GNSS, inertial sensors, audio devices, storage, USB, touch input, biometric hardware and always-on sensing. Much of that traffic is handled by dedicated hardware rather than main-CPU interrupt service.

If "ability" means the ability to operate the Lunar Module's actual electrical interfaces with deterministic timing, then the AGC wins by definition because it was designed and wired for them. If "ability" means the amount of I/O data the system can process concurrently, the phone wins by an overwhelming margin. The original rebuttal silently switches between these definitions.

6.2 The AGC counter mechanism

The counter-interrupt system is a strong design feature but also exposes a throughput limit. According to the Apollo 11 mission report, high-frequency inputs such as accelerometer and radar coupling-data-unit pulses were given the highest priority. Each pulse caused lower-priority work to be suspended for 11.72 microseconds while the counter event was processed [3]. The design delivered bounded latency and simple semantics, but the CPU paid for every event.

Modern mobile systems invert that balance for high-volume streams. Image sensors feed an ISP; network radio signals are processed in modem hardware; storage transfers use controller logic and DMA; sensor hubs can run continuously without waking high-performance CPU cores. The system therefore has more parallel I/O engines as well as much higher raw link rates. It is architecturally misleading to compare the AGC's direct control interfaces with a phone's external USB connector and conclude that the phone "has less I/O".

6.3 Determinism versus throughput

Property	AGC	Modern consumer phone
Interrupt timing	Designed around bounded, known event handling	Complex SoC and OS; general-purpose latency not usually safety-certified
Sensor/actuator electrical interfaces	Purpose-built for Apollo G&N and control hardware	Purpose-built for consumer sensors/radios; spacecraft interfaces require adapters
Aggregate bandwidth	Low by modern standards	Very high; radios/cameras/storage operate at MB/s to Gbit/s scales
Parallel offload	Limited; central CPU services important I/O	Extensive: modem, ISP, GPU, NPU, sensing hub, DMA
Qualification	Apollo flight hardware and software assurance	Consumer qualification; not a lunar-flight safety case

7 APOLLO 11 CASE STUDY: 1201 AND 1202

7.1 What actually happened

During the Apollo 11 lunar descent, five executive-overflow programme alarms occurred before the low-gate phase. NASA's mission report states that guidance and control performance was not affected. The principal cause was a configuration interaction between the rendezvous radar, its coupling data units and two 800 Hz reference/excitation sources that were frequency locked but not phase locked. The coupling data units interpreted the resulting difference as continual angular movement and transmitted unwanted incremental pulses to the guidance computer [3].

The two radar coupling data units could together generate a maximum of 12.8 thousand pulses per second. Each pulse required one 11.72 microsecond memory cycle. Multiplying the values gives approximately 0.150 seconds of processor occupation per second, or 15% of total computer time. NASA further states that the computer normally operated at approximately 90% of capacity during peak powered-descent activity [3]. The combination could therefore demand more than 100% of available processor time.

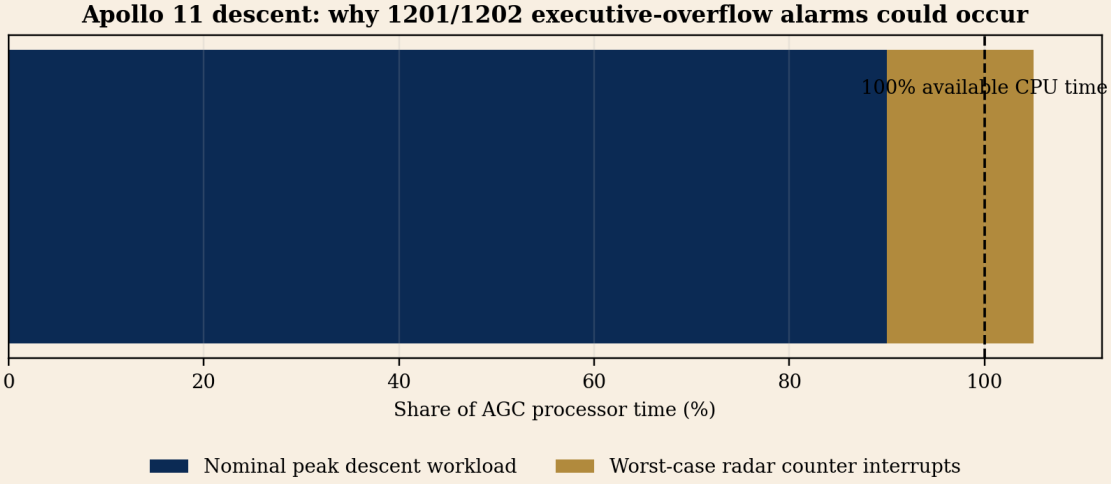


Figure 7.1 - Reconstructed CPU-time budget from the Apollo 11 Mission Report. The values are not estimates invented for this thesis: NASA reports ~90% nominal peak use and up to ~15% additional radar-interrupt load.

7.2 Why the spacecraft did not crash

The executive-overflow alarms are often described simplistically as "the computer overloaded". That is true but incomplete. The important engineering achievement was that overload did not imply arbitrary failure. The priority executive recognised that the requested workload could not be completed, initiated controlled software restarts and preserved high-priority guidance work. Lower-priority activities could be abandoned and rescheduled. Fred Martin's later account emphasises that this priority-driven design was intentional: the system continued the most important jobs instead of allowing every scheduled task to contend equally [4].

The episode is therefore strong evidence for the quality of the AGC's real-time software architecture. It is weak evidence for a claim of superior raw I/O capacity. Indeed, the fact that 12,800 small counter events per second could consume 15% of the entire computer is a quantitative demonstration of limited I/O headroom by modern standards. Both observations can be true simultaneously.

7.3 The lesson for modern systems

The Apollo lesson is architectural: safety-critical systems should define criticality, priorities and overload behaviour before overload occurs. A modern phone has far more spare throughput, but its general-purpose operating system is not necessarily configured to guarantee that a flight-control task will pre-empt every less important activity within a certified deadline. Modern spacecraft computers solve this through real-time operating systems, partitioning, watchdogs, redundancy and purpose-built I/O - not by returning to 1960s compute capacity.

8 THE WIDER APOLLO COMPUTING COMPLEX

8.1 The phrase entire Apollo Program

The claim that a phone has more compute than the entire Apollo Program is rhetorically attractive and technically underspecified. Apollo was not a single synchronous computer. It was a programme lasting years, involving flight hardware, ground systems, tracking stations, software-development computers, simulators, checkout equipment and contractor facilities. One must specify whether "entire" means the machines active during one mission, all machines owned by NASA and contractors at a point in time, or the cumulative computation performed over the programme's lifetime.

8.2 Houston Real Time Computer Complex

The central Mission Control comparison is much easier to bound. NASA and IBM sources state that the Apollo Real Time Computer Complex in Houston comprised five IBM System/360 Model 75 machines [5,6]. NASA documentation describing the Model 75 gives an average execution time of approximately one microsecond per instruction, corresponding to the order of one million instructions per second per processor for typical work [7]. A five-machine instantaneous envelope is therefore on the order of several million instructions per second, before considering I/O channels and peripheral systems.

That was formidable computing power in the late 1960s and supported real-time trajectory, telemetry, display and mission-control functions. Yet even a conservative modern mobile CPU can offer billions of core cycles per second across multiple cores, before GPU/NPU acceleration. It is therefore entirely plausible that a single current smartphone exceeds the central Houston mission-control CPU complex in raw instruction throughput by thousands of times. This narrower claim is much easier to defend than "the entire Apollo Program".

8.3 Goddard and the tracking network

IBM's Apollo history notes that the worldwide network used IBM 7094 systems and that data were transmitted back to two System/360 Model 75 machines at Goddard [5]. Remote sites also had local data-processing and command/telemetry equipment. NASA's historical study of mission-control computing describes the evolution of redundant ground computing from Mercury and Gemini into Apollo [13]. The ground estate was therefore large, geographically distributed and functionally diverse.

8.4 Saturn V onboard computing

During launch, guidance was normally performed not by the Command Module AGC but by the Saturn V Instrument Unit's Launch Vehicle Digital Computer (LVDC) and Launch Vehicle Data Adapter [2,8]. NASA documentation records a 512 kHz serial computer I/O rate in the adapter and a mixture of discrete, analog/digital and resolver interfaces, with reliability measures including triple-modular-redundant logic and duplex circuits [8]. The existence of the LVDC is another reason why "the Apollo computer" should not be treated as synonymous with the AGC.

8.5 Can the whole mission estate be beaten by one phone?

Probably, if the metric is instantaneous raw arithmetic throughput and the inventory is restricted to machines operationally involved in a single Apollo mission. But “probably” is not the same as a demonstrated theorem. A rigorous proof would require a complete contemporaneous inventory of active processors, a common workload or instruction-normalisation method, and a decision about whether specialised hardware counts as compute. The available historical sources are excellent for describing major systems but do not provide a single audited total suitable for one-number comparison.

Cumulative computation makes the statement plainly meaningless. A programme that ran for years can accumulate more total operations than any single phone performs during one second, simply because time is part of the integral. Conversely, instantaneous peak throughput heavily favours modern silicon. The Musk-style formulation is therefore best treated as an illustrative observation about technological progress, not as a precise engineering claim.

9 DISCUSSION

9.1 Why “false equivalence” is too strong

A comparison is not a false equivalence merely because the objects have different purposes. Engineers routinely compare unlike systems on a defined metric: mass, power, storage, peak throughput, drag coefficient, specific fuel consumption or reliability. A modern car can have more horsepower than an early aircraft even though the car cannot fly. Likewise, a phone can have more memory and raw compute than an AGC even though the phone cannot be wired directly into an Apollo Lunar Module and expected to land it.

The resource statement becomes a false equivalence only if it is used to infer mission equivalence: “because the phone has more raw compute, the Apollo flight-computer problem was trivial” or “an unmodified phone would be a better flight computer”. Neither inference follows. The AGC’s software, sensors, actuator interfaces, system identification, guidance laws and assurance process were the difficult engineering system. Raw arithmetic was one enabling resource inside that system.

9.2 Dave’s strongest point

The strongest interpretation of the rebuttal is that system capability is not reducible to CPU speed. On that point it is correct. The AGC’s purpose-built interfaces and deterministic executive made it effective at a spacecraft-control problem despite tiny resources. Its design embodies principles still relevant to real-time engineering: priority by criticality, bounded interrupt handling, simple data representation, explicit failure modes and deep integration between hardware, software and vehicle dynamics.

9.3 Dave’s weakest point

The weakest part is the phrase “far more I/O processing ability than a phone”. Modern phone SoCs are I/O-intensive machines with multi-gigabit radios, high-resolution multi-camera pipelines, high-speed storage and multiple always-on sensor domains. Their I/O engines are massively more capable in aggregate bandwidth. The AGC’s advantage is appropriate deterministic control I/O, not total I/O processing ability. The Apollo 11 radar interrupt numbers provide direct historical evidence that the AGC had limited spare I/O capacity.

9.4 Why the AGC is more impressive when described accurately

There is no need to mythologise the AGC as secretly more powerful than modern electronics. The engineering achievement is more remarkable when its limitations are stated plainly. Engineers landed humans on another world with roughly four physical kilobytes of erasable memory, tens of kilobytes of fixed programme storage and a processor whose basic operations took microseconds. They succeeded because the problem was decomposed carefully, priorities were explicit, algorithms were tailored to the machine and the overall system was rigorously verified.

10 CONCLUSIONS

10.1 Findings

- Raw resources: A current flagship smartphone has vastly more memory, storage and processor capability than a Block II AGC. The memory and storage gaps are measured in millions of times.
- I/O: The AGC does not have greater aggregate I/O processing capacity than a modern phone. It has specialised, deterministic, mission-specific I/O that is better suited to Apollo control tasks.
- Apollo 11 alarms: The 1201/1202 episode demonstrates both the AGC's limited throughput headroom and the excellence of its priority-driven overload behaviour.
- Mission suitability: A consumer phone is not a drop-in flight computer. Spacecraft control requires qualified hardware, electrical interfaces, deterministic software, fault management and system assurance.
- The entire Apollo programme: A phone probably exceeds the instantaneous raw CPU throughput of the principal operational Apollo mission computers, but the phrase "entire Apollo Program" is undefined and cannot be proven rigorously without an inventory and common metric.

10.2 Final answer to the disputed claim

Is the rebuttal right? Partly. It is right that raw processor comparisons do not establish equivalence as spacecraft-control systems. It is wrong to use that observation to reject the narrower claim about raw compute, memory and storage, and it is wrong if "far more I/O processing ability" is meant as an aggregate throughput statement. The technically precise formulation is: the AGC was vastly less powerful than a modern phone in raw computational resources, but far better adapted and assured for the specific deterministic task of controlling an Apollo spacecraft.

Further work is recommended, subject to the return of the departmental oscilloscope last seen in the Senior Common Room and the restoration of the tea urn to operational status.

REFERENCES

[1] NASA, “Characteristics of the Apollo Guidance Computer,” Apollo Systems Description / logical design, NASA Technical Reports Server, 1966. Fixed memory 36,864 words; erasable memory 2,048 words.
https://ntrs.nasa.gov/api/citations/19660019469/downloads/19660019469.pdf

[2] NASA / Grumman, Apollo News Reference - Lunar Module Guidance, Navigation and Control. Guidance computer dimensions, 70 lb weight, 2.048 MHz clock, 12 us memory cycle, 24 us add time.
https://www.nasa.gov/wp-content/uploads/static/history/alsj/LM08_Guidance-Navigation-Control_ppGN1-48.pdf

[3] NASA, Apollo 11 Mission Report, November 1969, section 16.2.5. Executive overflow alarms, 11.72 us counter-interrupt service time, 12.8 kpps maximum unwanted radar pulses, ~15% CPU load and ~90% nominal peak descent capacity.
https://www.nasa.gov/wp-content/uploads/static/apollo50th/pdf/A11_MissionReport.pdf

[4] F. Martin, “Apollo 11 Lunar Surface Journal: Program Alarms,” NASA History. Priority-driven executive, 2K erasable / 36K fixed memory, 11.7 us cycle and 23.4 us ADD.
https://www.nasa.gov/wp-content/uploads/static/history/alsj/a11/a11.1201-fm.html

[5] IBM, “Apollo,” IBM History. Five System/360 Model 75 machines in the Houston RTCC; IBM 7094 systems in the tracking network and two Model 75 systems at Goddard.
https://www.ibm.com/history/apollo

[6] NASA, Around the World / NASA management history (SP-4537), account of procurement of five IBM 360-75 systems for lunar Mission Control operations.
https://www.nasa.gov/wp-content/uploads/2023/04/sp-4537.pdf

[7] NASA, technical description of the IBM System/360 Model 75 used in real-time facilities: approximately 195 ns internal cycle; average instruction execution about 1 us; 1,048,576-byte main storage in cited configuration.
https://ntrs.nasa.gov/api/citations/19700024253/downloads/19700024253.pdf

[8] NASA, Description and Performance of the Saturn Launch Vehicle Navigation, Guidance and Control System, 1970. LVDC/LVDA functions and 512 kHz serial I/O rate.
https://ntrs.nasa.gov/api/citations/19700023342/downloads/19700023342.pdf

[9] NASA, aerospace report on Apollo/Skylab RTCC support, documenting five IBM 360-75 computers in the Apollo RTCC and four primary plus one reserve.
https://ntrs.nasa.gov/api/citations/19740024140/downloads/19740024140.pdf

[10] Qualcomm, Snapdragon 8 Elite Mobile Platform Product Brief. Oryon CPU up to 4.32/4.47 GHz depending variant, LPDDR5x, UFS 4.0, triple ISP, sensing hub and other SoC subsystems.
https://docs.qualcomm.com/bundle/publicresource/87-83196-1_REV_A_Snapdragon_8_Elite_Mobile_Platform_Product_Brief.pdf

[11] Qualcomm, Snapdragon 8 Elite Platform Product Brief. Peak 5G downlink up to 10 Gbit/s and Wi-Fi 7 up to 5.8 Gbit/s; integrated heterogeneous I/O subsystems.
https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/Snapdragon-8-Elite-Platform-Product-Brief.pdf

[12] Samsung, “Galaxy S25 Ultra Specifications,” Samsung UK. 12 GB memory and storage options to 1 TB.
https://www.samsung.com/uk/smartphones/galaxy-s25-ultra/specs/

[13] J. E. Tomayko, Computers in Spaceflight: The NASA Experience, NASA Contractor Report / NASA History, including chapters on Apollo onboard computing and computers in Mission Control.
https://ntrs.nasa.gov/api/citations/19880069935/downloads/19880069935_Optimized.pdf

[14] E. C. Hall, Reliability History of the Apollo Guidance Computer, NASA-CR-140340 / R-713, 1972.
https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19750004273.pdf

[15] NASA, Apollo Lunar Landing Experience Report, guidance-computer summary: approximately 70 lb and 55 W flight hardware, core-rope programme storage and DSKY crew interface.
https://www.nasa.gov/wp-content/uploads/2023/06/eppler-slides-apollo-lunar-landing-experience-report-20070-r4.pdf

APPENDIX A WORKED CALCULATIONS

A.1 AGC physical memory capacity

Quantity	Calculation	Result
Erasable physical capacity	2,048 words x 16 bits / 8	4,096 bytes = 4 KiB
Fixed physical capacity	36,864 words x 16 bits / 8	73,728 bytes = 72 KiB
12 GB phone RAM / AGC erasable	12,000,000,000 / 4,096	2,929,687.5 x
1 TB phone storage / AGC fixed	1,000,000,000,000 / 73,728	13,563,368.1 x
Clock-frequency ratio	4.32 GHz / 2.048 MHz	2,109.4 x
AGC maximum memory cycles	1 / 11.72 us	~85,324 cycles/s
AGC ADD throughput if continuous	1 / 23.44 us	~42,662 ADD/s
Worst radar interrupt load	12,800/s x 11.72 us	0.1500 s/s = 15.0% CPU

The memory calculations use physical 16-bit word capacity to avoid hiding the parity-bit convention. If only information bits are counted, both AGC capacities reduce modestly, which makes the modern/AGC ratios larger. Decimal GB/TB are used for Samsung's marketed capacities; binary GiB/TiB conventions alter the ratios by only a few percent and do not affect the conclusion.

A.2 Apollo 11 radar-interrupt arithmetic

NASA reports a maximum of 6.4 thousand pulses per second from each of two rendezvous-radar coupling data units. With both active, the maximum is 12.8 kpps. Each pulse occupies one 11.72 microsecond memory cycle:

$$12{,}800 \times 11.72 \times 10^{-6} = 0.150016 \text{ seconds of processor time per second} = 15.0016\%.$$

At an already reported peak workload of approximately 90%, the combined demand can reach roughly 105%. The actual alarm sequence depends on the timing of scheduled jobs and interrupts, but the arithmetic explains why executive-overflow conditions were possible.

APPENDIX B CLAIM-BY-CLAIM AUDIT

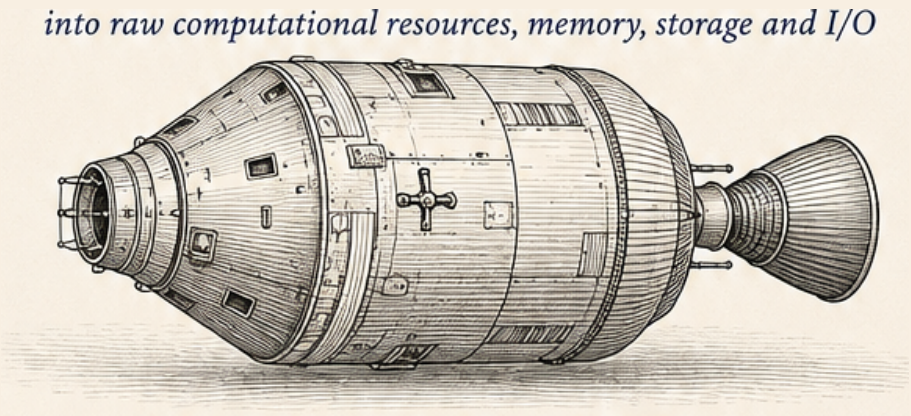
Claim	Assessment	Reason
“The phone in your pocket has vastly more raw computing power, memory, and storage than the Apollo Guidance Computer.”	Supported	Modern phone SoCs are many orders of magnitude larger; processor timing and architecture are many orders
“Your phone has more compute than the entire Apollo program.”	Plausible but undefined	A modern phone clearly exceeds one AGC and likely exceeds the major active mission CPU estate in
“This is a false equivalency.”	Not as applied to the narrow resource claim	Comparing unlike systems on a defined metric is valid. It becomes misleading only if raw resource su
“The AGC had far more I/O processing ability than a phone.”	Not supported as aggregate throughput	Modern phone SoCs process far more data and contain many dedicated I/O engines. The AGC had
“Computing primes? Sure. But they didn’t do that.”	Rhetorically fair but technically incomplete	The phone can execute both general and specialised workloads at much higher raw rates, but spacecr

The audit deliberately separates statements of fact from implications. This avoids the common failure mode in technology comparisons where a correct numerical observation is rejected because an unrelated conclusion would be wrong.

APPENDIX C ENGINEERING PERSPECTIVE

The AGC is historically significant not because it defeats modern computers on hidden metrics, but because it represents an unusually coherent union of hardware, software and vehicle requirements. Its designers chose an architecture that could be manufactured reliably, operated within strict mass and power budgets, interfaced directly with the spacecraft, survived expected faults and supported mission software whose complexity expanded dramatically as Apollo matured.

Modern engineers inherit semiconductor capabilities that remove many of Apollo's resource constraints, but they also inherit software stacks of vastly greater complexity. The enduring lesson is therefore not "old computers were secretly better". It is that predictable behaviour under stress comes from architecture, priorities, verification and system understanding. The 1201/1202 alarms remain a particularly elegant demonstration: the computer was overloaded, announced that fact, shed lower-priority work and continued the functions that mattered.



"Limited resources are not the same thing as limited engineering."

Department of Aeronautical Engineering and Applied Mathematics
King Ethelred's College of Engineering
Upper Piddlington