Unit 21

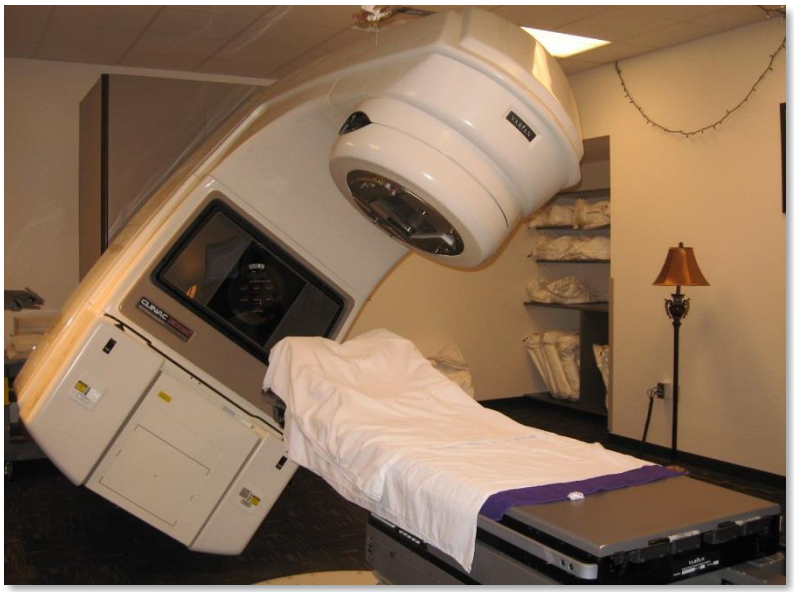
Embedded Failures

When Embedded Systems Attack…

- Embedded systems can fail for a variety of reasons
  - Electrical problems
  - Mechanical problems
  - Errors in the programming
  - Incorrectly specified
  - Errors caused by users
  - Zillion other reasons
- Some failures have been well documented and can be used to learn how to make systems better.

Therac-25

- The Therac-25 was a medical radiation therapy machine developed in Canada in the mid-1980s.
- Controlled by a PDP-11 (16-bit minicomputer)
- Errors in the hardware/software design led to three patients being killed and many injured.



Therac-25

- Examination of the system revealed numerous defects that could lead to improper operation:
  - Insufficient hardware/software interlocks to prevent dangerous types of actions.
  - Certain unusual patterns of keystrokes could put the system in the incorrect mode.
  - Software was reused from previous models despite changes in the overall design.
  - No way for software to tell if the hardware was doing what it was told to do (open loop control).
  - Control tasks and operator tasks were not synchronized leading to possible race condition.
  - Overflows in some variables were not detected.

Ariane 5

The European Space Agency's Ariane rockets were designed in the 1970's and the first generation Ariane 1 launched in 1979.

Later generations were developed and the first launch of an Ariane 5 rocket on June 4th, 1996 failed due to errors in the onboard software and with the design process.

Click to watch video.

Ariane 5

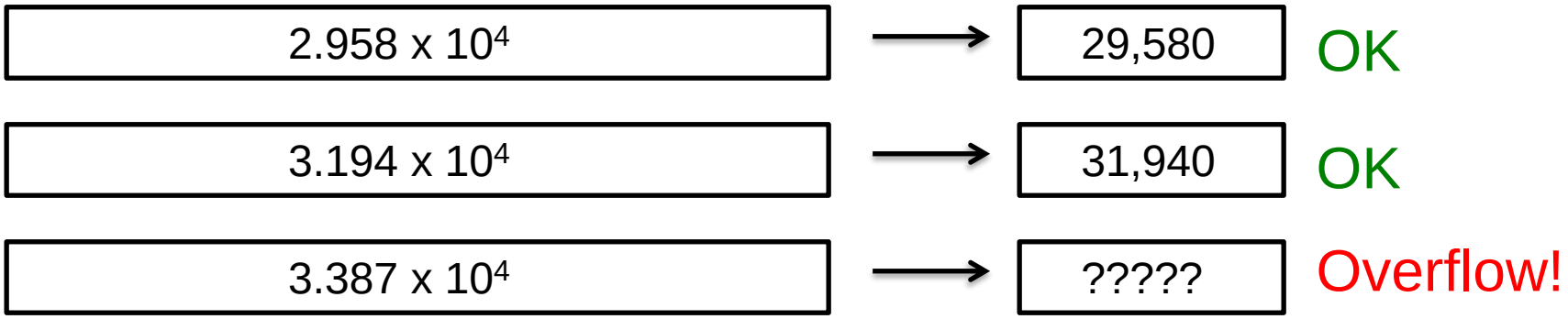
What Went Wrong?

- The Ariane 5 guidance system was from the older Ariane 4.
- The guidance system represented horizontal velocity by a 64-bit floating point number.
- As part of the guidance operations, the 64-bit number was converted to a 16-bit signed fixed point number.
- The newer rocket was faster, used a different launch trajectory, and could obtain higher velocities during launch. The 64-bit values exceeded those seen with the Ariane 4.

Ariane 5

What Went Wrong?

- As velocity increased, the floating point values exceeded the maximum value that could be represented with a 16-bit fixed-point number.
- The conversion to a 16-bit signed number resulted in an **overflow** and the processor executed a hardware **exception**.



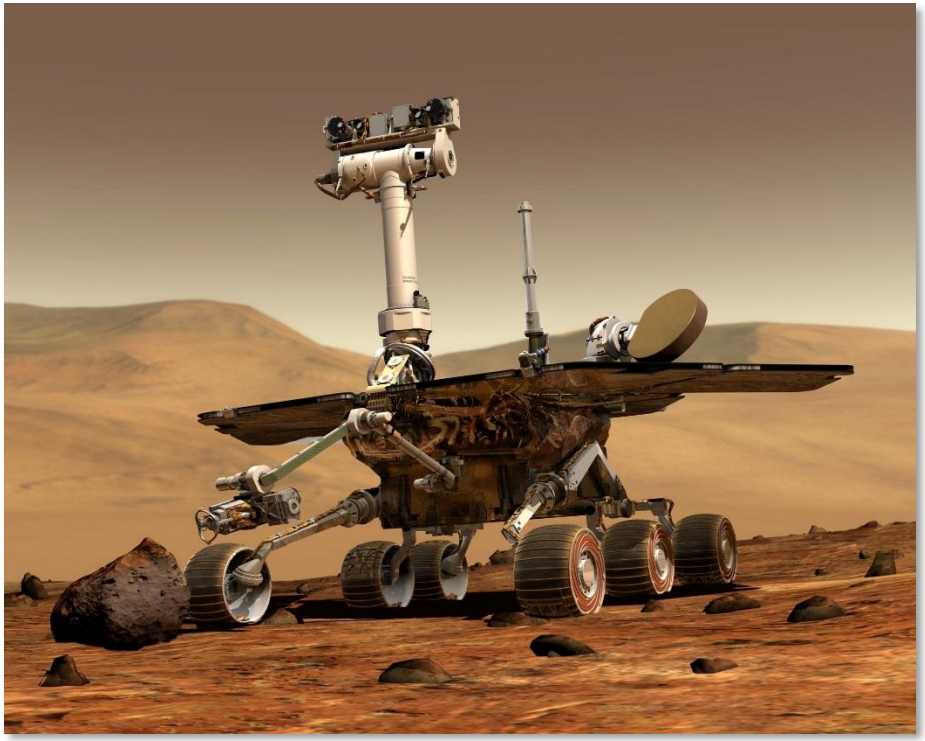
Ariane 5

Not Just a Software Problem

- Reviews of the design prior to launch did not address limitations on the guidance data.
- Checking of variables to see if their values were within acceptable bounds was turned off in the software.
- The guidance system was never tested using simulated Ariane 5 flight conditions.
- Simulated data, rather than real guidance output, was used in systems tests.
- When tests were done later using the actual flight conditions, the simulations failed in exactly the same way.

Mars "Spirit" Rover

- NASA/JPL robotic rover sent to Mars in 2004.
- Suffered a severe "anomaly" upon landing that nearly aborted the mission.



Mars "Spirit" Rover

- Spirit appeared to be working as expected after landing, but soon started having problems.
- JPL could contact it to give it commands and know that it was alive but very little data was being received.
- Eventually concluded that the rover was resetting continuously due to problems with the software stored in FLASH memory.
- Spirit was commanded to run in "crippled" mode where it doesn't use the FLASH data.
- JPL had control of it, sort of, but what was wrong?

Mars "Spirit" Rover

- For 11 Martian days, the JPL team worked to diagnose and fix the problem.
- Data in the FLASH memory was believed to be corrupted.
- Eventually reformatted the FLASH and loaded new data.
- Problem caused by way the OS used memory to implement a file system in the FLASH.
- Processes could run out of available memory and get stuck causing a reset.
- Eventually fixed and returned to full operation.

Toyota Unintended Acceleration

- Over the last several years many claims that some Toyota vehicles were subject to sudden unintended acceleration problems.
- Vehicle throttles use “drive-by-wire” system
  - No mechanical connection between the throttle pedal and the engine.
  - Computers sense the position of the throttle and adjust the engine power accordingly.
  - Similar to “fly-by-wire” system in use in current military and commercial aircraft and in the space shuttle.

Toyota Unintended Acceleration

- Toyota and NHTSA claimed the problem was with floor mats or drivers pressing the throttle instead of the brake.
- Eventually resulted in numerous lawsuits
- Testimony by expert witnesses for the plaintiffs have pointed to numerous potential problems in the embedded systems running the vehicles.
  - Disclaimer: Testimony is not proof, just an opinion.

Toyota Unintended Acceleration

- Some possible problems were identified during litigation:
  - Possible for a single bit flipped to cause the problem.
  - Portions of the memory were not protected against corruption due to stack overflows and software bugs.
  - One task was handling numerous functions including fail-safes and brake override.
  - Tasks could terminate without the OS noticing.
- Vehicle software is not designed to the same standards as required by law in aircraft, medical devices, etc.

Toyota Unintended Acceleration

- Do we have unreasonably high expectation for the reliability of consumer electronic devices?
- How much are people willing to pay for reliability?

  "Fly by wire is done on aircraft, and if you have flown on a 757,767,747-400,787,777, or any Airbus Airliner, you have depended on this technology from take-off to landing. The best of these systems are Quadruple Redundant, typically three redundant actuators and dual sticks, plus redundant trim switch controls -- plus a dissimilar backup system. In these systems the power systems are triple redundant or quadruple redundant as well." - EETimes.com blogger
- How much would a car cost if you demanded the same reliability and redundancy as in an aircraft?

Air France Flight 447

- On June 1, 2009, an AirBus A330 flying from Rio de Janeiro to Paris crashed in the ocean off Brazil.
- Brazilian Navy found debris and bodies within days of the crash but it took nearly two years to find the “black boxes” and another year to determine the cause of the crash.

Air France Flight 447

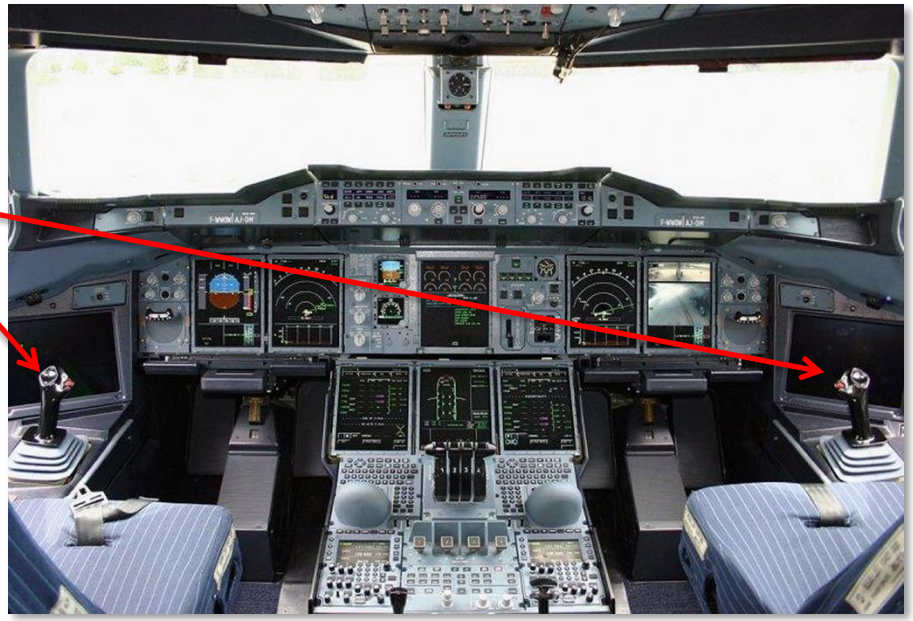
- Aircraft encountered icing conditions that caused ice crystals to clog the sensors that measure air speed.
- With no air speed data, the autopilot disengaged and pilots took manual control of the aircraft.

Pitot tubes

- Pilots were not experienced in flying without the autopilot under these conditions and did several things wrong.

Air France Flight 447

- Plane entered a severe nose-up configuration leading to a stall condition.
- Pilots were given confusing information from the instruments and did not correct the situation.



AirBus A330 cockpit

Air France Flight 447

- Software will sound an alarm to warn pilots if plane is in a stall condition.
- However, if the plane's attitude was outside expected ranges, software assumed it had bad data and turned alarm off.
- Result:
  - Stall condition ⇒ alarm sounds
  - Really, really bad stall condition ⇒ no alarm
- Pilots did not hear an alarm, but when they brought the nose down the stall warning would come on.

Air France Flight 447

- Pilot and copilot both had control sticks.
- Software would combine the position information from both and control the plane on that basis.
- During the flight the pilot was pushing his stick forward to bring the nose down but the copilot was pulling his back.
- No mechanical connection between the two control sticks.

Boeing 787 with connected “yokes”

Boeing 737 MAX

- The 737 MAX is an update of their best-selling airliner.
- New engines used for better performance, but were larger and had to be mounted differently, and this would cause changes in the handling in the air.

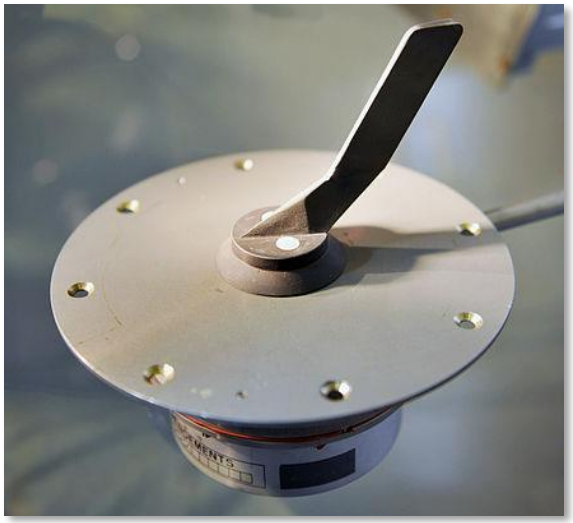
- Redesigning and recertifying the 737 was expensive so Boeing tried to compensate for the altered handling in the flight control software.

Boeing 737 MAX

- With engines more forward and higher, plane tended to nose-up at times.
- **Maneuvering Characteristics Augmentation System** (MCAS) flight control software was designed to compensate by adjusting the horizontal stabilizer.
- Two crashes in Indonesia (2018) and Ethiopia (2019) killed 346 people and were determined to be caused by MCAS.
- Grounded worldwide from March 2019 to November 2020.

Boeing 737 MAX

- Pilots were not told about the presence of MCAS, what it would do, or how to disable it.
- MCAS received input from Angle-of-Attack (AoA) sensors to determine if the nose was too high.
- If AoA sensor was defective, MCAS could bring the nose down by deflecting the horizontal stabilizer.
- Pilots could not override MCAS's actions by operating the controls



Boeing 737 MAX

- Investigations revealed numerous problems with the MCAS implementation.
- MCAS could cause overly large deflections of the horizontal stabilizer.
- Two AoA sensors on 737, but only one was used at a time, resulting in single point of failure.
- Many 737's had no AoA reading in the cockpit, and no warning of one being defective, lacking an "AoA Disagree Alert" indicator.
- FAA did not conduct a full safety analysis of adding MCAS to the aircraft.

Boeing 737 MAX

- Several changes have been made to fix the problem.
- Pilots trained about MCAS and how to disable it.
- Both AoA sensors used to activate MCAS. If sensors differ significantly, MCAS disabled
- Only one activation of MCAS for each high-AoA event.
- Lower limit on the deflection of the horizontal stabilizer by MCAS.
- Flight control computers now operate in a redundant manner where results are compared. Previously only one used at a time.