IAF0030
Arvutitehnika erikursus I

Loeng 7
Enemies of Dependability

Gert Jervan

Tallinna Tehnikaülikool
Arvutitehnika instituut

Important Dates

- ✓ Next lecture: March 19
- ✓ No lecture on March 26
- ✓ Case Study drafts: April 1st
- ✓ Case Studies discussion: April 2nd
- ✓ Presentations: April 30, May 7
- ✓ **Final report: May 18**

Lecture Outline

- ✓ **Introduction**
- ✓ **Software**
- ✓ **Hardware**
- ✓ **Humans**

Downtime

- ✓ Planned downtime
  - Maintenance, repair, upgrade
- ✓ Unplanned downtime
- ✓ Dependability:
  - Turn unplanned uptime into planned downtime
  - Reduce downtime (magic nines)

Sources of Problems

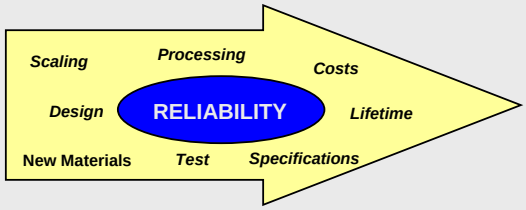
Category	Early 80s	Late 80s	90s	2000s
Hardware + environment	32%	29%	20%	?
Software	26%	58%	40%	?
Human Operators	42%	13%	40%	?

Hardware

Hardware and Environment Failures

- Moving parts, high speed, low tolerance, high complexity: disks, tape drives/libraries
- Lowest MTBF found in fans and power supplies
- Often fans fail gradually → subtle, sporadic failures in CPU, memory, backplane
- Environment: power, cooling, dehumidifying, cables, fire, collapsing racks, ventilation, earthquakes, ...

Hardware Reliability Challenges

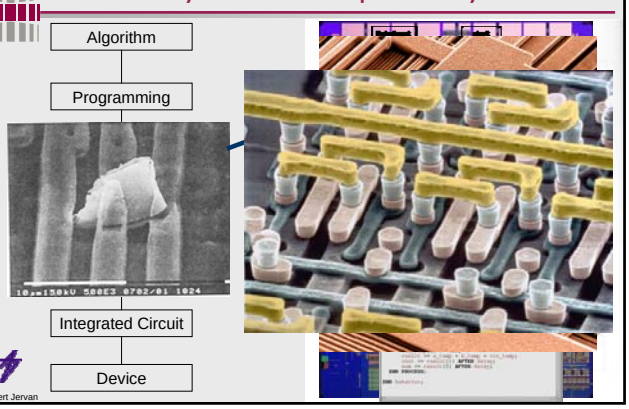


Reliability Dependencies and Impact to Cost

Hardware - Background

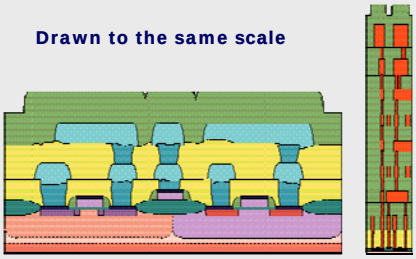
- Chip designers, device engineers and the high-reliability community recognize that reliability concerns ultimately limit the scalability of any generation of microelectronics technology
- Statistical methods and reliability physics provide the foundation for better understanding the next generation of scaled microelectronics
  - Microelectronics device physics
  - Reliability analysis and modeling
  - Experimentation
  - Accelerated testing
  - Failure analysis
- The design, fabrication and implementation of highly aggressive advanced microelectronics requires expert controls, modern reliability approaches and novel qualification strategies

A Few Layers of Computer Systems



What is Technology Scaling

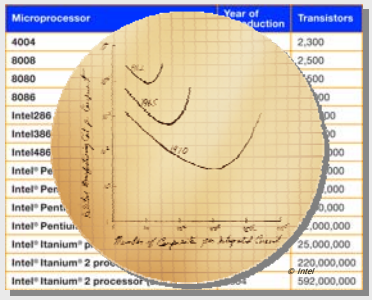
Drawn to the same scale



1.0 μm
Mid 1980s
Speed: 10 MHz

0.1 μm
Early 2000's
Speed: 3 GHz

Moore's Law

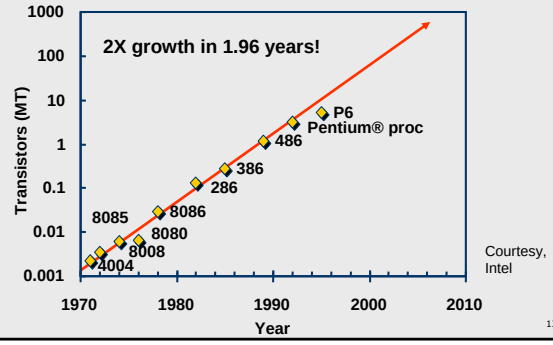


Microprocessor	Year of production	Transistors
4004		2,300
8008		2,500
8080		500
8086		00
Intel286		00
Intel386		0
Intel486		000
Intel® Pe		000
Intel® Per		,000
Intel® Pentiu		0,000
Intel® Pentiu		,000,000
Intel® Itanium® p		25,000,000
Intel® Itanium® 2 pro		220,000,000
Intel® Itanium® 2 processor		592,000,000

"...(T)he first microprocessor only had 22 hundred transistors. We are looking at something a million times that complex in the next generations—a billion transistors. What that gives us in the way of flexibility to design products is phenomenal."
Gordon E. Moore, April 19, 1965

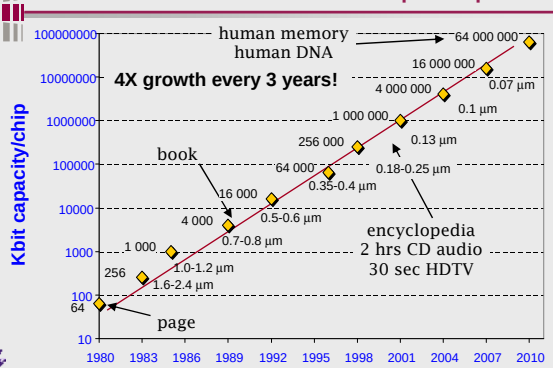
Moore's Law in Microprocessors

Transistors on lead microprocessors double every 2 years



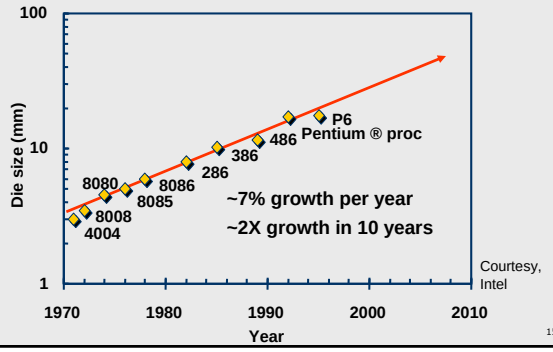
Courtesy, Intel

Evolution in DRAM Chip Capacity



Die Size Growth

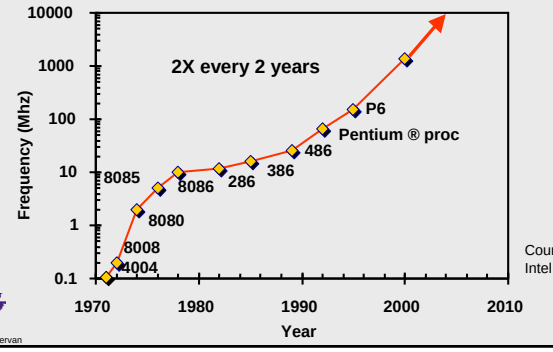
Die size grows by 14% to satisfy Moore's Law



Courtesy, Intel

Clock Frequency

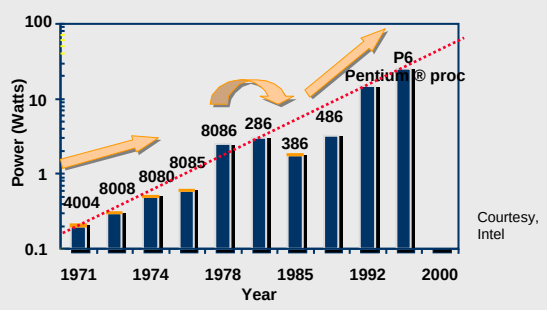
Lead microprocessors frequency doubles every 2 years



Courtesy, Intel

Power Dissipation

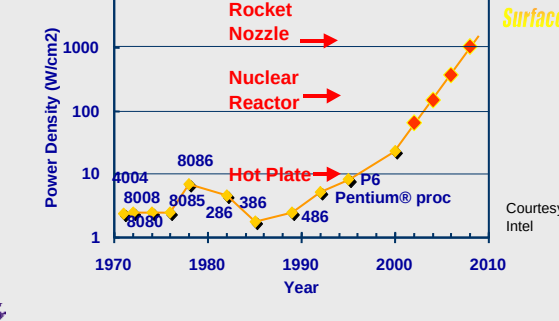
Lead Microprocessors power continues to increase



Courtesy, Intel

Power delivery and dissipation will be prohibitive

Power Density



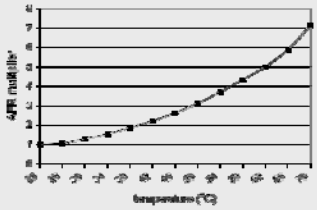
Courtesy, Intel

Power density too high to keep junctions at low temp

Hot Chips

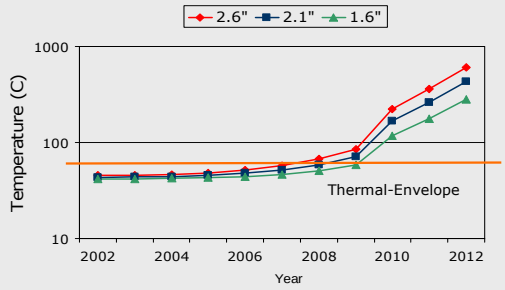
Temperature Affects Disk Drive Reliability

- Heat-Related Problems
  - Data corruption
  - Higher off-track errors
  - Head-crashes
- Disk drive design constrained by the thermal-envelope
  - Puts a limit on drive performance



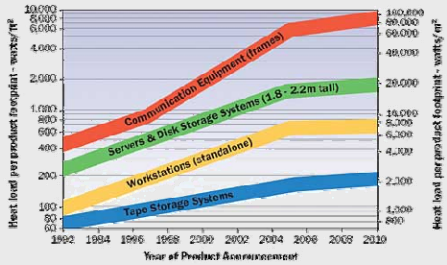
Source: D. Anderson et al, "More than an Interface – SCSI vs. ATA", FAST 2003.

Drive Temperature

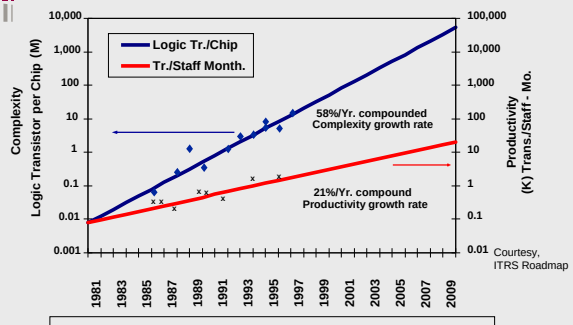


40% annual growth in the data-rate

Heat Density



Design Productivity Trends



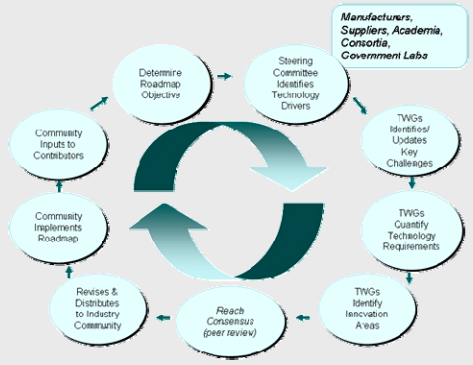
Courtesy, ITRS Roadmap

Complexity outpaces design productivity

ITRS Roadmap

- ITRS predicts the main trends in the semiconductor industry spanning across 15 years into the future.
- The International Technology Roadmap for Semiconductors is sponsored by the five leading chip manufacturing regions in the world: Europe, Japan, Korea, Taiwan, and the United States.
- The objective of the ITRS is to ensure cost-effective advancements in the performance of the integrated circuit and the products that employ such devices, thereby continuing the health and success of this industry.

ITRS Roadmap



ITRS Roadmap

- www.itrs.net
- Editions:
  - 1994, 1997, 1999, 2001, 2003, 2005
  - Previously: SIA Roadmap

Technology Directions: ITRS Roadmap

Year	1999	2002	2005	2008	2011	2014
Feature size (nm)	180	130	100	70	50	35
Mtrans/cm^2	7	14-26	47	115	284	701
Chip size (mm^2)	170	170-214	235	269	308	354
Signal pins/chip	768	1024	1024	1280	1408	1472
Clock rate (MHz)	600	800	1100	1400	1800	2200
Wiring levels	6-7	7-8	8-9	9	9-10	10
Power supply (V)	1.8	1.5	1.2	0.9	0.6	0.6
High-perf power (W)	90	130	160	170	174	183
Battery power (W)	1.4	2.0	2.4	2.0	2.2	2.4

For Cost-Performance MPU
(L1 on-chip SRAM cache; 32KB/1999 doubling every two years)

http://www.itrs.net/

Industry Scaling Trends & Reliability Considerations

- Reduced gate oxide thicknesses
- Increased thermal/power densities
- Reduced interconnect dimensions
- Higher device operating temperatures
- Increased sensitivity to defects and statistical process variations
- Introduction of new materials with each new generation, replacing proven materials
  - e.g. Cu and low K inter-level dielectrics for Al and SiO2

Industry Scaling Trends & Reliability Considerations

- Dramatic increase in processing steps with each new generation
  - approx. 50 more steps per generation and a new metal level every 2 generations
- Rush to market - Less time to characterize new materials than in the past
  - e.g. reliability issues with new materials not fully understood and potential new failure modes
- Manufacturers' trends to provide 'just enough' lifetime, reliability, and environmental specs for commercial & industrial applications
  - e.g. 3-5 yr product lifetimes, trading off 'excess' reliability margins for performance

Industry Scaling Trends & Reliability Considerations

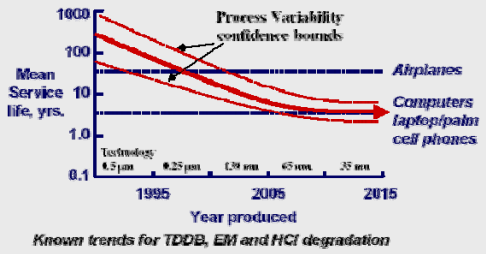
- Significant rise in the amount of proprietary technology and data developed by manufacturers, reluctance to share information with hi-rel customers
  - e.g. process recipes, process controls, process flows, design margins, MTTF
- Next generation microelectronics focus on the performance needs of the commercial customer, with little or no emphasis on the needs of the space customer
  - e.g. extended life, extreme environments, high reliability
- Increasingly difficult testability challenges due to device complexity

Product Technical Trends

	1990	2000	2010
Operating temperature, °C	-55 to 125	-40 to +85	0 to 70
Supply voltage	5v	1.5v	0.6v
Max. power (high perf.)	5	100	170
No. of package types	<10	<80	??
Design support life	>10 yrs.	1-5 yrs.	<1yr.
Production life	>10 yrs.	3-5 yrs.	<3yrs.
Service life	>20 yrs.	5-10 yrs.	<5yrs.

*MRQW-2002, Bernstein

Commercial Chip Reliability Estimation



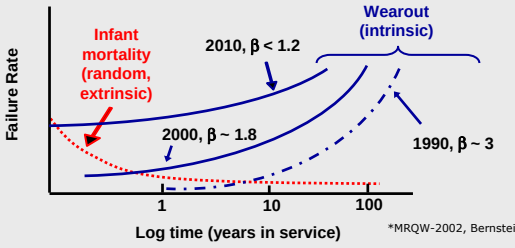
*Extrapolated from ITRS roadmap, MRQW-2002, Bernstein

Impact of scaling on wear-out failure mechanisms

- Dominant Failure Mechanisms
  - Electromigration (EM)
    - Migration of atoms in a conductor
  - Hot Carrier Injection (HCI)
    - High energy carriers degrade oxide
  - Negative Bias Temperature Instability (NBTI)
  - Time-Dependent-Dielectric-Breakdown (TDDB)
    - Oxide breakdown: Formation of a conduction path through gate oxide

Device Reliability Trends

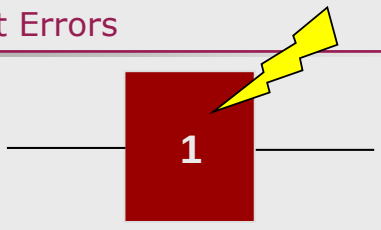
As technology progresses, wearout failures become statistically indistinguishable from infant mortality failures with the same wearout drivers.



*MRQW-2002, Bernstein

Soft Errors

Soft Errors



- Transient bit-flip (soft memory error)
  - Random event
  - Corrupts the value but not the cell
  - Can be corrected (in contrast to hard errors caused by faults in the hardware itself)
  - Happen continuously during system lifetime (*i.e.*, can not be screened by burn-in tests)

Sources

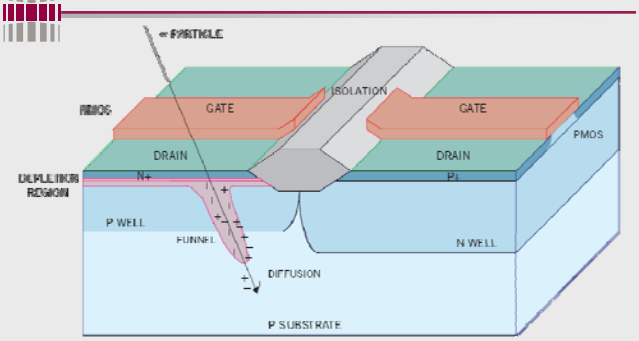
- First traced to alpha particlce emissions from chip packaging materials
  - Most sources removed (pure materials, different designs, shielding)
- Today's main problem: cosmic radiation
  - Cosmic particles from deep space (actually 5th- or 6th-hand collision particles)
    - At ground level ca 95% neutrons, 5% protons
  - Radioactive material in manufacturing process

Sources (cont.)

- Four main sources:
  - Low-energy alpha particles
  - High-energy cosmic particles
  - Thermal neutrons
  - Poor system design

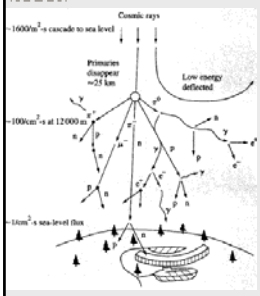
SER type	Source	Mechanism	Trend
Alpha	Thorium and uranium contamination in-mold compound, silicon, or lead bumps	2- to 9-MeV alpha particle creating electron-hole funnel traveling 25 microns in silicon	Exponential increase with scaling
Cosmic	Intergalactic sources modulated by solar flares	High-energy neutrons/protons (10 MeV to 1 GeV) colliding with silicon nuclei	Decrease in failures in time per megabit
Thermal neutron	Boron present in BPSG25-meV neutrons	Collision with B10 in BPSG	Highest, always dominates if present

Soft Errors



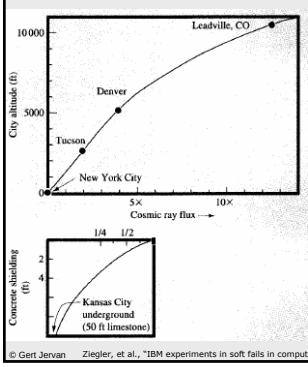
The electric field in the depletion region directly generates electron-hole pairs in its wake, causing the charges to drift so that the transistor sees a current disturbance

Source and Characteristics (cont)



The figure shows a schematic view of how cosmic rays cascade through the earth's atmosphere. The high-energy particle flux which hits the earth's outer atmosphere contains about 1000 particles/m^2-s, mostly protons with energies far above 1 GeV. As the particles hit atmospheric atoms, they shatter them, causing a cascade which increases to a particle flux of 1,000,000/m^2-s at airplane altitudes (12,000 m). The lower atmosphere is so dense that much of the flux is absorbed by sea level, where the flux is only ten times higher than the incident flux. The cascades contain a zoo of particles, of which only the neutrons and pions can cause significant LSI fails

Impact of Elevation

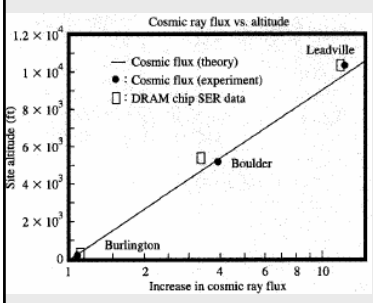


This plot shows how the soft-error rate of an LSI chip changes with altitude. Shown are the altitudes of New York City (sea level), Tucson (2,390 ft), Denver (5,280 ft), and Leadville, CO (10,152 ft).

Also shown is the estimated reduction of sea-level fails if concrete shielding is introduced. The point marked "Kansas City underground' assumes about 5000 g/cm2 of limestone, which should totally block out all cosmic rays so that there should be zero fails. This calculation assumes that the only important particles for SER effects are protons, neutrons, and pions. At sea level, the flux is >96% neutron, and these determine the soft-error rate. Above sea level, the percentage of protons and pions increases rapidly until at 10,000 ft altitude, they account for about 35% of the fails.

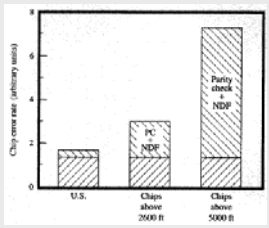
Ziegler, et al., "IBM experiments in soft fails in computer electronics (1978 - 1994)," IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

The Altitude Factor



Summary of data for field test of DRAM chips. The plot shows the theoretical prediction for the cosmic ray flux change with altitude (solid line), the measured cosmic ray flux (dots), and the change in fail rate for a 288Kb DRAM chip. The experiment included a total of 71M bits. This result was the first life test of an IBM chip and conclusively showed the dramatic effect of altitude on the fail rate of a chip.

Source and Characteristics (cont)



Altitude effects on repairs of memory modules (1984). The figure shows the data extracted from repair records for memory modules in 1984. The modules have been divided into three groups depending on their altitude, with the leftmost group showing the average for all U.S. modules, the center section for those which came from sites above 2600 ft, and the rightmost section for those from sites above 5000 ft. The lower hatched section in each bar indicates the number of normal hard fails (some memory bit had permanently failed). The upper hatched section shows the number of modules with no electronic defect (called an NDF, for no defect found). For the United States as a whole (mean altitude 770 ft), this NDF result accounted for less than 10% of the modules, but in the mountain states (mean altitude 3200 ft) it was five times this level, and accounted for about 50% of the modules. For the modules installed in Denver, CO (altitude 5280 ft), the NDF rate was ten times the rate for the country as a whole. Data from W. S. Graff, IBM Data Systems Division, internal report, 1985.

Evidence of Cosmic Ray Strikes

- Documented strikes in large servers found in error logs
  - Normand, "Single Event Upset at Ground Level," IEEE Transactions on Nuclear Science, Vol. 43, No. 6, December 1996.
- Sun Microsystems, 2000 (R. Baumann, Workshop talk)
  - Cosmic ray strikes on L2 cache with defective error protection
    - caused Sun's flagship servers to suddenly and mysteriously crash!
  - Companies affected
    - Baby Bell (Atlanta), America Online, Ebay, & dozens of other corporations
    - Verisign moved to IBM Unix servers (for the most part)

Current Situation

- Soft errors induced the highest failure rate of all other reliability mechanisms combined

Rober Baumann, TI

Measuring

- The rate at which SEUs (single-event-upsets) occure is given as SER, measured in FITs (failures in time)
  - 1 FIT = 1 failure in 1 billion device-operation hours
  - 1000 FIT ≈ MTTF 114 years

Examples

- A Cell Phone with 4 Mbit memory, SER=1000 FIT per megabit. Result: 1 soft error every 28 years
- A high-end router with 10 Gbit SRAM. SER=600 FIT per megabit. Result: 1 error every 170 hours
- Router farm with 100 Gbits: 17 hours
- Laptop in the airplane: 256 Mbit memory. SER=100,000 FIT per megabit. Result: error in every 5 hours

Recently Reported Data on Soft Error Rates

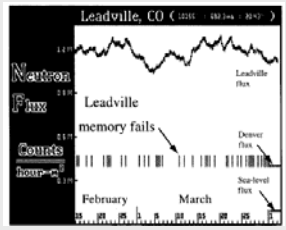
Type of Memory	Reported SER	Error per bit-hour	FIT*/Mbit
Goal for new Cypress products	200 FIT*	?	?
SRAM (quoted by vendors)	200 to 2,000 FIT	?	?
"typical"	1,000 FIT	?	?
DRAM at full speed	Few hundred to few thousand FIT	?	?
SRAMs at 0.25 micron* and below	10,000 to 100,000 FIT	?	?
Commercial CMOS* memory	>1E-5 to 1E-7 per bit-day*	>4E-7 – 4.2E-9	4 million – >400 million
"some" 0.13-micron technologies	10,000 or 100,000 FIT/Mbit*	1E-11 – 1E-10	10,000 – 100,000
1Gbit* memory in 0.25µm*	One error per week	6E-12	6,000
4M SRAM	<1E-10 upset*/bit-day	<4.2E-12	<4,200
1 Gbit of DRAM (Nite Hawk)	2.3E-12 upset/bit-hour*	2.3E-12	2,300
SRAM and DRAM	1 – 2 E-12 upset/bit-hour	1 – 2E-12	1,000 – 2,000
~8.2 Gbits of SRAM (CRAY YMP-8)	1.3E-12 upset/bit-hour	1.3E-12	1,300
SRAM	1,000 FIT/Mbit	1E-12	1,000
256 MBytes*	One error per month	7E-13	700
160 Gbits of DRAM (Fermilab)	2.5 errors per day	7E-13	700
32 Gbits of DRAM (CRAY YMP-8)	6E-13 upset/bit-hour	6E-13	600
MoSys 1T-SRAM (no ECC*)	500 FIT/Mbit	5E-13	500
Micron estimate, 256 MBytes	2 – 4 errors per year	1.2 – 2.4E-13	120 - 240
"ultra-low" failure rates	50 to 100 FIT per Mbit	5E-14 – 1E-13	50 – 100

Physical Solutions are hard

- Shielding?
  - No practical absorbent (e.g., approximately > 1 m of concrete)
  - unlike Alpha particles
- Technology solution: SOI?
  - Partially-depleted SOI of some help, effect on logic unclear
  - Fully-depleted SOI may help, hard to manufacture in high volumes
- Radiation-hardened cells?
  - 10x improvement possible with significant penalty in performance, area, cost
  - 2-4x improvement may be possible with less penalty
- Some of these techniques will help alleviate the impact of Soft Errors, but not completely remove it

Testing

- Accelerated testing and predictions
  - Use concentrated particle beam to "bomb" chips
  - Found to agree with long-term field testing very well



Arvutitehnika instituut
ati.ttu.ee

Software Failures

Software

- Is software getting worse?
  - Tandem OS (1985): 4 MLOC
  - Linux (2001): 30 MLOC (kernel 2.4 MLOC)
  - Windows XP (2001): 40-50 MLOC
  - Jim Gray's estimate: 1 bug/KLOC
  - Reducing bugs/KLOC vs. increasing KLOCs/product

Failures

- Hard to pinpoint a single root cause:
  - Coca-cola → disk crash → database failure
- Software bugs are faults!

Types of Bugs

- **Heisenbug**: disappears (or manifests differently) when you try to research it
  - Named after "Heisenberg uncertainty principle"
  - Debug mode versus release mode
    - Uninitialized variables
    - Fandango on core
  - Race conditions

Types of Bugs

- **Bohrbug**: constant, reproducible, easy to deal with
  - Named after the Bohr atom model
  - Bohrbug does not disappear or alter its characteristics when it is researched

Types of Bugs

- **Schrödingbug**: only starts manifesting when
  - is used in an unusual way
  - someone realizes it should be there
  - Named after Schrödinger's cat thought experiment
  - Determinism!
  - It is important to repair, not to determine the cause
  - For example: DB system works with small amount of data but not with many records

Types of Bugs

- **Mandelbug**: underlying cause is so complex and obscure, it makes the bug seems nondeterministic
  - Named after fractal innovator Benoît Mandelbrot
  - A bug whose behavior does not appear chaotic, but whose causes are so complex that there is no practical solution.
  - For example: a flaw in the fundamental design of the entire system.

Duration of Failures

- Permanent failure: once it manifests, won't go away unless you repair the system
  E.g., cut a network cable
- Intermittent failure: only occurs on occasion, for unknown reasons (until debugged… often workload)
  E.g., Patriot missile defense
- Transient failure: if you wait or retry, goes away
  E.g., various media corruption

Software Failures

- crash
- hang
- respond correctly but too late
- provide wrong data
- how to classify ? (fail-stop, fail-fast, Byzantine)
- how does recovery affect classification ?

Bug Triggers

- Timing
  - interleaving of events → many execution traces
  - hard to test all
- Recovery code
  - deals with exceptions → hard to simulate prior to shipping (ex. check NULL on return from malloc())
  - fault injection often used
- Third-party code
  - customer software, drivers, extensions, library users
  - Microsoft's "driver certification" → a way to combat this
- Boundary conditions
  - simple ones found through static analysis, complex ones are hard
- Bug-fix patches
  - customer system diverges over time
  - OS patches particularly evil

A Solution

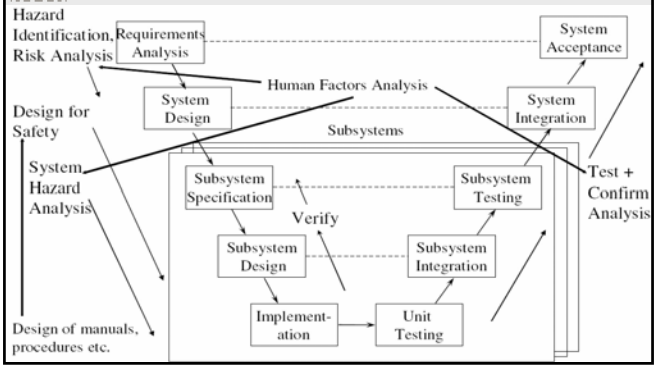
- In the Web community: high availability is achieved via three-tiered model:
  - Reliable back end (databases)
  - Stateless middle tier (application servers)
  - Front end (web servers)
- Other communities?

Human Factors

Human Factors

- The role of humans in safety-critical systems
- Human Reliability Analysis
  - task analysis
  - human error identification
    - human error model: Reason
  - human reliability quantification
  - mitigating human error
- Safe user interface design

Human Factors



Have we learnt since Therac-25

Software for Certain Medtronic Implanted Infusion Pumps Recalled

FDA Patient Safety News: Show #32, October 2004

- Medtronic is recalling certain software application cards.They're used in the company's Model 8840 N'Vision Clinician Programmers. These hand-held devices are used to program a number of implantable devices, including the SynchroMed and SychroMed EL implantable infusion pumps.

Have we learnt since Therac-25

- The recall is prompted by reports of data entry errors that have led to serious drug overdoses, including two patient deaths. The overdoses occurred when clinicians who were programming the pump entered the wrong time duration or the wrong interval --- for example, mistakenly putting the time interval between periodic drug boluses in the "minutes‟ field, instead of the "hours" field.

Have we learnt since Therac-25

- The recalled software may have contributed to these errors because one part of the screen did not have labels on the fields for hours, minutes, and seconds. Medtronic is now distributing replacement software that adds time labels to the screen to help reduce the risk of these kinds of programming errors.

Have we learnt since Therac-25

- If you use the Model 8840 N'Vision Programmer with SynchroMed or SynchroMed EL infusion pumps, the company says you should pay particular attention to selecting the appropriate time field when entering time duration or time intervals. You should also be sure to check your software application card for your N'Vision Programmer. If you have the older software version (AAA 02), Medtronic says you should order the new version (AAD 02).

Automation

- A driving force of automation is to compensate for human disadvantages
  - humans are unreliable components of systems requiring replacement by reliable computers
  - humans have limited capabilities in response time and capacity
- However, humans play an essential role in safety-critical decision making
  - computers are not flexible or adaptable, e.g., response in emergency situations
  - computers cannot make creative judgements or strategic decisions

Human Error and Risk

- Automation yields
  - Increased capacity and productivity
  - Reduction in manual workload and fatigue
  - Increased safety
- But
  - Need specialised training
  - Cost of maintenance
- Impact on human operators
  - Unclear if overall workload reduced
  - Increased complacency due to overconfidence?

Role of Humans

- **Monitor**: detecting errors
  - it may not be possible to determine if an error has occurred
  - the system may provide inadequate feedback
  - operators may become complacent
- **Backup**: in an emergency
  - operators may become de-skilled
  - information provided may be inadequate for intervention
  - automated systems are usually too complicated

Role of Humans

- **Partner**: responsible for part of a task
  - humans may be assigned "hard to automate" part
  - humans may be responsible for monitoring and maintaining
  - division of responsibility may make building a mental model harder

Do Humans Cause Most Accidents?

- 85% of work accidents are due to **unsafe acts by humans** rather than unsafe conditions
- Should we believe the statistics?
  - Data may be biased and incomplete: in 60-80% of accidents caused by operator's loss of control, 75% of those had system/safety malfunction that preceded the operator action
    - e.g. DC-10 crash deemed pilot error, involved autopilot headings alteration without telling the crew
  - Positive actions are not usually recorded
    - only 10% of recovery from emergency are pilot errors
  - Operators are expected to always recover from emergency
    - Error can be due to poor design

Do Humans Cause Most Accidents?

- Should we believe the statistics?
  - Operators have to intervene at limits, diagnose/respond quickly
    - E.g. consequences can be serious
  - Hindsight allows to identify a better decision
    - Operator's knowledge may be partial, or understanding erroneous
  - Separating operator error from design error is difficult
    - Examples from nuclear power plants:
      - Dials measuring the same quantities calibrated in different scales
      - Location of critical decimal points unclear
      - Critical displays located at back panels
      - Labels/colours inconsistent and misleading

What are humans good at?

- Detecting correlations and exceptions
  - Patterns/clusters in graphical data
  - Breaks in lines
  - Visual/sound disturbances
- Detecting isolated movement
  - Waving
  - Flashing lights
- Detecting differences
  - Sounds, alarms, etc
  - Lights on/off
  - etc.

Example of Dial Controls

- **Bad interface**, cannot tell normal from abnormal.
- Advice is to fix normal at 12 o'clock position.

Example of Dial Controls

- **Good interface**: can spot abnormal position even for 5 deg change

Humans vs Machines

- Where machines have advantage…
  - Sensing/Actuating: broader range of sensors, able to perform in harsh environments
  - Cognition: no boredom, precision of calculations, repeatability, predictability
- Where humans have advantage…
  - Sensing/Actuating: image processing, edge & anomaly detection, flexibility
  - Cognition: ability to respond in unknown situations
- Should you trust humans or machines?
  - Boeing trusts people (pilot has ultimate authority).
  - Airbus trusts machines (flight control software has authority over pilot).

Human Machine Interaction (HMI)

- Hybrid discipline: psychology, engineering, ergonomics, medicine, sociology, mathematics
- Concerned with the impact of human operators and maintainers on system performance, safety and productivity
- Concerned with enhancing the efficiency, flexibility, comprehensibility and robustness of user interaction
- In the safety-critical context, the primary concern is to enhance robustness, possibly at the expense of efficiency and flexibility

Human Reliability Analysis (HRA)

- Identify potential operator errors that may lead to hazards and reduce error where risk is sufficiently high
- Four steps:
  - **task analysis**: characterise the actions performed to achieve particular goals
  - **human error identification**: identify possible erroneous actions in performing a task
  - **human reliability quantification**: estimate likelihood of error
  - **mitigation of human error**: identify control options

Task Analysis

- Tasks are activities to transform some given initial state into a goal state, i.e., goal-directed
- Structured from sub-tasks and elementary actions
- Each elementary action is concerned with a manipulation to be performed upon an object in the task domain
- Procedures for
  - normal operation of the system
  - maintenance of the system
  - emergency situations
- Logical sequence of actions that the operator engages in and the detailed physical executions that the operator

Human-Task Mismatch

- Human error is not a useful term
  - Implies possible to improve humans
- Human-Task Mismatch better term
  - Erroneous behaviour inextricably connected to the behaviour needed to complete a task
- Tasks
  - Involve problem solving, decision making
  - Need adaptation, experimentation, optimisation
- Levels of cognitive control [Rasmussen's]
  - Skills-based behaviour (smooth sensory based)
  - Rule-based behaviour (conscious problem solving)
  - Knowledge-based behaviour (goal known, planning by selection, trial and error, etc)

Experimentaton versus Error

- Designer relies mostly on knowledge-based behaviour
- Operator employs all three
  - In training, from knowledge- or rule-based to skills based
  - In unfamiliar situation, use knowledge-based to develop rules-based
  - Needs to maintain knowledge-based throughout
- Experimentation
  - Test a set of hypothesis through mental reasoning
  - May be unreliable
- Human error
  - unsuccessful experiments, in unkind environment
- Design for error tolerance

Human as Monitor

- Monitoring, rather than active control
  - Responsible for detecting/repairing problems
- Humans perform badly…
  - Task may be impossible
    - Cannot check in real-time if computer performs correctly
  - Operator dependent on information provided
    - Too much or too little is bad
  - Information is indirect
    - System handles most functionality
  - Failures may be silent or masked
    - E.g. autopilot disengages
  - Tasks are such that lower alertness results
    - Mechanical, lack of stimulation, can act without noticing

Human as Back-up

- Emergency only, rather than active control
  - Expected to take appropriate action
- Good design is essential
  - Can lower proficiency and increase reluctance to intervene
    - Infrequent usage
    - Cognitive and physical skills decline in absence of practice
    - High skills often needed!
      - E.g. emergency shutdown of nuclear plant
  - Fault-intolerant systems may lead to larger errors
    - May fail in ways difficult to anticipate
  - Harder to manage in crisis
    - Not fully aware of the internal state
    - Computer support for decision making

Human as Partner

- Both humans and automated system assigned control tasks
  - Number of human tasks reduced
  - Must be planned appropriately
- Modes
  - Partial automation
  - Shared control (primary responsibility with humans, but computer continuously performs checks)
- Potential problems
  - Good mental models are important
    - Must know the system state
  - Good communication is essential
    - Clarity, correctness

Accident Models

- Reduce description of accident to a set of events and conditions
  - Used in investigations, for prediction, etc
- Domino models
  - Social environment
  - Fault of a person
  - Unsafe act or mechanical/physical hazard
  - Accident
  - Injury
- Chain-of-events
  - Event trees, fault trees
- System theory
  - Accidents result from complex interactions

Human Tasks

- Simple tasks
  - Uncomplicated sequences
- Vigilance tasks
  - Detection of signals
- Emergency response tasks
  - May involve complex reactions
  - Performed under stress
- Complex tasks
  - Defined tasks, involve decision-making

Human Error Models

- Cognitive, e.g. Reason's model eight primary error groups
  - False sensation (lack of correspondence between subjective experience and reality)
  - Attentional failures (distraction, dividing attention)
  - Memory lapses (forgetting items)
  - Unintended words/actions
  - Recognition failures (wrongly observed signals)
  - Inaccurate and blocked recall (misremembering sequences)
  - Errors in judgement (misconceptions)
  - Reasoning errors (false deduction)
- Also Norman model of slips, mistakes in planning

Human-Task Mismatch again…

- Errors are an integral part of learning!
- Mechanisms of human malfunction
  - Skills-based level
    - Disorientation, motor skills failure
    - Stereotype take-over
  - Rule-based level
    - Incorrect recall of rules
    - Stereotype function
  - Knowledge-based level
    - Mental overload
    - Premature hypothesis (way of least resistance, point of no return)
- Also performance affecting factors (separately)
  - Work conditions, stress, social aspects

Human Factors Summary

- ✓ Understanding cognitive aspects essential
- ✓ Probability of failure difficult to predict
  - Human response affected by stress, fatigue, etc
- ✓ Must assume human error will happen sooner or later
  - Hardware support, failsafe operations
- ✓ Design for safety
  - Fault-tolerance
  - HCI (layout, communication, correctness etc)

Arvutitehnika instituut
ati.ttu.ee

Questions?

Gert Jervan

Tallinna Tehnikaülikool
Arvutitehnika instituut