Department of computer Engineering
ati.ttu.ee

# IAF0030
# Arvutitehnika erikursus I

Süsteemide usaldusväärsus ja veakindlus
Dependability and fault tolerance

## Loeng 7
## Enemies of Dependability

**gert.jervan@pld.ttu.ee**

Tallinn University of Technology
Department of Computer Engineering
Estonia

---

## Important Dates

- ✓ Case Study drafts: April 6 (Monday)
- ✓ Case Studies discussion: April 7
  (Be there or be square!)
- ✓ Presentations: April 28, May 5 (5+5)
- ✓ **Final report: May 19, 12:00** (sharp!)

---

## Lecture Outline

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

---

## Downtime

- ✓ Planned downtime
  - Maintenance, repair, upgrade
- ✓ Unplanned downtime
- ✓ Dependability:
  - Turn unplanned downtime 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% | ? |

---

Department of computer Engineering
ati.ttu.ee

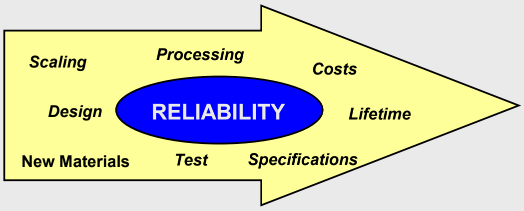
## Hardware

## 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

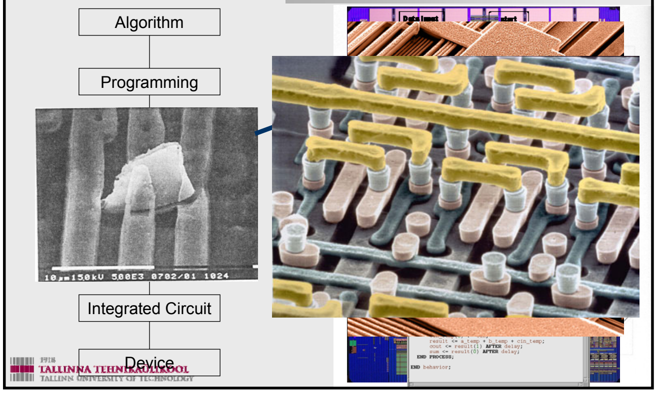
---

## 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



Algorithm

Programming

Integrated Circuit

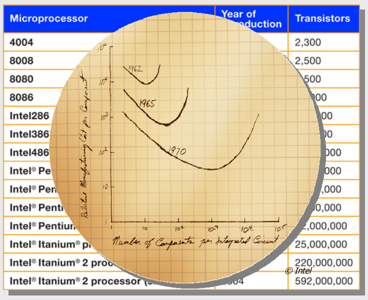
Device

```
    result <= a_temp + b_temp + cin_temp;
    cout <= result(1) AFTER delay;
    sum <= result(0) AFTER delay;
  END PROCESS;

END behavior;
```

---

## Moore's Law



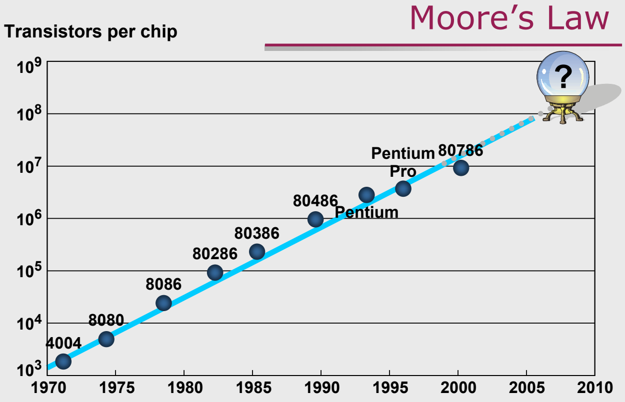
| Microprocessor | Year of ...oduction | Transistors |
|---|---|---|
| 4004 | | 2,300 |
| 8008 | | 2,500 |
| 8080 | | 500 |
| 8086 | | 00 |
| Intel286 | | 00 |
| Intel386 | | 0 |
| Intel486 | | 000 |
| Intel® Pe | | 000 |
| Intel® Pen | | ,000 |
| Intel® Penti | | 0,000 |
| Intel® Pentiun | | 2,000,000 |
| Intel® Itanium® p | | 25,000,000 |
| Intel® Itanium® 2 proc | | 220,000,000 |
| Intel® Itanium® 2 processor ( | 004 | 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**

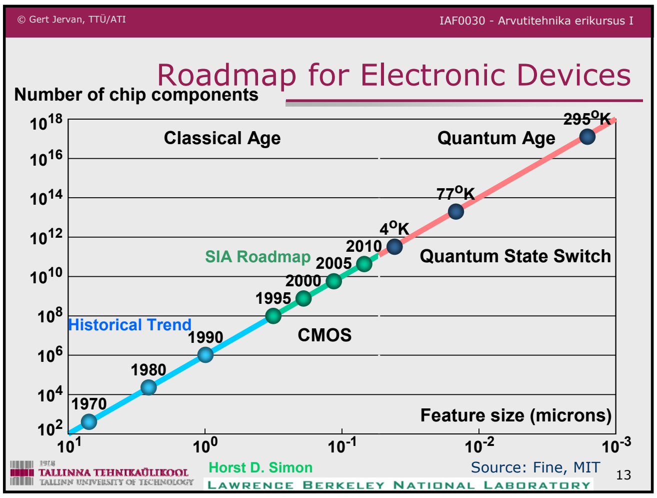
---

## Moore's Law



Source: Fine, MIT

## Roadmap for Electronic Devices



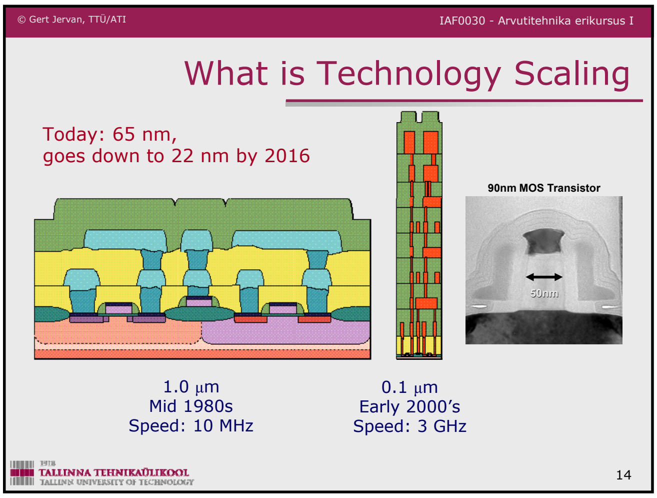
Horst D. Simon

Source: Fine, MIT

Lawrence Berkeley National Laboratory

## What is Technology Scaling

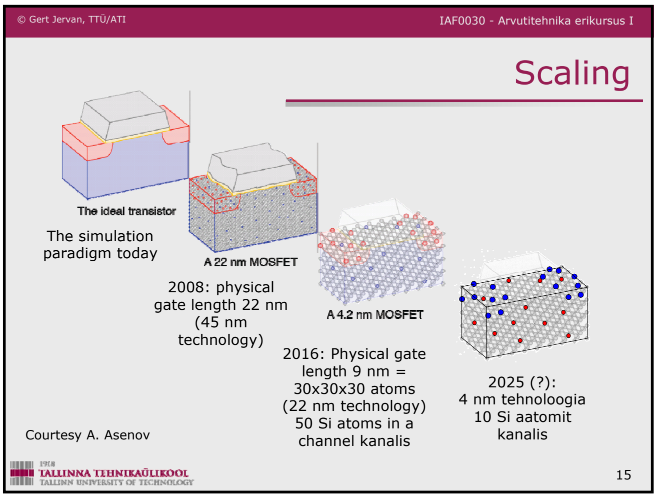
Today: 65 nm,
goes down to 22 nm by 2016



1.0 µm
Mid 1980s
Speed: 10 MHz

0.1 µm
Early 2000's
Speed: 3 GHz

## Scaling



The simulation paradigm today

2008: physical gate length 22 nm (45 nm technology)

2016: Physical gate length 9 nm = 30x30x30 atoms (22 nm technology) 50 Si atoms in a channel kanalis

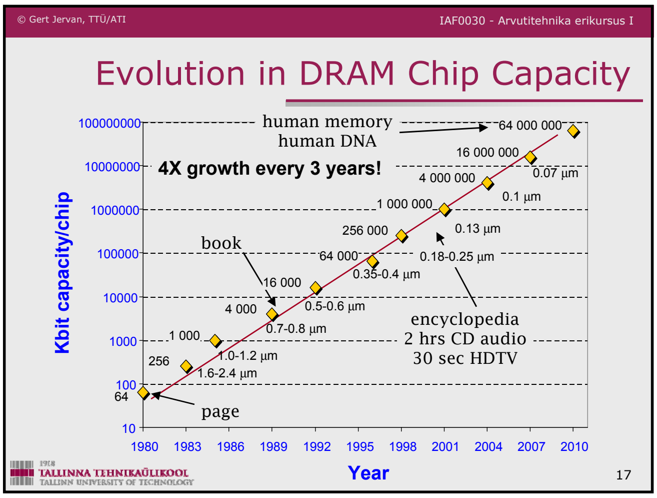
2025 (?): 4 nm tehnoloogia 10 Si aatomit kanalis

Courtesy A. Asenov

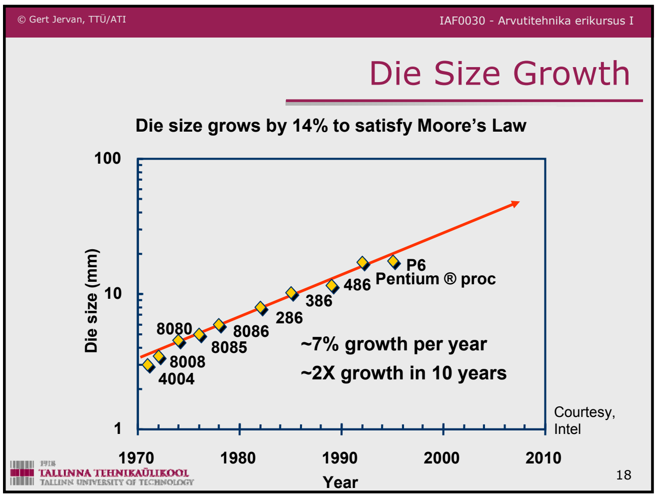
## Benefits of Technology Scaling

- ✓ Benefits of scaling the dimensions by 30%:
  - Reduce gate delay by 30% (increase operating frequency by 43%)
  - Double transistor density
  - Reduce energy per transition by 65% (50% power savings @43% increase in frequency)

## Evolution in DRAM Chip Capacity



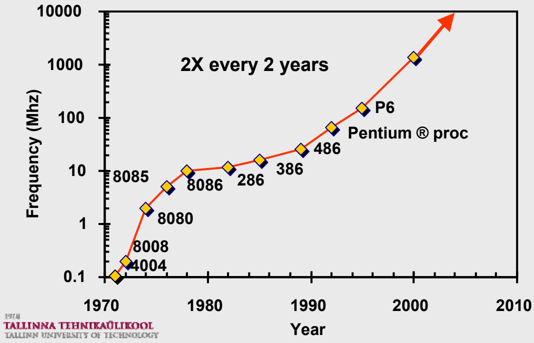
## Die Size Growth



Courtesy, Intel

## Clock Frequency

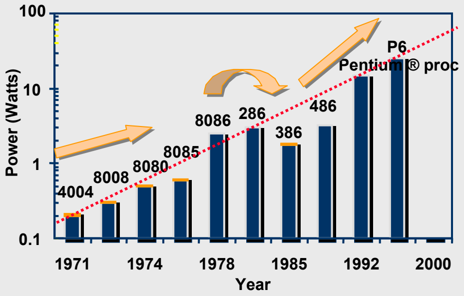
**Lead microprocessors frequency doubles every 2 years**



Courtesy, Intel

## Power Dissipation

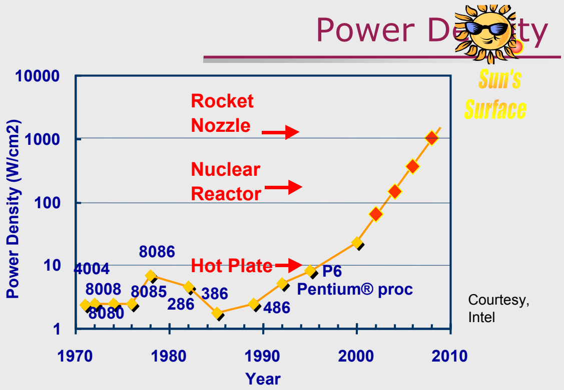
**Lead Microprocessors power continues to increase**



Courtesy, Intel

**Power delivery and dissipation will be prohibitive**

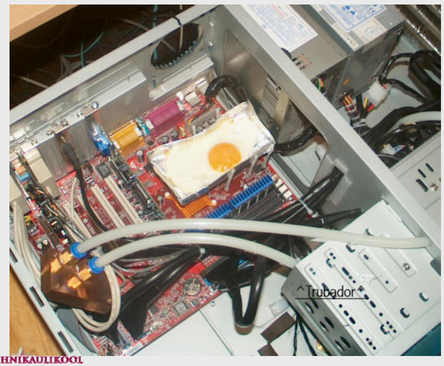
## Power Density



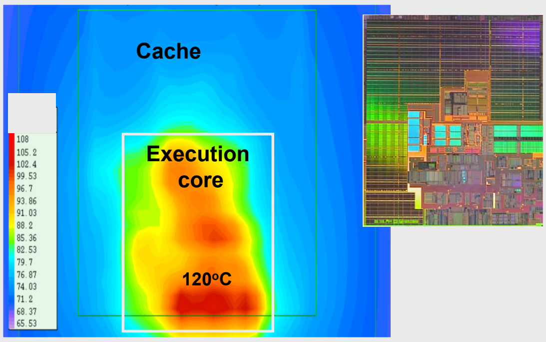
Courtesy, Intel

**Power density too high to keep junctions at low temp**

## Hot Chips



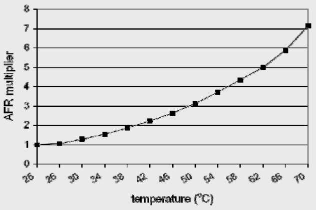
## Thermal map: 1.5 GHz Itanium-2



[Source: Intel Corporation and Prof. V. Oklobdzija]

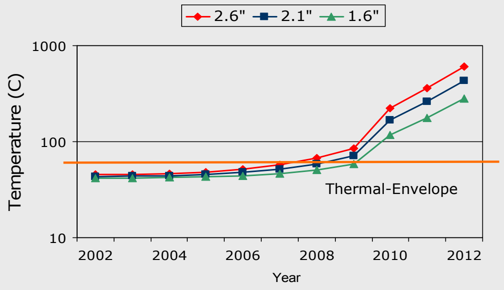
## Temperature Affects Disk Drive Reliability

- ✓ 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



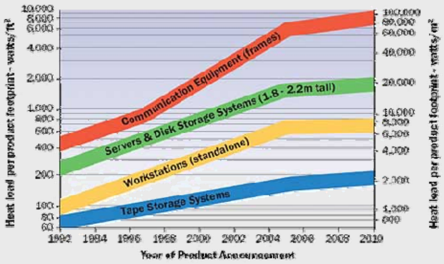
Source: D. Anderson et al, “More than an Interface – SCSI vs. ATA”, FAST 2003.

## Drive Temperature

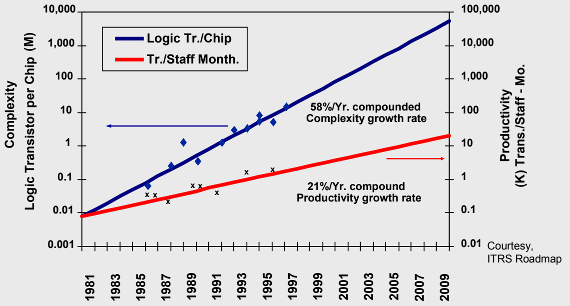


40% annual growth in the data-rate

## Heat Density



## Design Productivity Trends



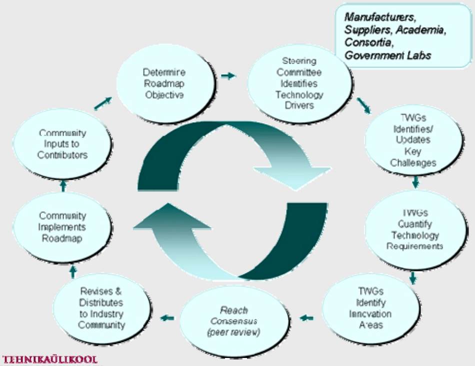
Courtesy, ITRS Roadmap

**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

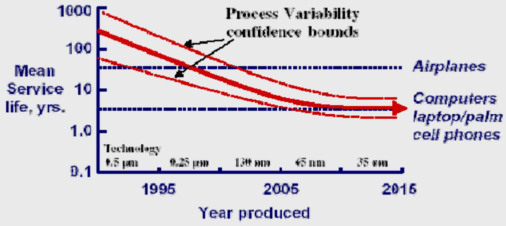
- ✓ Significant rise in the amount of proprietary technology and data developed by manufacturers, reluctance to share information with hi-rel customers
  - e.g. process recipes, process controls, process flows, design margins, MTTF
- ✓ Next generation microelectronics focus on the performance needs of 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 | <60 | ?? |
| 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



*Known trends for TDDB, EM and HCI degradation*

*Extrapolated from ITRS roadmap, MRQW-2002, Bernstein

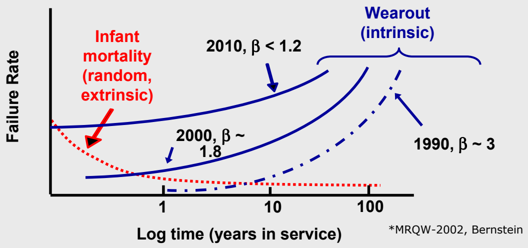
## Impact of scaling on wear-out failure mechanisms

- ✓ 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.



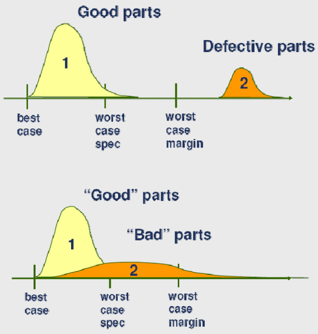
*MRQW-2002, Bernstein

---

## Correct of defective?

Theory:

Reality:



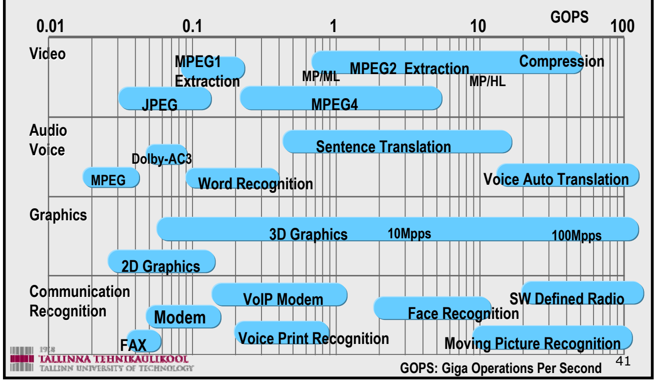
---

Department of computer Engineering
ati.ttu.ee

**Why it is all needed???**

---

## Req'd Performance for Multi-Media Processing



GOPS: Giga Operations Per Second

---

## Implications to Design

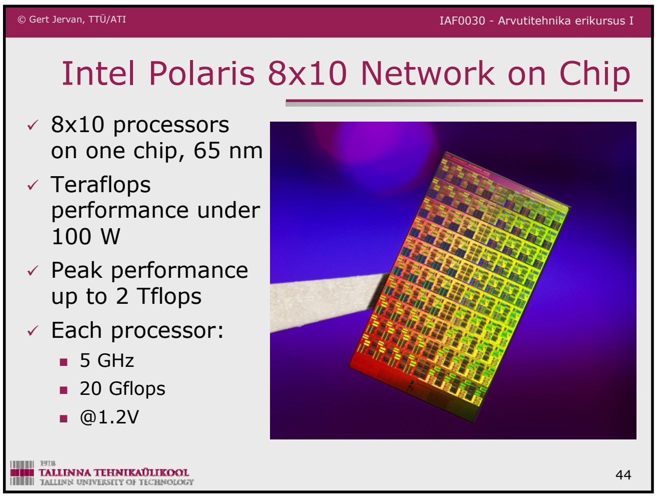
- ✓ Design fabric will be Regular
- ✓ Will look like Sea-of-transistors interconnected with regular interconnect fabric
- ✓ Shift in the design efficiency metric
  - From Transistor Density to Balanced Design

**BUT**

- ✓ Manufacturing of these sub-nanometer chips defect-free is almost impossible (yield is below acceptable levels)
- ✓ Increasing importance of transient and intermittent faults (due to the environment)

## New Architectures

- Massively parallel architectures (Von Neuman is dead...) based on hundreds (millions) of (non-) reliable components
  - Multiple Input stream, Multiple Data stream machines
  - Wide use of network infrastructures (Networks-on-Chip)
  - Built-In Self-Repair will become a widespread technology
    - Dynamic reconfiguration



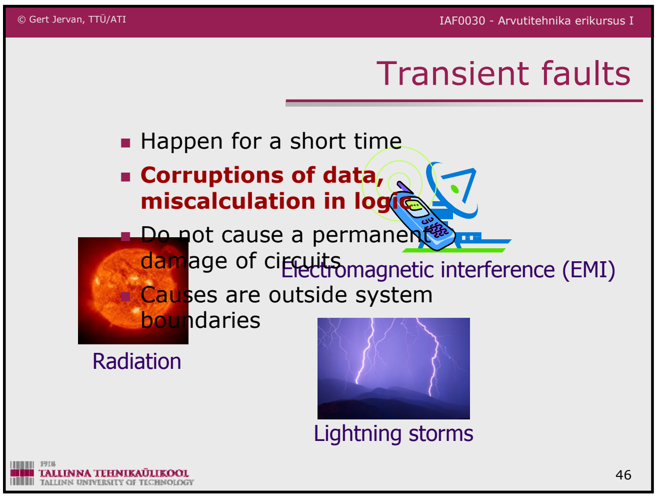
## Intel Polaris 8x10 Network on Chip

- 8x10 processors on one chip, 65 nm
- Teraflops performance under 100 W
- Peak performance up to 2 Tflops
- Each processor:
  - 5 GHz
  - 20 Gflops
  - @1.2V

TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering
ati.ttu.ee

## Soft Errors



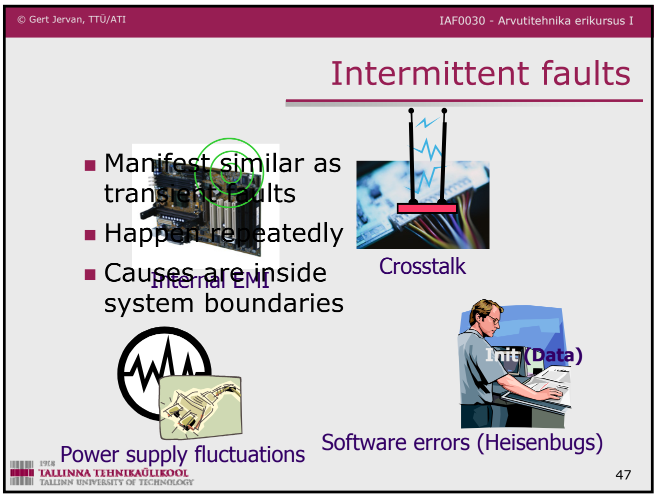
## Transient faults

- Happen for a short time
- **Corruptions of data, miscalculation in logic**
- Do not cause a permanent damage of circuits
- Causes are outside system boundaries

Electromagnetic interference (EMI)

Radiation

Lightning storms



## Intermittent faults

- Manifest similar as transient faults
- Happen repeatedly
- Causes are inside system boundaries

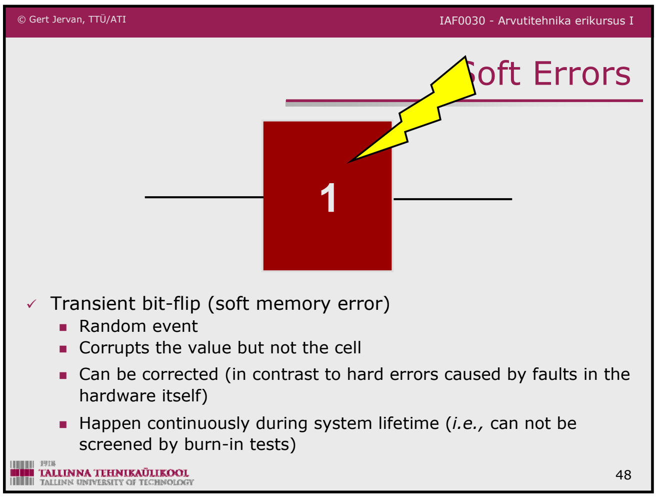
Internal EMI

Crosstalk

Init (Data)

Power supply fluctuations

Software errors (Heisenbugs)



## 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

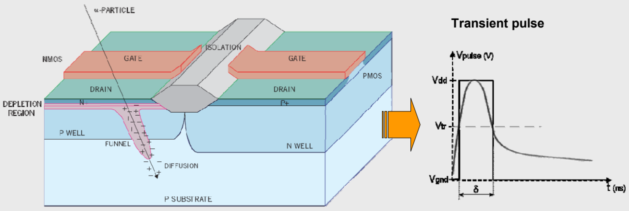
- ✓ First traced to alpha particlce emissions from chip packaging materials
  - Most sources removed (pure materials, different designs, shielding)
- ✓ Today’s main problem: cosmic radiation
  - Cosmic particles from deep space (actually 5th- or 6th-hand collision particles)
    - At ground level ca 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

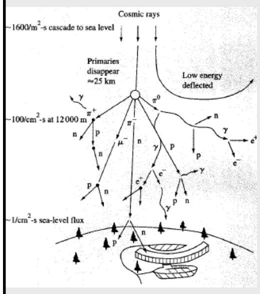
| SER type | Source | Mechanism | Trend |
|---|---|---|---|
| Alpha | Thorium and uranium contamination in-mold compound, silicon, or lead bumps | 2- to 9-MeV alpha particle creating electron-hole funnel traveling 25 microns in silicon | Exponential increase with scaling |
| Cosmic | Intergalactic sources 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



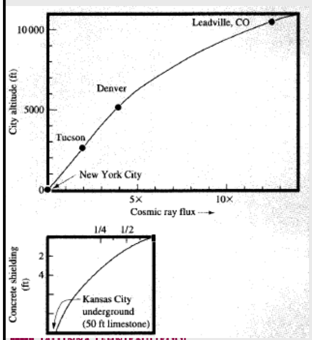
The electric field in the depletion region directly generates electron-hole pairs in its wake, causing the charges to drift so that the transistor sees a current disturbance

# 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

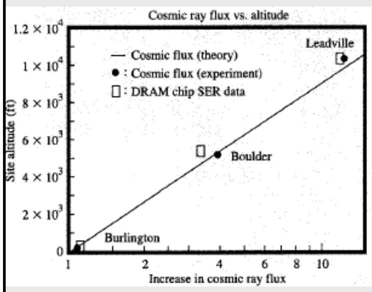


This plot shows how the soft-error rate of an LSI chip changes with altitude. Shown are the altitudes of New York City (sea level), Tucson (2,390 ft), Denver (5,280 ft), and Leadville, CO (10,152 ft).

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.

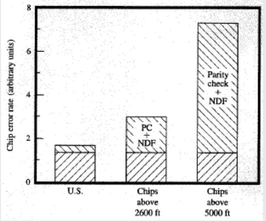
Ziegler, et al., “IBM experiments in soft fails in computer electronics (1978 - 1994),” IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

# 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.

## The Altitude Factor

- ✓ A person on an airplane over the Atlantic at 35,000 ft working on a laptop with 256 Mbytes (2 Gbits) of memory. At this altitude, the SER of 600 FITs per megabit becomes 100,000 FITs per megabit, resulting in a potential error every five hours.
- ✓ 1 FIT (failures in time), is the number of failures in 1 billion device-operation hours. A measurement of 1000 FITs corresponds to a MTTF (mean time to failure) of approximately 114 years.

## 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

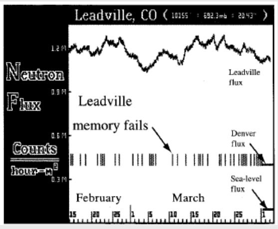
| Type of Memory | Reported SER | Error per bit-hour | FIT*/Mbit |
|---|---|---|---|
| Goal for new Cypress products | 200 FIT* | ? | ? |
| SRAM (quoted by vendors) | 200 to 2,000 FIT | ? | ? |
| "typical" | 1,000 FIT | ? | ? |
| DRAM at full speed | Few hundred to few thousand FIT | ? | ? |
| SRAMs 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



Department of computer Engineering
ati.ttu.ee

**The problem to be solved:**

**How to design reliable system out of non-reliable hardware?**

Department of computer Engineering
ati.ttu.ee

## Software Failures

Some information on the following slides: © George Candea

## Software

- ✓ Is software getting worse?
  - Tandem OS (1985): 4 MLOC
  - Linux (2001): 30 MLOC (kernel 2.6.29: 11 MLOC)
  - Windows XP (2001): 35 MLOC
  - MS Vista (2006): 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

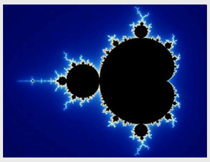
- ✓ **Bohrbug**: constant, reproducible, easy to deal with
  - Named after the Bohr atom model
  - Bohrbug does not disappear or alter its characteristics when it is researched
  - Ghost in the code
    - Overflow

## 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

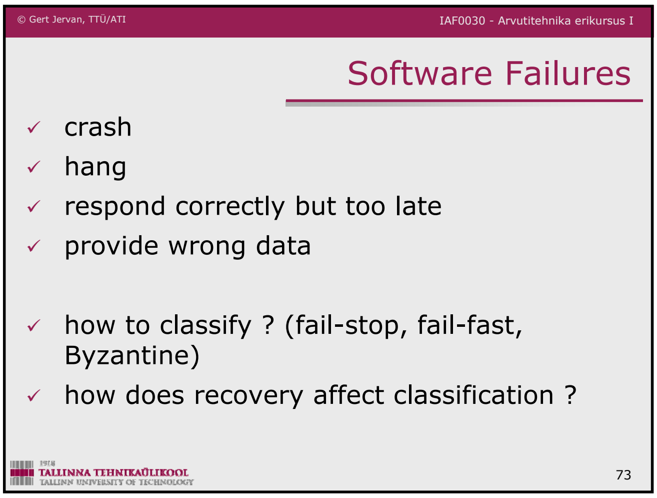
## Types of Bugs

- ✓ **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

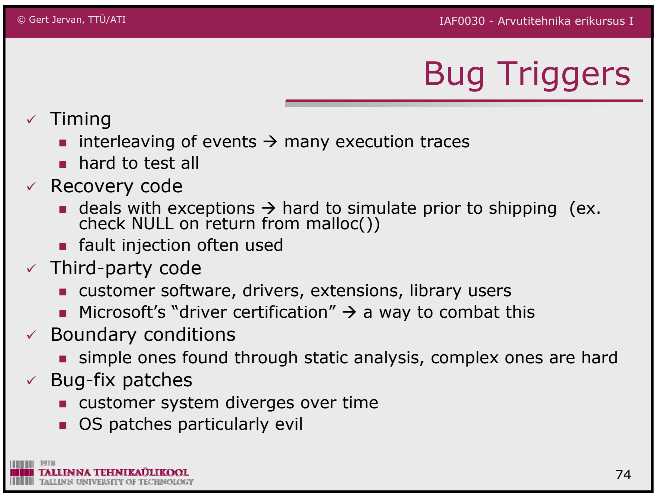
- ✓ Permanent failure: once it manifests, won't go away unless you repair the system
  E.g., cut a network cable
- ✓ Intermittent failure: only occurs on occasion, for unknown reasons (until debugged… often workload)
  E.g., Patriot missile defense
- ✓ Transient failure: if you wait or retry, goes away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



## Software Failures

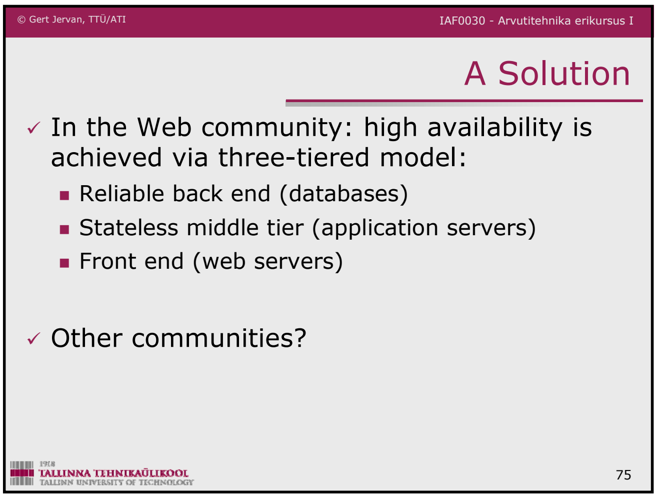
- ✓ crash
- ✓ hang
- ✓ respond correctly but too late
- ✓ provide wrong data

- ✓ 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?