

IAF0542

## Sardsüsteemid

Gert Jervan  
Arvutitehnika instituut  
ati.ttu.ee/~gerje




Graphics: © Alexander Nolle, Gert Jervan, 2003

© Gert Jervan

## Üldine informatsioon

- Sisu:  
Sardsüsteemid  
Embedded Systems  
või  
Cyber-physical Systems  
[www.pld.ttu.ee/IAF0542](http://www.pld.ttu.ee/IAF0542)
- Loengud ja eksamineerija:  
Gert Jervan  
IT-229 620 2261  
gert.jervan@ati.ttu.ee  
ati.ttu.ee/~gerje



© Gert Jervan

## Üldine informatsioon

- Laboritega tegelevad doktorandid Maksim Gorev ja Vadim Pesonen, TTÜ/ATI, IT-307

© Gert Jervan

## Eksam

- Valikvastustega küsimused
- Koodi analüüs
- Kirjeldavad/lahendust pakkuvad küsimused
- Eesti/inglise keeles
- Eeltingimuseks laborite sooritamine! (range nõue)
- Laborid tuleb kaitsta semestri jooksul!

© Gert Jervan

## Sissejuhatus

Miks on sellist ainet vaja?

© Gert Jervan

## Miks?

- Erinevate ennustuste kohaselt võib tuleviku arvutisüsteemid kirjeldamiseks kasutada järgnevaid termineid:
  - Nähtamatud arvutid
  - Lausandmetöötlus (Ubiquitous, pervasive computing)
  - Arukad keskkonnad (Ambient intelligence)
  - PC-järgne ajastu
  - Küber-füüsikalised süsteemid
- Seda võimaldavad tehnoloogiad:
  - Sardüsteemid
  - Kommunikatsioonitehnoloogiad

© Gert Jervan

## Miks?

- "Information technology (IT) is on the verge of another revolution. ....
- networked systems of embedded computers ... have the potential to change radically the way people interact with their environment by linking together a range of devices and sensors that will allow information to be collected, shared, and processed in unprecedented ways. ...
- The use ... throughout society **could well dwarf previous milestones in the information revolution.**"

National Research Council Report (US)  
Embedded Everywhere, 2001



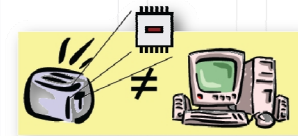
The future is embedded,  
embedded is the future

## Sissejuhatus

Mis on sardsüsteemid?

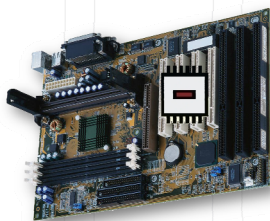
## Arvutitehniline vaade

- Arvuti, mis ei paista väljastpoolt välja nagu arvuti
- Suhtleb välise maailmaga
- Võib omada väga keerukat või primitiivset kasutajaliidest, või puudub see täiesti
- Osa suuremast süsteemist



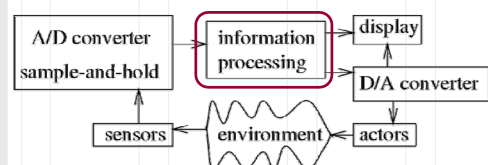
## Elektroniline vaade

- Toode, mis sisaldab programmeeritavat protsessorit
- Tarkvara on toote loomise lahutamatu koostisosa



## Sardsüsteem

- Arvutit kasutatakse toolistes süsteemides ennekõike süsteemi lihtsustamiseks ning pakkumaks paindlikust.
- Seadme kasutaja ei ole tihti isegi teadlik arvuti olemasolust.



## Sardsüsteemide levik



## Sardsüsteemid

„Dortmund“ Definition [Peter Marwedel]:

**Information processing systems embedded into a larger product**

Main reason for buying is **not** information processing

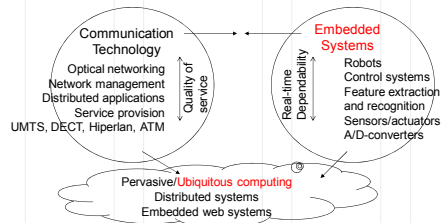
Berkeley Model [Edward A. Lee]:

**Embedded software is software integrated with physical processes. The technical problem is managing time and concurrency in computational systems.**

☛ **Definition:** **Cyber-Physical (cy-phy) Systems (CPS)** are integrations of computation with physical processes [Ed Lee, 2006].

## Sardsüsteemid ja lausandmetootlus

- Lausandmetootlus: Informatsioon, mis on kõikjal ja alati.
- Sardüsteemid on põhiline tehnoloogia selle realiseerimiseks



## Sardsüsteemide tähtsus (1)

- Sardüsteemid osatähtsus (nii meie igapäevases elus kui ka rahalises mõttes) kasvab VÄGA kiiresti:
  - 4 miljardit mobiiltelefonide kasutajat aastaks 2011, iga-aastane müük: 1 miljard [Nokia@Norchip 2008]
  - 2011 müüakse maailmas rohkem nutitelefone kui PC-sid. [www.itfacts.biz]
  - WiFi toetusega seadmete arv kahekordistub iga aastaga [www.itfacts.biz]



## Sardsüsteemide tähtsus (2)

- Püsiühenduste arv on kasvanud 5 mln. kliendilt (1999) 215 miljoni kliendini (2005) ja 635 kliendini (2010) [www.itfacts.biz]
- USA DVRi (digital video recorders) kasutajaskond kasvas 5% majapidamistest (2004) 41% (2009). [www.itfacts.biz]

## Sardsüsteemide tähtsus (2)

- Euroopas on autotööstusega otseselt seotud rohkem kui 4 mln. inimest, kuid kaudselt rohkem kui 8 mln. Kokku annab nimetatud sektor 7% EU GNPst. [OMI bulletin]
- Elektroonika moodustab tänase auto hinnast ca. 25% (võib tõusta kuni 50%-ni) [EE Times 2010].
- Globaalse mobiilse meelelahutuse väärtus on 32 mld. USD, kasvades 2010 28% [www.itfacts.biz]
- Koduste meditsiiniliste jälgimissüsteemide aastatulu peaks 2011 olema ca. 225 mln USD (2006 – 70 mln USD) [www.itfacts.biz, Sep. 4th, 2007]

## Sissejuhatus

Rakendusvaldkonnad

## Rakendusvaldkonnad

- Autode elektroonika
- Lennundus
- Rongid
- Telekom
- Meditsiin



## Autod

- ✓ Erinevad funktsioonid
  - ABS: Anti-lock braking systems
  - ESP: Electronic stability control
  - Turvapadjad
  - Automaatkäigukast
  - Immobiliser
  - Surnud nurga hoiatusüsteem
  - ... jne ...
- Mitmed võrgud
  - Kere, mootor, telemetria, media, ohutus
- Mitmed protsessorid
  - Kuni 100
  - Ühendatud võrku
- Miljoneid ridu koodi



## Autod

- Väga erinevad protsessorid:
  - 8-bit – ukسلukud, tuled, jne.
  - 16-bit – enamus funktsioone
  - 32-bit – mootori kontroll, turvapadjad
- Uued vajadused:
  - Töötlamine seal, kus tegevus toimub
  - Sensorid ja aktuaatorid on hajutatud üle sõiduki

## Lennundus

- Juhtimissüsteemid (fly-by-wire)
- Ohutussüsteemid (TCAS, TAWS, GPWS, ...)
- Piloodi informatsioonisüsteemid (paperless cockpit)
- Energiasüsteemid, meelelahutussüsteemid, ...
- Kõige olulisem on **usaldusväärsus**

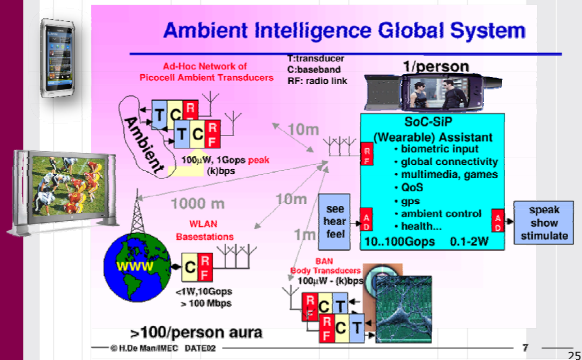


## Telekommunikatsioon

- Üks kiireima kasvuga sektoreid
- Uued (asukoha põhised) teenused
- Ülikiired ühendused (LTE, ...)
- Spetsiifilised lahendused (militaar, pääste, politsei, ...)



## Arukad keskkonnad



## Meditsiinisüsteemid



## Turvalisus

- Lugerid (sõrmejälje, võrkkesta, ...)
- Läbipääsusüsteemid
- Turvasüsteemid (lennujaamad, ...)
- Smartpen@
- Smart cards
- ...



[tomsguide.com]

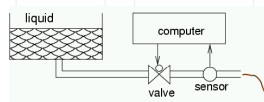
## Rakendusvaldkonnad

### Militaarrakendused



## Rakendusvaldkonnad (5)

### Tööstuslik automaatika



### Targad hooned



## Sissejuhatatus

Mõningad sardsüsteemide näited igapäevases elust

## Sammulugeja

- Tavaline arvutustöö:
  - Sammude lugemine
  - Aja arvestamine
  - Keskliste arvutamine
  - jne.
- Tõsine arvutustöö:
  - Sammude identifitseerimine
  - Sensor tunnetab seadme liikumist, mitte jalgade tööd



© Jakob Engblom

31

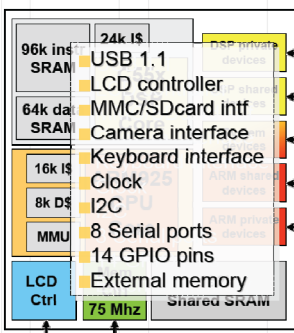
## Mobiiltelefonid

- Multiprotsessor
  - 8-bit/32-bit kasutajaliidesele
  - DSP raadio osale
  - 32-bit IR liidesele
  - 32-bit Bluetoothile
  - CPU (iPhone 4: 1 GHz ARM Cortex-A8 CPU koos PowerVR GPUga)
- Palju mälu (<32 GB storage, <512 MB DRAM)
- Individualiseeritud kiibid
- Kaamera, GPS, WiFi, ...
- Võimustarve & aku eluiga sõltuvad ennekõike tarkvarast



32

## Mobiiltelefonid: TI OMAP 5910



- Kahetuimaline HW/SW platvorm mobiiltelefonidele
  - ARM925T 150 MHz
  - TI C55 DSP 150 MHz
  - Võimsustarve: 230 mW
- Kasutavad Nokia, Sony-Ericssoni jpt.
- Analogiid:
  - Motorola
  - Infineon

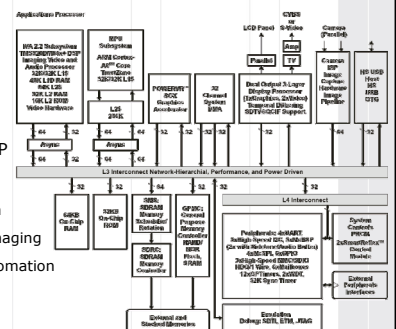
<http://focus.ti.com/docs/prod/folders/print/omap5910.html>

© Jakob Engblom

33

## Multipurpose: TI DaVinci (DM3730)

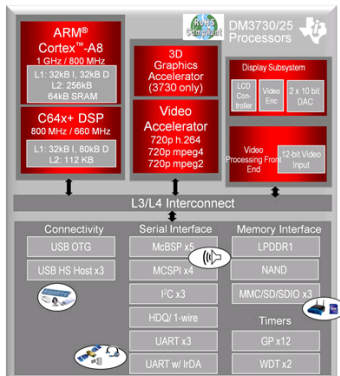
- 45 nm
- Android & Linux ready
- 1 GHz ARM
- 800 MHz DSP
- Applications:
  - Navigation
  - Medical imaging
  - Home automation
  - HCI
  - ...



34

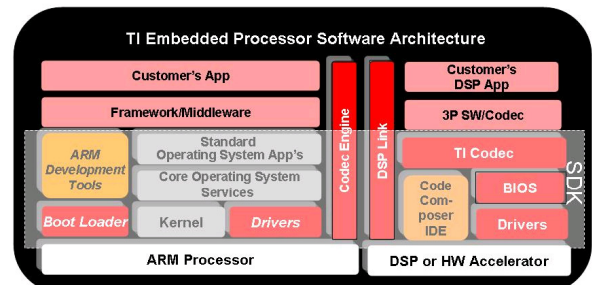
## DM3730/25 DaVinci™ digital media processor 720p30 encode/decode

- **Benefits**
  - 2000DMIPS for OSs like Linux™, Win CE, RTOS
  - Up to 30% reduction in power
  - 3-D graphics w/ up to 20M polygons per second for robust GUIs
  - Backward compatible with OMAP3530
- **Sample applications**
  - Smart connected devices
  - Patient monitoring
  - Single board computers
  - Low power PC
  - Media player
- **Power**
  - Dynamic Voltage and Frequency Scaling (DVFS)
  - Total Power: 735mW (800MHz)
  - Up to 40% reduction in power
  - Standby Power: 0.1mW (600MHz)
- **Schedule and packaging**
  - CBP: 12x12mm PBGA, 0.4mm, 515-ball, Package on Package (PoP)
  - CBC: 14x14mm PBGA, 0.5mm, 515-ball, PoP
  - CUS: 16x16mm PBGA, 0.65mm, 423-ball, Utilizes Via Channel™ array technology with 0.8mm pitch plus design rules



35

## Software Architecture – DM Devices

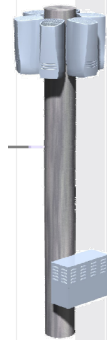


- = Industry Standard OS S/W Component – Free
- = TI Provided Components - Free
- = Customer, 3rd Party Code, or Open Source

SDK – Software Development Kit Available Free from TI

## Mobiilside tugijaamad

- Massiivne signaalitöötlus
  - Mitmeid tegumeid iga ühenduses oleva mobiiltelefoni kohta
- Põhinevad DSP-del
  - Standardised või individualiseeritud
  - Sadu protsessorid

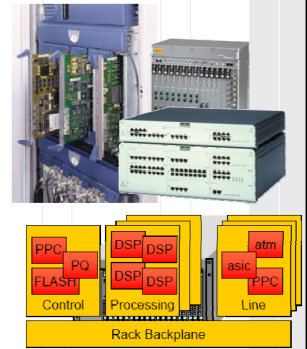


© Jakob Engblom

37

## Switchid

- Racki põhised
  - 12-20 kaarti, mitmeid riuleid, 100 CPUsid
  - Juhtimine, IO, DSP
- Kontroll ja andmed
  - Kontroll: üldisem
  - Andmed: DSP & ASIC
- Optilised ja vasepõhised ühendused
- Digitaal- & analoogsignaaliid
- Liiasus (võimus, HW)



© Jakob Engblom

38

## Targad keevitusaparaadid

- Elektronika kontrollib pinget ja traadi etteandmist
- Kohandub operaatoriga
  - andmete lugemine kHz sagedusel
  - 1000-id otsuseid/sekundis
- Ideaalne keevitus isegi viletsa operaatori korral
- Lihtsustatud kasutamiseks toode, kuigi ei tundu olevat seotud arvutitega...



© Jakob Engblom

39

## Õmblusmasin

- Kasutajaliides
  - Tikkimustrid
  - Puutetundlik ekraan
- "Tark"
  - Avaldab õiget survet
  - Vabastab riide operatsiooni lõppemisel
- Uusi funktsioone lisatakse läbi tarkvara uuenduste

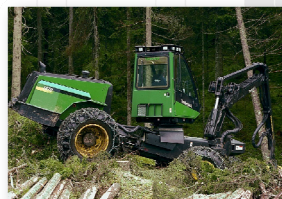


© Jakob Engblom

40

## Harvesterid

- Võrku ühendatud arvutisüsteem
  - Haaratsite ja töövahendite kontroll
  - Navigeerimine metsas
  - Raamatupidamine metsa ülestootmise kohta
  - Oluline efektiivseks tööks
- Protsessorid
  - 16-bitised protsessorid, ühendatuna CAN võrku
- Ekstreemsed tingimused



© Jakob Engblom

41

## C167CS

- Infineon
- Spetsiaalselt autotööstusele
- Protsessor
  - 16-bit C16X tuum
  - 4-osaline lihtne konveier
  - 40 MHz
  - 16 MB mälu
- 144 väljaviiguga korpus
  - Kannatab -40 ... +125 °C
- Ca 25 USD

Devices	
CAN 2.0b controllers	2
General-Purpose Timers (GPT)	5
Watch-Dog Timer (WDT)	1
Pulse-Width Modulator (PWM)	1
Analog-Digital Converter Channels	24*8
USART	1
Synchronous Serial Comms (SSC)	1
Capture/Compare Channels	2x16
External Ports	
CAN interfaces	2
8-bit ports from devices	8
16-bit ports from devices	1
Memory	
ROM	32 kB
Fast General Internal RAM (IRAM)	3 kB
Extension Internal RAM (XRAM)	8 kB

42

## Operaatori paneel

- Embedded PC
  - Graafiline ekraan
  - Puutetundlik ekraan
  - Joystick
  - Nupud
  - Klaviatuur
- Kuid piisavalt keeruline, et "ära eksida"



© Jakob Engblom

43

## Ülisuured süsteemid

- Funktsioonid, mis nõuavad arvuteid:
  - Radar
  - Relvasüsteemid
  - Kahjustuste kontroll
  - Navigatsioon
  - sisuliselt kõik
- Arvutid:
  - Suured serverid
  - Tuhandeid protsessorid



© Jakob Engblom

44

## Aine ülevaade

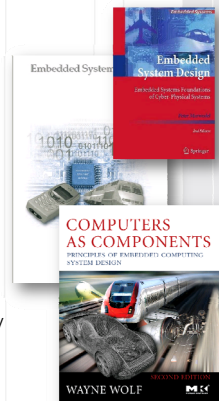
## Teemad

- Sissejuhatus
- Arvutusmudelid
- Protsessorite arhitektuurid
- Arvutite arhitektuurid, mälusüsteemi hierarhia
- Arvutusprotsesside organiseerimine
- Sardtarkvara
- RTOS ja planeerimine
- Võimsus- ja energiatarbe optimeerimine
- Töökindlus

46

## Kirjandus

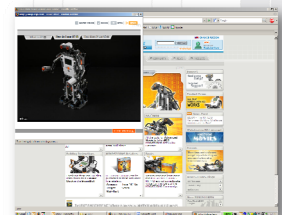
- Soovituslik lugemine:
  - Embedded System Design, Peter Marwedel, Springer, 2006 (1st ed.) 2011 (2nd ed.)
  - Computers as Components. Principles of Embedded Computing System Design, Wayne Wolf, Morgan Kaufmann/ Elsevier Publishers, 2005.



47

## Võimalus ise katsetada

- Lego mindstorms robotics kit
  - Standartne kontrollor
    - 8-bit protsessor
    - 64 kB mälu
  - Mootorite ja sensorite elektrooniline kontroll
- Hea moodus sardsüsteemide õppimiseks



© Jakob Engblom

48

## Sardsüsteemide üldisloomustus



Graphics: © Alessandro Noldi, Corbis, Microsoft, 2003

© Gert Jervan

## Sardsüsteemide üldisloomustus

- Peab olema **usaldusväärne** (dependable),
  - **Töökindlus** (Reliability)  $R(t)$  = tõenäosus, et süsteem töötab korralikult kui ta töötas ajahetkel  $t=0$
  - **Remonditavus** (Maintainability)  $M(d)$  = tõenäosus, et süsteem töötab taas korralikult  $d$  ajahüki peale vea esinemist.
  - **Töövalmidus** (Availability)  $A(t)$ : Tõenäosus, et süsteem töötab ajahetkel  $t$
  - **Ohutus** (Safety): Süsteem ei põhjusta kahju.
  - **Turvalisus** (Security): Konfidentsiaalne ja usaldatav kommunikatsioon
- Isegi ideaalselt loodud süsteemid võivad läbi kukkuda, kui eeldused süsteemi töökoormuse võimalike vigade kohta on valed. Süsteeme ei saa teha usaldusväärseks tagantjäreli, vaid sellega tuleb arvestada süsteemi loomise algusest alates.

© Gert Jervan

50

## Sardsüsteemide üldisloomustus

- Peavad olema efektiivsed
  - Energiaefektiivsus
  - Koodi efektiivsus (eriti kiipsüsteemide puhul)
  - Töö efektiivsus
  - Dimensiooniline efektiivsus
  - Maksumuse efektiivsus
- Loodud mingi spetsiifilise ülesande jaoks. Teadmine süsteemi käitumisest süsteemi loomise ajal aitab vähendada süsteemi loomisele kuluvaid ressursse ning suurendada stabiilsust
- Spetsiaalne kasutajaliides (mitte aga hiir, klaviatuur ja ekraan)

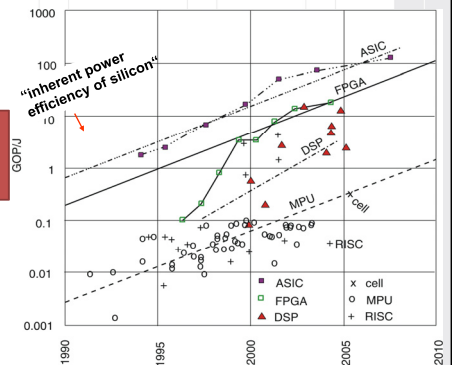
© Gert Jervan

51

## Energiaefektiivsuse tähtsus

Efficient software design needed, otherwise, the price for software flexibility cannot be paid.

© Hugo De Man, IMEC, Philips, 2007



© Gert Jervan

## Sardsüsteemide üldisloomustus

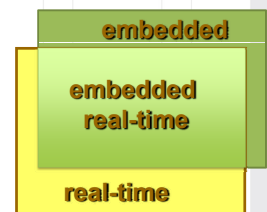
- Mitmed sardsüsteemid peavad vastama reaalaaja nõudmistele
  - Reaalaaja süsteem peab reageerima väliste stiimulitele mingi keskkonna poolt etteantud ajahüki jooksul
  - Reaalajasüsteemides on õiged vastused, mis hilinevad, valed vastused.
  - „Reaalaaja piirangut nimetatakse tugevaks, kui selle mitte saavutamine viib katastroofini“ [Kopetz, 1997].
  - Kõiki ülejäänud reaalaaja piiranguid nimetatakse pehmeteks.

© Gert Jervan

53

## Reaalaaja süsteemid

- Kas reaalaaja ja sardsüsteemid on sünonüümid?
  - Enamus sardsüsteeme on reaalajasüsteemid
  - Enamus reaalaaja-süsteeme on sardsüsteemid



© Jakob Engblom

© Gert Jervan

54

## Sardsüsteemide üldiseloomustus

- Tihti ühendatud välismaailmaga läbi sensorite ja täiturite
- Hübridsüsteemid (analoog + digitaal).
- Tüüpilised sardsüsteemid on reageerivad süsteemid: „Reageeriv süsteem on see, mis pidevalt suhtleb oma keskkonnaga ja töötab keskkonna poolt dikteeritud tempoga“ [Bergé, 1995]  
Käitumine sõltub sisenditest ja praegusest olekust.  
☞ sobivad automaadi mudelid.

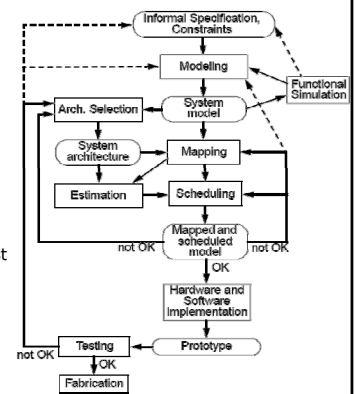
## Sardtarkvara väljakutsed

- Dünaamilised keskkonnad
- Ettenähtud funktsionaalsuse hõivamine!
- Valideeri spetsifikatsioon!
- Efektne implementeerimine
- Kuidas tagada reaalaja nõudmised?
- Kuidas valideerida reaalaja sardtarkvara? (suured andmemahud, testimine võib olla ohutus-kriitiline)

## Disainivoog ja selle muutused

## Disainivoog

- Arhitektuuri valik
- Ressursside sidumine (mapping)
- Planeerimine (scheduling)
- Palju simuleerimist ja emuleerimist
- Design space exploration



## Maailm on paralleelne...

- Threads!
  - Probleemid:
    - Absoluutselt mittedeterministlikud
    - Täitmisjärjekord tuleb jõuga paika panna (näiteks mutex-itega)
  - "... threads as a concurrency model are a poor match for embedded systems. ... they work well only ... where best-effort scheduling policies are sufficient."

Ed Lee: Absolutely Positively on Time, IEEE Computer, July, 2005

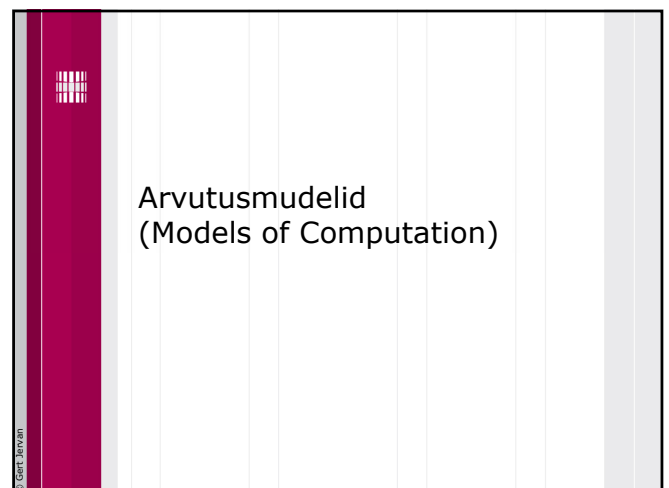
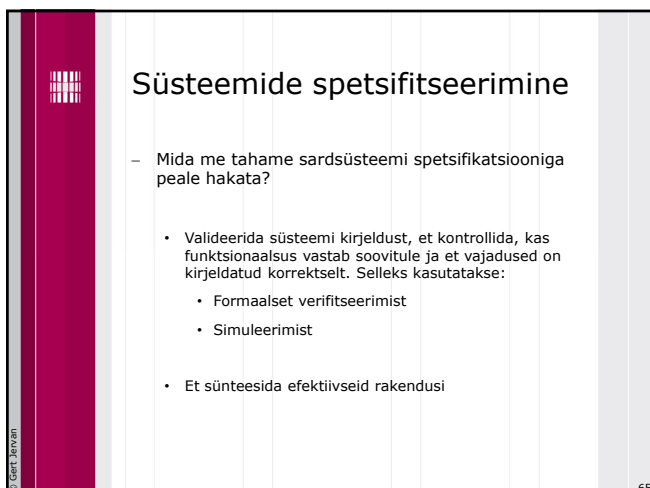
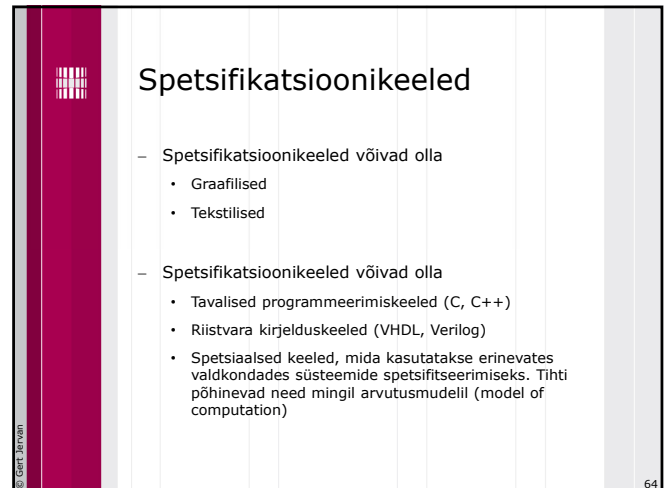
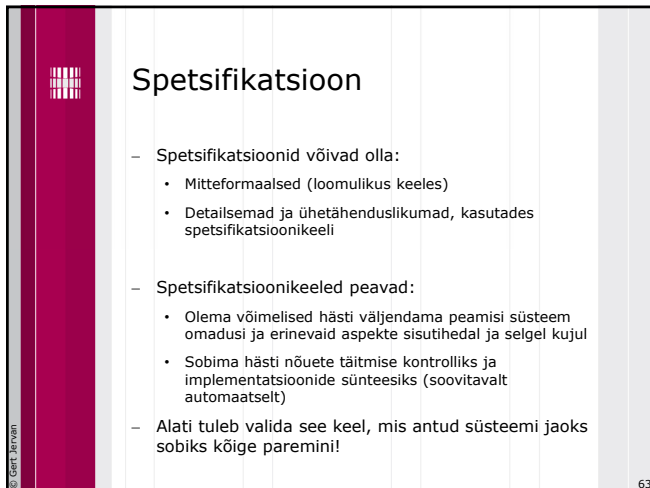
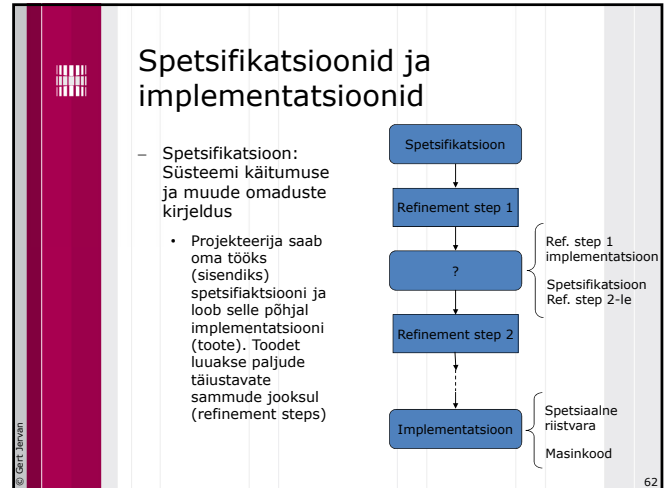
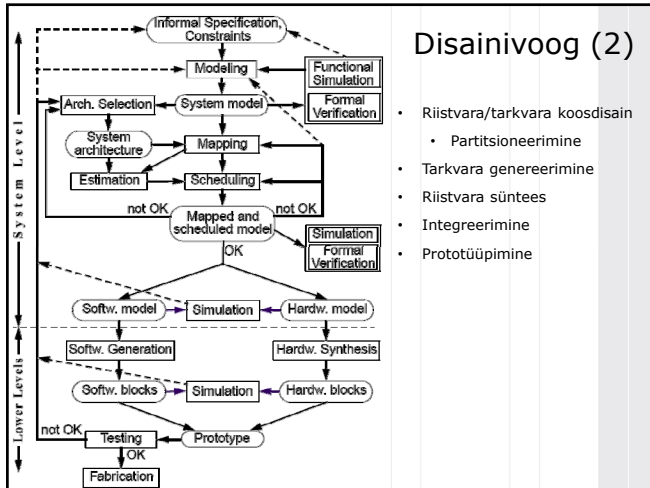
## Von Neuman'i mudel on surnud!

- "The lack of timing in the core abstraction is a flaw, from the perspective of embedded software, ..."
- "Timing is everything"
- What is needed is nearly a reinvention of computer science.

Ed Lee: Absolutely Positively on Time, IEEE Computer, July, 2005

Frank Vahid, WESE 2008

Ed Lee: Absolutely Positively on Time, IEEE Computer, July, 2005



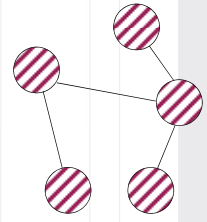
## Arvutusmodelid

- Arvutusmodelid (models of computation) käsitlevad keele täitmismudeli (execution model) loomiseks vajalikke teoreetiliste valikute kogumeid
  - Disain on esitatud kui komponentide kogum, mida võib vaadelda kui isoleeritud monoliitseid mooduleid (tihti kutsutakse neid protsessideks – processes või ülesanneteks – tasks), mis suhtlevad omavahel ja ümbruskonnaga
  - Arvutusmodel defineerib nende modulite käitumise ning omavahelise suhtlemise

67

## Arvutusmodelid (2)

- Arvutusmodelid esitavad:
  - Kuidas iga moodul (protsess või ülesanne) teostab oma sisemisi arvutusi
  - Kuidas moodulid vahetavad omavahel informatsiooni
  - Kuidas nad seostuvad omavahel kattuvuse mõistes
- Mõningad arvutusmodelid ei kajasta moodulite sisemust, vaid üksnes nende suhtlemist ja kattuvust

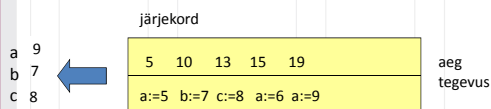


68

## Komponendid

- Von Neumann'i mudel  
Järjestikuline täitmine

- Diskreetsed sündmused



69

## Komponendid (2)

- Automaadid



- ✓ Differentsiaalvõrrandid

$$\frac{\partial^2 x}{\partial t^2} = b$$



70

## Kattuvus (Concurrency)

- Süsteemid koosnevad tegevuste (protsessid või ülesanded) kogumist. Neid tegevusi võib potentsiaalselt täita paralleelselt, ehk teisisõnu: nad on kattuvad (concurrent).
- Kuidas väljendada kattuvust?  
See on üks aspekt, milles arvutusmodelid erinevad
  - Andmete-põhine kattuvus
  - Kontrolli-põhine kattuvus

71

## Andmete-põhine kattuvus

- Süsteem on kirjeldatud kui protsesside kogum ilma määratlemata täitmisjärjekorraga



- Protsesside täitmise järjekord (ja selle põhjal võib kaudselt teha järeldusi paralleelsuse kohta) on fikseeritud ainult andmete sõltuvuse põhjal
  - Väga tüüpiline paljudes DSP rakendustes

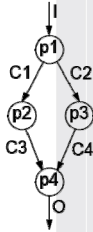
72

## Andmete-põhine kattuvus (2)

```

Process p1( in int a, out int x, out int y) {
.....
}
Process p2( in int a, out int x) {
.....
}
Process p3( in int a, out int x) {
.....
}
Process p4( in int a, in int b, out int x) {
.....
}
channel int I, O, C1, C2, C3, C4;
p1(I, C1, C2);
p2(C1, C3);
p3(C2, C4);
p4(C3, C4, O);

```



Ei ole oluline, mis järjekorras me need kirjutame

73

## Kontrolli-põhine kattuvus

- Protsesside täitmise järjekord on üheselt kirjeldatud süsteemi spetsifikatsioonis
- Kasutatakse spetsiaalseid konstruktsioone et määrata ära täitmise järjekord ja kattuvus

74

## Kontrolli-põhine kattuvus (2)

```

module p1:
.....
end module
module p2:
.....
end module
module p3:
.....
end module
module p4:
.....
end module
run p1;
[run p2 || run p3];
run p4;

```

- See näide on kirjutatud ESTERELis
- Protsess p1 algab esimesena ja peab lõppema enne kui p2 ja p3 algavad
- p2 ja p3 algavad paralleelselt
- p2 ja p3 peavad mõlemad lõppema, enne kui p4 saab alustada

Siin on kirjutamise järjekord esmaoluline

77

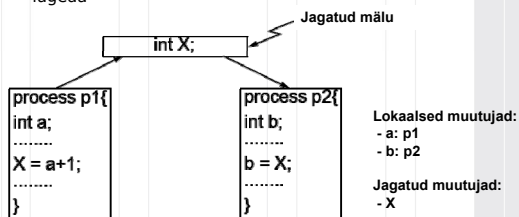
## Kommunikatsioon

- Protsessid peavad info vahetuseks kommunikeeruma
- Erinevad arvutusmudelid kasutavad erinevaid arvutusmudeleid
- Jagatud mälu (shared memory)
- Sõnumite edastamine (message passing)
  - Blokeeriv
  - Mitte-blokeeriv

76

## Jagatud mälu

- Iga saatev protsess kirjutab jagatud muutujatesse, mida saavad omakorda vastuvõtavad protsessid lugeda



77

## Jagatud mälu (2)

- Võivad tekkida võidujooksud (race conditions) ☞ tagajärjeks vastuolulised andmed ☞ Kriitilised sektsioonid = sektsioonid, kus ressursile (näiteks jagatud mälu) tuleb garanteerida ainuõiguslik ligipääs

```

process a {
..
P(S) //lukustamine
.. //kriitiline sektsioon
V(S) //luku vabastamine
}
process b {
..
P(S) //lukustamine
.. //kriitiline sektsioon
V(S) //luku vabastamine
}

```

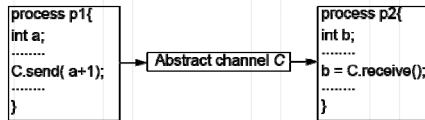
Ilma võidujooksuta ligipääs jagatud mälule, mida kaitseb lukk S

- Seda mudelit kasutavad:
  - Kriitiliste sektsioonide vastastikune välistamine
  - Cache coherency (jagatud vahemälu) protokollid

78

## Sõnumite edastamine

- Andmed edastatakse üle abstraktse kommunikatsioonikeskkonna, mida nimetatakse kanaliks



- See kommunikatsioonimudel on piisav kirjeldamaks hajussüsteeme (distributed systems)

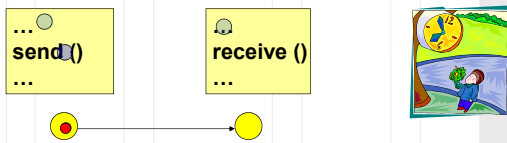
## Mitteblokeeriv (asünkroonne)

- Saatja ei pea ootama kuni sõnum on kohale jõudnud; Potentsiaalne probleem: Puhvri täitumine



## Blokeeriv sõnumite edastamine - rendez-vous

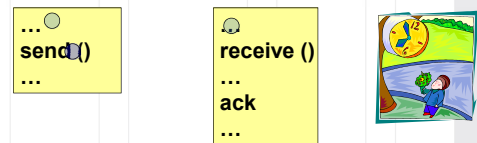
- Saatja ootab kuni vastuvõtja on sõnumi kätte saanud



- Protsess, mis suhtleb, blokeerib end (suspends), kuni teine protsess on valmis andmete ülekandeks
- Need kaks protsess peavad ennast enne andmete ülekannet sünkroniseerima

## Laiendatud rendez-vous

- Vastuvõttev pool peab saatma spetsiaalse teate kättesaamise kohta. Vastu võttev pool võib teostada enne teate saatmist andmete kontrolli.



## Sünkroniseerimine

- Sünkroniseerimist ei saa eraldada kommunikatsioonist
  - Iga protsesside vaheline suhtlemine eeldab mõningast kommunikatsiooni ja sünkroniseerimist
- Sünkroniseerimine: Üks protsess on seisatud (suspended) kuni teise täitmine jõuab mingi punktini
  - Kontrolli-põhine sünkroniseerimine
  - Andmete põhine sünkroniseerimine

## Kontrolli-põhine sünkroniseerimine

```
module p1:
  .....
end module
```

```
module p2:
  .....
end module
```

```
module p3:
  .....
end module
```

```
module p4:
  .....
end module
```

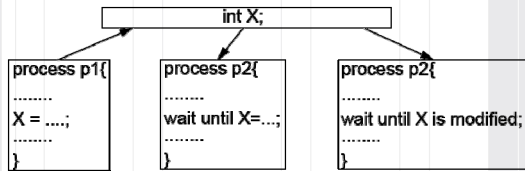
```
run p1;
[run p2 || run 3];
run p4
```

- Kontrolli-põhine sünkroniseerimise puhul tegeleb sünkroniseerimisega kontrollstruktuur

- Siin on mitmeid sünkroniseerimise punkte:
  - peale p1 lõpetamist ja enne p2, p3 algust
  - Peale p2 ja p3 lõppemist ning enne p4 algust

## Andmete põhine sünkroniseerimine

- Kommunikatsioonimehhanismid väljendavad kaudselt ka sünkroniseerimist
- Jagatud mälu põhine sünkroniseerimine



85

## Andmete põhine sünkroniseerimine (2)

- Sõnumite edastamise põhine sünkroniseerimine
  - Kommunikatsiooni blokeerimine sõnumitega tähendab automaatselt saatja ja vastuvõtja vahelist sünkroniseerimist

86

## Tüüpilised arvutusmodelid

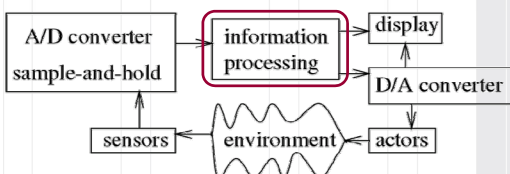
- Olekudiagrammid (StateCharts)
- Andmevoo (dataflow) mudelid
- Petri võrgud (Petri Nets)
- Diskreetsed sündmused (Discrete events)
- (Sünkroonsed) lõplikud olekumasinad (Finite State Machines)
- Sünkroonsed/reaktiivsed keeled
- Koosdisaini lõplikud olekumasinad
- Timed Automata

87

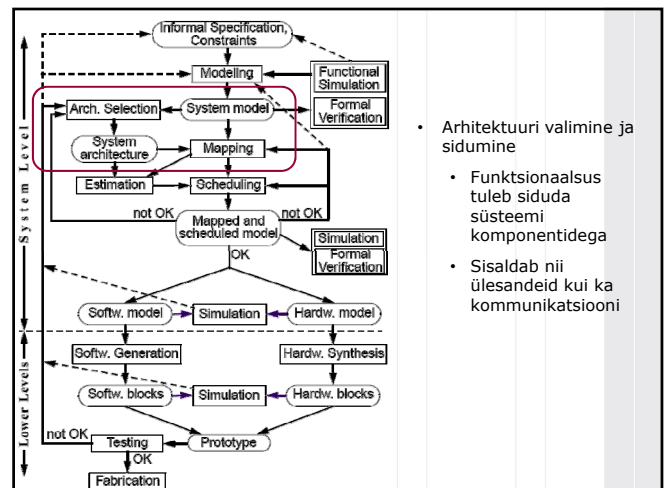
## Arhitektuurid ja platvormid

## Sardsüsteemide riistvara

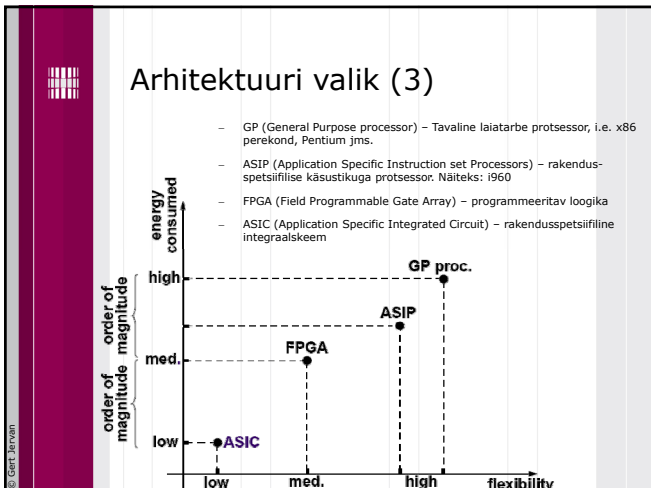
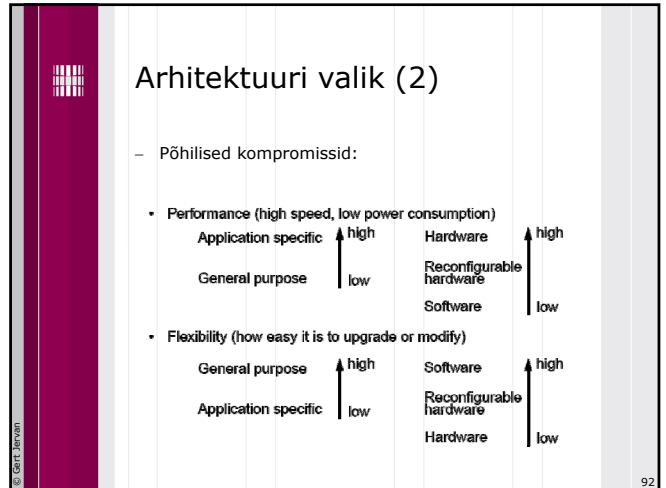
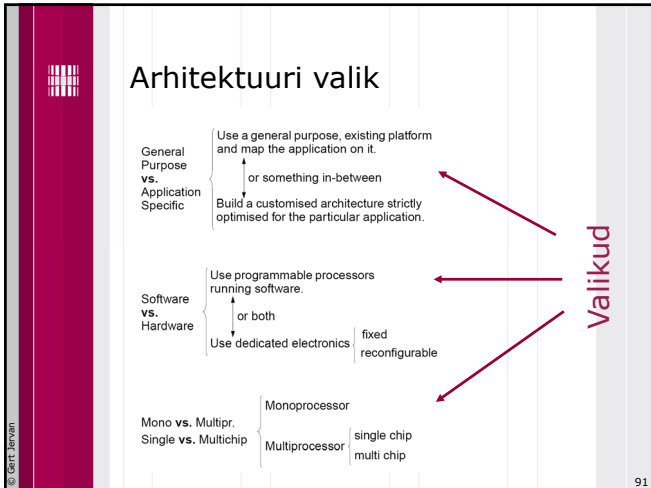
- Sardsüsteemide riistvara kasutatakse tihti tsüklis:



89



- Arhitektuuri valimine ja sidumine
- Funktsionaalsus tuleb siduda süsteemi komponentidega
- Sisaldab nii ülesandeid kui ka kommunikatsiooni



## Energia- ja võimsustarve

© Gert Jervan

## Miks on energiatarve nii oluline?

- Kaasaskantavad süsteemid – aku eluiga!
- Süsteemid väga limiteeritud energiaeelarvega: Mars Pathfinder, UAV
- Desktopid ja severid: väga suur võimustarve
  - Tõstab temperatuuri ning vähendab jõudlust ning usaldusväärsust
  - Tõstab vajadust kallite jahutusmehhanismide järele
- Üks kõrge jõudlusega kiipide loomise põhikastusi on kuumuse eemaldamine
- Suur võimsustarve toob kaasa ka majanduslikud ja keskkonna-alased probleemid

© Gert Jervan 95

## Sard-OS, vahevara, planeerimine (Embedded OS, middleware, scheduling)

Gert Jervan

© Gert Jervan

## Reaalaja süsteemid

- Enamus sardsüsteeme on reaalaja süsteemid
  - Aeg:
    - Süsteemi korrektsus ei sõltu mitte ainult tulemuste loogilisest korrektsusest vaid ka ajast, millal need tulemused on saadud
  - Reaal-:
    - Reaktsioon välistele sündmustele peab toimuma samal ajal sündmusega. Süsteemi aeg peab olema mõeldav samades ühikutes kui keskkonna aeg
- Näited:
  - Kontrollisüsteemid, tööstussüsteemid, lennundus, autonodus, meditsiin, tuumaenergia, militaar, telekommunikatsioon, multimeedia, ...

97

## Reaalaja süsteemid – tüüpilised omadused

- Nad on aja-kriitilised
  - Ajaliste piirangute mittejärgimine võib vähendada teenuste kättesaadavust või viia katastroofiliste tagajärgedeni
- Sisaldavad mitmeid paralleelselt täidetavaid ülesandeid
  - Ülesanded jagavad ühiseid ressursse, nagu näiteks protsessor, kommunikatsioonikanalid. Suhtlevad omavahel. Seetõttu on üheks peamiseks probleemiks ülesannete planeerimine
- Töökindlus ning veakindlus on esmatähtsad
  - Palju on ohutus-kriitilisi rakendusi

98

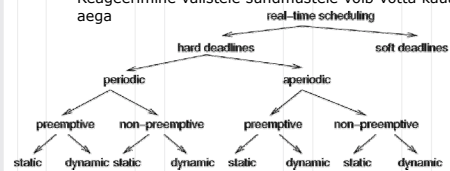
## Nõrgad ja ranged reaalaja süsteemid

- Ajalised piirangud on tüüpiliselt esitatud piir-aegadega (deadline), mis määravad ära aja, millal ülesande (task) täitmine peab lõppema.
- Ülesandele seatav piir-aeg võib olla:
  - Range (hard deadline): tuleb täielikult ja alati saavutada. Mittesaavutamine võib tuua katastroofilised tagajärjed
    - Garanteerida eelnevalt ja off-line
  - Nõrk (soft deadline): ülesanne võib lõppeda peale sellele ette nähtud piir-aega, kuid tulemuse väärtus võib aja jooksul väheneda
  - Kindel (firm deadline): sarnane rangel, kuid ei järgne katastroofilisi tagajärgi. Tulemus ei oma peale piir-aega mingit väärtust

99

## Planeerimisalgoritmide klassifikatsioon

- Katkestavad (preemptive) planeerijad: kasutatakse, kui
  - Mõned ülesanded on pikkade täitmisaegadega, või
  - Reageerimine välissündmustele peab olema lühike
- Mitte-katkestavad (non-preemptive) planeerijad:
  - Kõik ülesanded töötavad, kuni on lõpetanud. Reageerimine välistele sündmustele võib võtta kaua aega



100

## Töövahendid

- Madalamatel tasemetel on saadaval palju:
  - Koodi generaatorid, kompilaatorid, testide generaatorid ja debuggerid, simulaatorid, emulaatorid, sünteesivahendid
- Kõrgemal tasemel paljud töövahendid puuduvad ja sinna on koondunud tänapäevase CAD teadustöö teravik
  - Saadaval mitmeid akadeemilisi vahendeid

101

## Küsimusi?

Gert Jervan  
ati.ttu.ee/~gerje

© Gert Jervan