

## Luento 10: Tosiakakäyttöjärjestelmät

*Lähteet:* Liu: Real-Time Systems, luku 12,  
Stankovic & Rajkumar: Real-Time Operating Systems. Real-Time  
Systems, 28,237-253,2004.  
Lisäksi kalvosarjassa on kuvia myös kirjasta:  
Cooling: Software Engineering for Real-time Systems, luku 6

## Sisältö

- n Yleistä
  - n Ominaisuuksia
  - n Arkkitehtuuriratkaisuja
  - n Piirteitä
  - n Kehitysympäristö ja -työkaluja
- n POSIX
- n RTOS esimerkkejä

## Ominaisuuksia

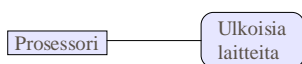
- n Ennustettavuus ja suorituskyky
- n Pieni yleiskuorma
- n Aikapalvelut ja niiden tarkkuus
- n Konfiguroitavuus
- n Työkalut (kääntäjät, GUI)
- n Siirrettävyys (sekä KJ että sovellus)

## Arkkitehtuuriratkaisuja

- n Ei yhtä standardiratkaisua
- n Tarpeet ja ratkaisut vaihtelevat yksinkertaisesta sulautetusta järjestelmästä hajautettuun yleiskäyttöiseen järjestelmään
- n Käytettävät työkalut vaihtelevat ratkaisujen mukana

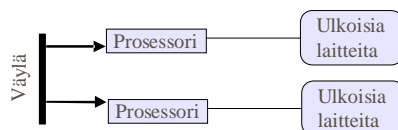
## Sulautettu järjestelmä: Yksi ohjaava prosessori

- n Kaikki toiminnallisuus tiedossa ennakkoon
- n Usein ei erillistä käyttöjärjestelmää lainkaan
- n Ohjelmakehitys muualla, esim. simuloidussa ympäristössä



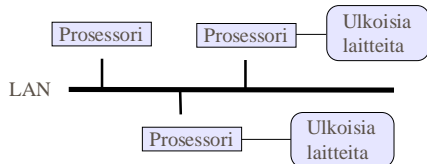
## Sulautettu järjestelmä: Useita ohjaavia prosessoreja

- n Kaikki toiminnallisuus tiedossa ennakkoon
- n Usein ei erillistä käyttöjärjestelmää lainkaan
- n Prosessorien välinen kommunikointi
- n Ohjelmakehitys muualla

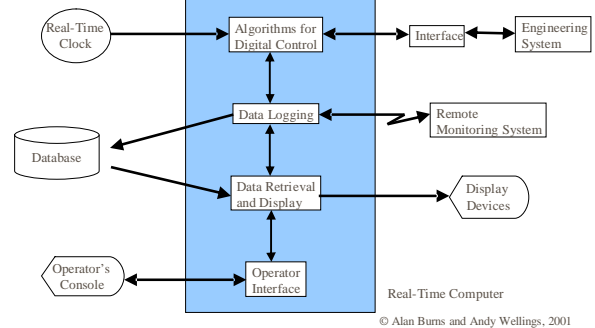


## Sulautettu ja hajautettu järjestelmä: Useita prosessoreja, yksi verkko

- Prosessorien välinen kommunikointi
- Usein käytössä jokin RTOS (tosiaikakj)
- Ohjelmakehitys suoraan järjestelmässä tai muualla



## Ohjausjärjestelmä: yleiskuva kuvattava halutulle toteutusalustalle



## Vain ohjain (Controller) - ei kjtä

- Esimerkiksi keskeytyksensittelyn avulla
- Kaksi vaihetta
  - Käynnistys (initialisation)
  - Toiminta
    - Sovelluksen koodi keskeytyksensittelijöissä
    - Aktivointi esim. kellokeskeytyksillä
    - Usein tausta-ajona silmukka, joka ei tee mitään tai kerää alhaisella prioriteetilla erilaista tilastotietoa

## Malli (vain ohjain)

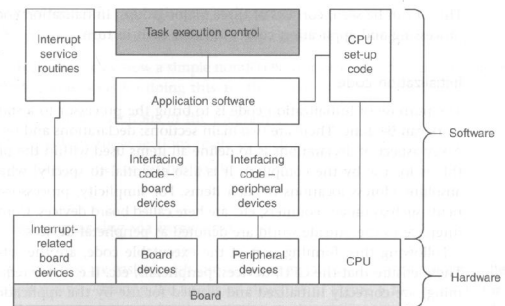


Figure 6.3 Conceptual model - simple single-processor hardware-software structure.

## Nanoydin (Nanokernel) - real-time executive

- Hyvin minimalistinen käyttöjärjestelmä, vain
  - Tehtävän (task) luonti
  - Skedulointi ja prosessinvaihto
  - Ajoitus ja keskeytykset
- Kaikkein aikakriittisimmät tehtävät keskeytyksensittelillä
- HAL - Hardware Abstraction Layer

## Malli - nanoydin

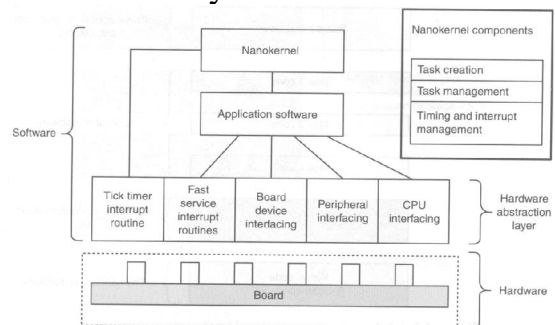


Figure 6.4 Conceptual model - nanokernel-based system.

## Mikroydin

- n Palveluja
  - n Järjestelmän käynnistys ja lataus
  - n Prosessin (tehtävän) skedulointi ja valvonta
  - n Poissulkeminen ja synkronointi
  - n Tiedonsiirto synkronoinnilla ja ilman
  - n Muistinhallinta
- n BSP – Board Support Package

## Mikroydin

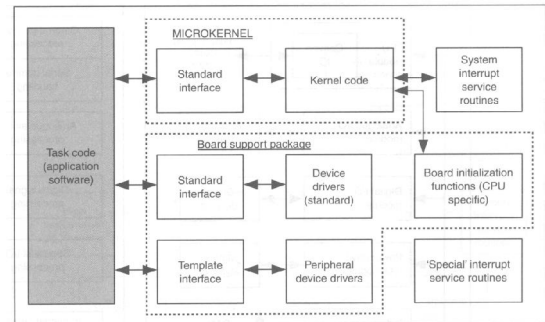


Figure 6.7 Software conceptual model – small microkernel-based system.

## Mikroydin

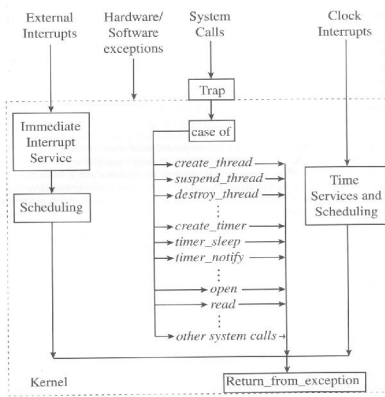


FIGURE 12-2 Structure of a microkernel.

## Yleiskäyttöinen sulautettu RTOS

- n Lisänä tavanomaiset kj-palvelut
  - n Verkkotuki ja tietoliikenneprotokollia
  - n Pysyvää muistia (esim. levy)
  - n Graafinen käyttöliittymä
  - n Prosessien ja muistin suojaus

## Yleiskäyttöinen RTOS

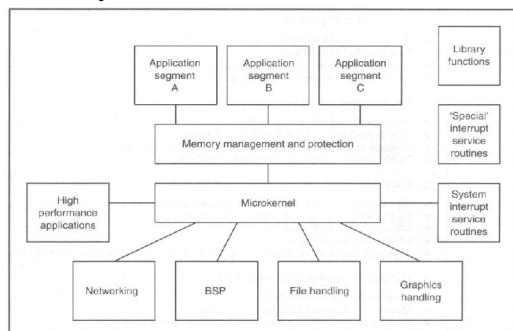


Figure 6.9 Software conceptual model – typical large general-purpose system.

## Yleiskäyttöinen RTOS

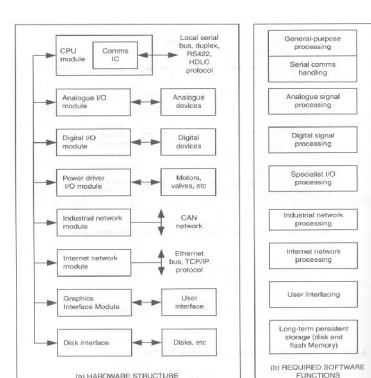
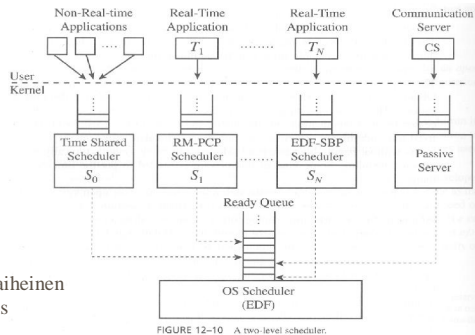


Figure 6.8 Typical large general-purpose embedded system – hardware structure and software functions.

## Sekakuorma: sekä tosiaikaisia että ei-tosiaikaisia samalla koneella



## Sisältö

- n Yleistä
  - n Ominaisuuksia
  - n Arkkitehtuuriratkaisuja
  - n **Piirteitä**
  - n Kehitysympäristö ja -työkaluja
- n POSIX
- n RTOS esimerkkejä

## Piirteitä

- n Tietoliikenne
  - n Oli aiemmalla luennolla
- n Pysyvä tiedon tallennus (levy)
  - n Ei yleensä mitään poikkeavaa tavanomaisesta
- n Prosessien hallinta ja skedulointi
  - n Ajoitusmenetelmät oli aiemmalla luennolla
- n Graafinen käyttöliittymä
  - n Tosiaikainen (????)

## Prosessien (?) hallinta

- n Eri järjestelmät käyttävät hiukan eri termejä
  - n Tehtävät (task)
  - n Prosessit (process)
  - n Säikeet (threads)
  - n Aktorit (actors)
- n Useimmiten käsitellään säikeitä, joten niin tässäkin

## Säietyyppit

- n Jaksollinen säi (periodic thread)
  - n Tuetaan esim. RT-Mach, EPIQ
  - n Säiettä ei luoda aina uudelleen, vaan
  - n Kunkin jakson alussa säie alustetaan uudelleen ja se 'nukkuu' jaksojen välit
- n Jaksottomat ja sporadiset säikeet
  - n Nukkuvat myös toimintavälit, alustetaan uudelleen usein ennen nukahtamista
  - n Käynnistyvät usein ulkoisten tapahtumien perusteella

## Jaksollistuvan palvelimen toteutus

- n Esim. Sporadiset ja "Vakio käyttöaste" palvelimet (luento 3)
- n Voidaan toteuttaa yhdellä säikeellä
- n Kukin erillinen sporadinen tai jaksoton työ muodostaa oman funktionsa säikeen sisälle
- n Säikeellä on funktiotaulu, jossa on kunkin työn funktion osoite (kts. esim. C-kurssi)
- n Ulkoinen herätykseen käynnistää tietyn työn, jolloin säi käynnistyy ja valitsee suoritettavaksi ko. funktion
- n Töiden vuorotus palvelimen sisällä tapahtuu yksikertaisesti vaihtamalla suoritettavaa funktiota.

## Säikeen tilat

- n Nukkuu (sleeping)
- n Valmis (ready)
- n Suorituksessa (executing)
- n Estetty (suspended, blocked)
- n Päättynyt (terminated)

## Aikapalvelut (perusmalli)

- n Kellokeskeytykset (clock interrupt)
- n Joka keskeytyksessä käydään läpi
  - n Prosessien ajastintapahtumat (timer events)
  - n Suoritusbudjetin (execution budget) keruu
  - n Valmisjonon päivitys ja kontrollin siirto
- n 10 millisekunnin välein (tyypillinen)

## Keskeytykset

- n Yleensä kahdessa vaiheessa
- n Välitön käsittely
  - n Hyvin lyhyt kesto
- n Ajoitettu käsittely
  - n Prioriteetin perusteella

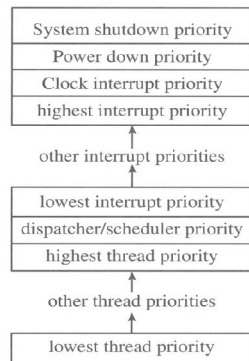


FIGURE 12-3 Hardware and software interrupt priorities.

## Graafinen käyttöliittymä

- n Näitä todella on tosiaikajärjestelmissäkin
  - n Lentäjien erilaiset valvontanäytöt
  - n Valvontahuoneiden näytöt
  - n Web-palvelin esim. tietoliikennekytkimessä
- n Tarve laajenee
  - n Puhelimet
  - n PDA:t
  - n Autojen navigointijärjestelmät
  - n ...

## Sisältö

- n Yleistä
- n **POSIX**
  - n **Portable Operating System Interface**
- n RTOS esimerkkejä

## POSIX - standardi 1003.1.-2001

- n Säikeet
- n Skedulointirajapinta
- n Kello ja ajastin
- n Prosessien välinen kommunikointi
- n Synkronointi
- n Jaettu muisti ja lukitus
- n Synkroninen ja asynkroninen I/O
- n Tosi aikaiset signaalit

## Kello ja ajastin

- n Perinteiset kellot
  - n *time*
  - n *gettimeofday*
  - n *Aika alkoi 1.1.1970 ja päättyy helmik. 2038*
- n Perinteiset aikamuutoksia
  - n *sleep*
  - n *setitimer*
  - n *getitimer*

## Kello ja ajastin: laajennokset

- n Useita kelloja: ainakin `CLOCK_REALTIME`
  - n *clock\_settime(clock\_id)*
  - n *clock\_gettime, clock\_getres*
- n Ajastimet
  - n *timer\_create, timer\_delete*
  - n *timer\_settime, timer\_gettime*
  - n *timer\_getoverrun*
- n *nanosleep*

## Sisältö

- n Yleistä
- n POSIX
- n **RTOS esimerkkejä**
  - n **VxWorks + Tornado**
  - n **QNX**
  - n **Linuxin tosiaikamuunnelmat**

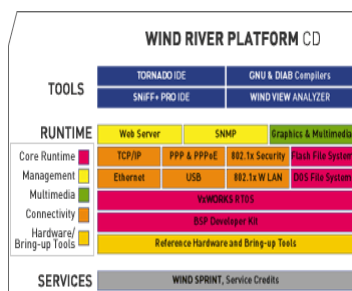
## VxWorks 5.5 + Tornado 2.2

- n [www.windriver.com](http://www.windriver.com)
- n Kehuu asiakkailaan, mm.
  - n Honda: ASIMO humanoid robot
  - n JPL: Pathfinder ja Rover



## Wind River tuotteita

- n Käyttäjärj.
- n Työkalut
- n Laitteisto-  
sovitus
- n Verkko-  
yhteys



## VxWorks tosiaikakäyttäjärj.

- n *wind mikrokernel*
- n *Verkko- ja IO-tuki*
- n *Kohdearkkitehtuureja*
  - n *Motorola 68K, ColdFire, IBM PowerPC*
  - n *Intel Pentiums, StrongArm, Xscale*
  - n *MIPS*
  - n *ARM*

## Tornado kehitysympäristö

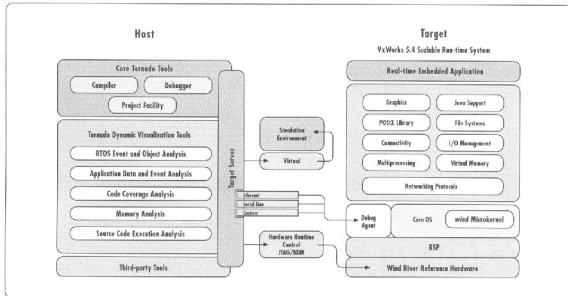


Figure 5. Tornado II Development Environment

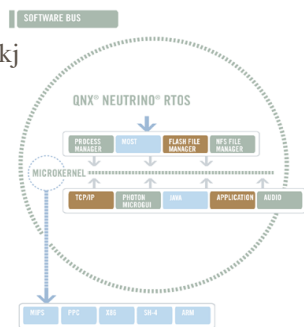
## QNX

- www.qnx.com
- Vastaavasti 'success stories'
  - Inform@fon, Multiphone
  - Nukutusmonitori



## QNX

- Neutrino tosiaikakaj
  - Mikrokemeli
  - Verkkotuki
- Photon microGUI
- Alustoja
  - MIPS, PPC,
  - X86, ARM

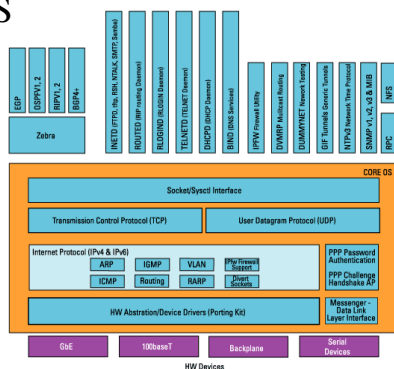


## LynxOS

- LynuxWorks (www.lynx.com)
- POSIX conformance tested
- BlueCat Linux / RT
- ISO 9001 sertifioitu
- HP:n kirjoittimissa
- PhatNoise – jukeboxi autoon



## LynxOS



## Linux ja tosiaikaisuus

- Kolme erilaista lähestymistapaa:
  - Yhteensopiva ydin:
    - Täysin erillinen toteutus, josta on tehty binääriyhteensopiva (LynxOS)
  - Ytimen muokkaus:
    - Tehdään erityisiä paikkauspaketteja (patch), joilla muutetaan ytimen rakennetta paremmin tosiaikaisuutta tukemaan (esim. aiempien ytimien low-latency patch)
  - Kaksi ydintä:
    - Laitetaan varsinainen linux-ydin yhdeksi tosiaikaprozessiksi tosiaikaisen mikroytimen päälle (RTLinux)

## RTLinux

- n Victor Yodaikenin idea, patenti ja nykyisin myös yritys (<http://www.fsmlabs.com>) tämän ympärillä
- n Perus-Linuxin toiminnasta on vaihdettu vain keskeytyskäsitteitä.
- n Se suorittaa tuon tosiaikaytimen ja vuorottaa tosiaikaiset prosessit. Niitä vapaaksi jäävät osat annetaan perus-Linuxille ja sen prosesseille.

## RTLinux – tosiaikaprosessi

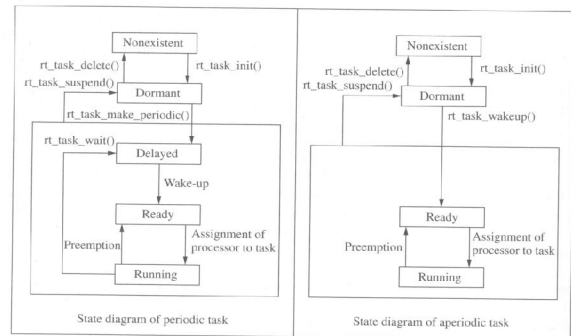


Figure 8.2 State diagram of task

## Tosiaikakäyttöjärjestelmät

- n Open source – Linux modifikaatiot
  - n Ei lisenssimaksuja, edullinen hankintahinta
  - n Koviin aikarajojen tuki ?
  - n Kehitysympäristöt ?
- n Kaupalliset käyttöjärjestelmät
  - n Kallis hankintahinta + lisenssi per myynti
  - n Koviin aikarajojen tuki ?
  - n Tehokkaita kehitysympäristöjä
  - n Siirrettävissä (maksua vastaan) uudelle alustalle

## Yhteenveto

- n Tosiaikaisen käyttöjärjestelmän piirteet
  - n Deterministinen käyttäytyminen
  - n Lyhyet keskeytysviipet, nopea kontekstin vaihto
  - n Prioriteetteja tukeva ajoitusmekanismi
  - n Kellon tarkkuus ja hyvät ajastimet
  - n Muunnettavissa parametreja vaihtamalla erilaisiin ympäristöihin