

# Datorsystemteknik

## *för D2, ICT2, E3 och Mek3*

Nicholas Wickström

Högskolan i Halmstad  
Sverige

# Outline

- Förra föreläsningen
- **Specifikation**
  - Kravspecifikation
  - Funktionsspecifikation
  - Blockdiagram
- **Operativsystem**
  - Grunder, Definition
  - Meddelanden
  - Schemaläggning

# Kravspecifikation

- Kravspecifikation skall ge en överblick av det som skall utföras.
- Skall innehålla:
  - \* **Backgrund/Översikt.**  
Ex. Enheten är avsedd att användas i pappersindustrin för mätningar på pappersmassa. bla bla ...
  - \* **Krav.** Ex. Enheten skall tåla värme från -10 till 55 grader C. Enheten skall ha minne för 50 mätningar.

# Kravspecifikation (forts.)

- \* Tid. Ex. En prototyp skall finnas den 20:e December 2000.
- \* Mjukvara/Hårdvara - Vilka egenskaper/funktioner skall objektet ha. Vilka moduler kan identifieras.

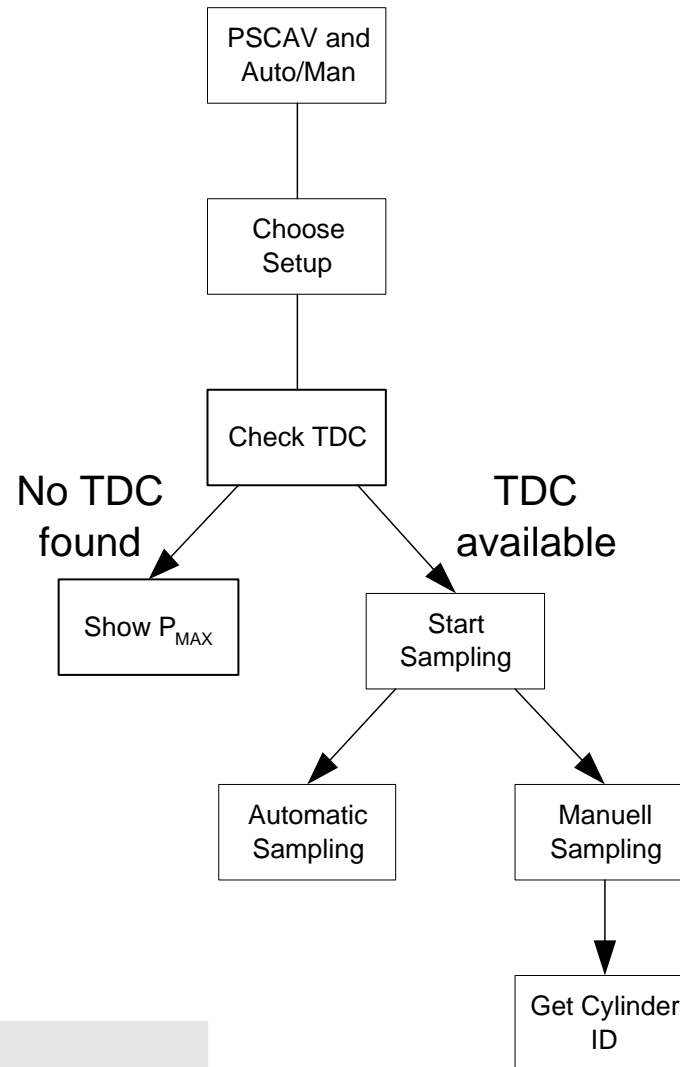
# Funktionsspecifikation

- Funktionsspecifikationen bygger på ett eller flera krav i en kravspecifikation.
- Funktionsspecifikationen kan ses som en detaljering av kravspecifikationen och skall beskriva objektets funktion och gränssnitt mot omvärlden.
- Det är viktigt att det är funktionen som beskrivs och inte dess implementering.
- Funktionsspecifikationen skall gälla för funktionen under hela livscykeln och bör därför skrivas i presens.

# Funktionsspecifikation (forts.)

- Funktionsspecifikationen har flera användare:
  - \* kravställaren (eller annan granskare) för att kontrollera att funktionen uppfyller uppställda krav.
  - \* den/de som skall implementera funktionen
  - \* den/de som skall testa funktionen
  - \* den/de som skall skriva användardokumentation
  - \* den/de som skall underhålla funktionen

# Blockdiagram



# Def. Process

Ett program som exekveras. Processen styrs och schemaläggs av operativsystemet.



# Def. Operativsystem (OS)

*(eng. operating system)* Mjukvara som styr hopp mellan processer och tillgång till gemensamma resurser (som minne, IO) samt schemaläggning.

# Def. “Multitasking”

Växelsvis utföra två eller flera uppgifter, virtuellt samtidigt, på en processor.

# Def. “Task”/Process/Tråd

Ett “task” är likt en process. “Tasket” har en egen kopia av register och en egen stack (precis som en process), men delar minne med andra task (en process har privat minne). Tråd (*eng. thread*) och “task” är samma sak.

# Tillstånd hos "task"

- “Running” - körande på processorn
- “Ready” - färdiga att köra vid ledig CPU
- “Blocked” - blockerad av någon anledning (resurs, tid, synkronisering).

# "Context Switch"

Vid en "Context Switch" byts processen som kör, "Running", ut mot en som är redo att köra, "Ready". Den som bestämmer vilken process som skall köra är schemaläggaren.

# "Context Switch" (forts.)

Spara context (save context) från körande task genom att:

- interrupt stängs av
- processorns tillstånd (register, stack) sparas undan (i taskets TCB)
- interrupt sätts på

# "Context Switch" (forts.)

- "Running" byts eventuellt här till annat task.
- Ladda context från det nya tasket genom att slå av interrupten, återställ register från den lokala stacken (i TCB) och slå på interrupten.

# Meddelandeförmedling

- Kommunikation mellan task (eller processer) kallas *inter process communication*.
- Kan användas för att skicka resultat eller annan information mellan task.
- Kan användas för "task synchronization". Dvs om task är beroende av varandra för att lösa en uppgift kan beroendet styra exekveringen genom synkronisering mellan task.
- Meddelanden synkrona alt. asynkrona.



# Schemaläggare och prioriteter, EDF

- Schemaläggaren avgör vilken task som skall köra på processorn under nästa "time slice".
- Vanligt att använda prioriteter på RTOS.
- Med begrepp om tiden, EDF, "earliest deadline first", dvs den task som skall vara färdig först skall också köra först.

# Def. Realtidssystem

Ett *realtidssystem* (eng. *real-time system*) är ett (inbyggt) datorsystem med speciella krav på "Timing". Datorsystemet har således den speciella uppgiften att leverera resultaten vid en förutbestämd tidpunkt ("deadline").