

# Applikationsspecifika Integrerade kretsar

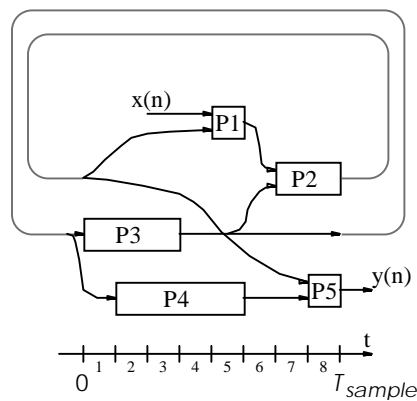
## Tentamen

TSTE81

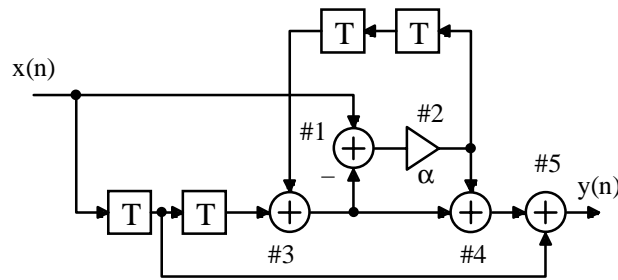
för Y4, D4 och TE

- Tid:** Torsdag 16 Mars 2000 kl. 14.00 - 18.00
- Plats:** MJ9
- Ansvarig lärare:** Kent Palmkvist, 281347, 0708917839
- Hjälpmedel:** Räknedosa, Formelsamling i Aktiva och Tidsdiskreta filter, Formelsamling i kretsteori samt allmänna tabellverk.
- Anvisningar:** För godkänd tentamen fordras ca 30 poäng.  
OBS: Poäng på muntlig deltentamen tillgodoräknas automatiskt på uppgift 1. Max poäng på uppgift 1 är 10, oberoende av hur många poäng som erhållits på muntlig deltentamen.
- Visning:** Torsdag 30 Mars 2000 kl. 13.00-14.00  
(Kent Palmkvists tjänsterum)
- Lösningar:** Anslås på Systemtekniks anslagstavla i labkorridoren
- Betygslista:** Anslås senast 2000-03-29 på anslagstavlan i labkorridoren

1. a) Ange ett nödvändigt och tillräckligt krav för att en rekursiv algoritm skall vara sekvensiellt beräkningsbar.  
*Give a necessary and sufficient condition for a recursive algorithm to be sequentially computable.* (2)
- b) Vilket går fortast att beräkna: en addition av två SDC-tal, eller extraktion av tecknet hos ett SDC-tal. Antag ordlängd  $> 10$  siffror. Motivera ditt svar.  
*What has the lowest latency: an addition of two SDC-numbers, or the extraction of the sign of a SDC number. Assume a wordlength  $> 10$  digits. Motivate your answer.* (2)
- c) Namnge två adderarstrukturer för parallell addition som har en kortare latency än en ripple-carry adder.  
*Name two adder structures for parallel addition that has a shorter latency than a ripple-carry adder.* (2)
- d) När bör man använda en exclusion graph istället för en inclusion graph vid resursallokering och tilldelning?  
*When should an exclusion graph be used instead of an inclusion graph when doing resource allocation and assignment?* (2)
- e) Vilka två typer av tillståndsmaskiner finns?  
*What two generic forms of finite state machines is there?* (2)



2. Processerna i schemuleringen ovan skall implementeras med homogena beräkningselement. Variablerna skall lagras i ett minne.  
*The processes in the schedule above will be implemented using homogenous processing elements. The variables will be stored in a memory.*
  - a) Allokeras och tilldelas beräkningselement med hjälp av cliquepartitionering.  
*Allocate and assign processing elements using clique partitioning.* (6)
  - b) Allokeras minnesceller och tilldelas variabler med hjälp av left-edge algoritmen. Beskriv kortfattat de olika stegen i algoritmen.  
*Allocate memory cells and assign variables using the left-edge algorithm. Describe shortly the different stages in the algorithm.* (6)



3. Filtret ovan skall implementeras med distribuerad aritmetik.  $\alpha = 0.125$ .  
*The filter above shall be implemented using distributed arithmetic.  $\alpha = 0.125$ .*
- Rita strukturen för hela implementationen. Använd block som ROM, skiftackumulatörer och skiftregister.  
*Draw the structure of the complete implementation. Use blocks such as ROM, shiftaccumulators, and shift registers.* (6)
  - Beräkna innehållet i ROM:en. Ange innehållet i ROM i binär form.  
*Compute the contents of the ROM(s). Describe the ROM contents using a binary representation.* (6)
4. En multiplikation med konstant koefficient och 2's komplement data skall implementeras med en seriell/parallel multiplikator. Koefficienten har värdet  $0.875_{10}$ .  
*A bitserial multiplication with a constant coefficient and data in 2's complement form is to be implemented using a serial/parallel multiplier. The coefficient has the value  $0.875_{10}$ .*
- Rita strukturen för en generell bitseriell multiplikator. Beskriv dom kontrollsignaler som behövs.  
*Draw the structure of a general bit-serial multiplier. Describe the necessary control signals.* (4)
  - Rita den förenklade strukturen om koefficienten beskrivs som ett tvåkomplements tal.  
*Draw the simplified structure when the coefficient is described using two's complement.* (4)
  - Förenkla multiplikatorn så endast 1 fulladderare behövs. Rita den nya strukturen.  
*Simplify the multiplier so the resulting multiplier only uses 1 full adder. Draw the new structure.* (6)

5. Filtret nedan skall implementeras med tidsdelade homogena beräkningselement. Additionerna har en latency som är 1 tidsenhet, multiplikation  $a$  har en latency som är 3 tidsenheter och multiplikation  $b$  har en latency som är 4 tidsenheter.

*The filter below is to be implemented using time-shared homogenous processing elements. Additions have a latency of 1 time unit, multiplication  $a$  has a latency of 3 time units, and multiplication  $b$  has a latency of 4 time units.*

- a) Beräkna minimal samplingsperiod.  
*Compute the minimal sample period.* (4)
- b) Rita upp precedensgraf för filtret.  
*Draw the signal flow graph in precedence form.* (6)
- c) Schemulera filtret med en samplingsperiod av 8 tidsenheter.  
*Schedule the filter using a sample period of 8 time units.* (12)

