

Tentamen

EDAA05 Datorer i system

Lösningsförslag

2012–10–22, 08.00–13.00

1. När det blir svårare och svårare att göra de enskilda kretsarna snabbare får man istället utnyttja möjligheten att parallellisera arbetet. De flesta moderna datorer är därför utrustade med flera separata processorer som arbetar parallellt. Ofta integreras processorerna på ett och samma chip och kallas då vanligen för "kärnor".

2.
 - a) Den siffra som är mest värd, dvs står längst till vänster när man skriver ut ett binärt tal, är den mest signifikanta siffran. Den minst signifikanta siffran är på motsvarande sätt den som siffra som står längst till höger och alltså motsvarar det minsta värdet. I det binära talet 10 är alltså 1 den mest signifikanta siffran och 0 den minst signifikanta siffran.
 - b) 10111011 och BB
 - c) 11011101
 - d) 93
 - e)

| | | |
|-------|---|----------|
| xxx | x | |
| | | 10101101 |
| - | | 1110010 |
| ----- | | |
| | | 00111011 |

"x" markerar positioner från vilka lån gjorts.

3.
 - a) Påstående 2 och 4 är sanna. Valet av teckenkodningsstandard styr inte vilket (eller vilka) tecken som används för att markera radslut. En s.k. BOM *kan* förekomma i början av en textfile kodad enligt UTF-8, men är inte nödvändig.
 - b) UTF-32 representerar varje tecken med 32 bitar (4 byte). UTF-16 representerar tecken i form av 16-bitarsenheter (2 byte). Eftersom alla tecken inte kan representeras med 16 bitar måste vissa tecken representeras med två 16-bitarsenheter. Vi representerar alltså tecken med ett varierande antal bitar (16 eller 32). UTF-8 representerar de vanligaste (i västerländska språk) tecknen med 8 bitar, lite mer ovanliga med 16, och ännu mer ovanliga med 24 eller 32 bitar. Fördelen med UTF-8 jämfört med UTF-32 är att UTF-8 kräver mindre minnesutrymme för att lagra en text. Nackdelen är att det blir lite svårare att tolka texten eftersom det finns så många olika varianter på hur tecknen representeras. UTF-16 kan sägas vara ett mellanting mellan de två ytterligheterna.

4. a) Sanningstabellerna blir...

| x | y | z | $x \wedge (y \vee z)$ | $x \wedge \neg y \vee y \wedge \neg z$ |
|-----|-----|-----|-----------------------|--|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 |

Eftersom sanningstabellerna för de två uttrycken skiljer sig åt gäller inte heller likheten.

b) $x \vee \neg(y \wedge x) =$ (De Morgans lag)

$x \vee \neg y \vee \neg x =$ (kommutativa lagen)

$(x \vee \neg x) \vee \neg y =$ (inverslagen)

$1 \vee \neg y =$ (ett och noll-lagen)

1

c) Logik-Lars har felaktigt tagit bort parenteserna runt $y \vee z$. Eftersom \wedge har högre prioritet än \vee är detta inte tillåtet.

Korrekt genomförd förenkling bör istället vara:

$x \wedge y \vee x \wedge (y \vee z) =$

$x \wedge y \vee x \wedge y \vee x \wedge z = (x \wedge y) \vee x \wedge z =$

$x \wedge y \vee x \wedge z =$

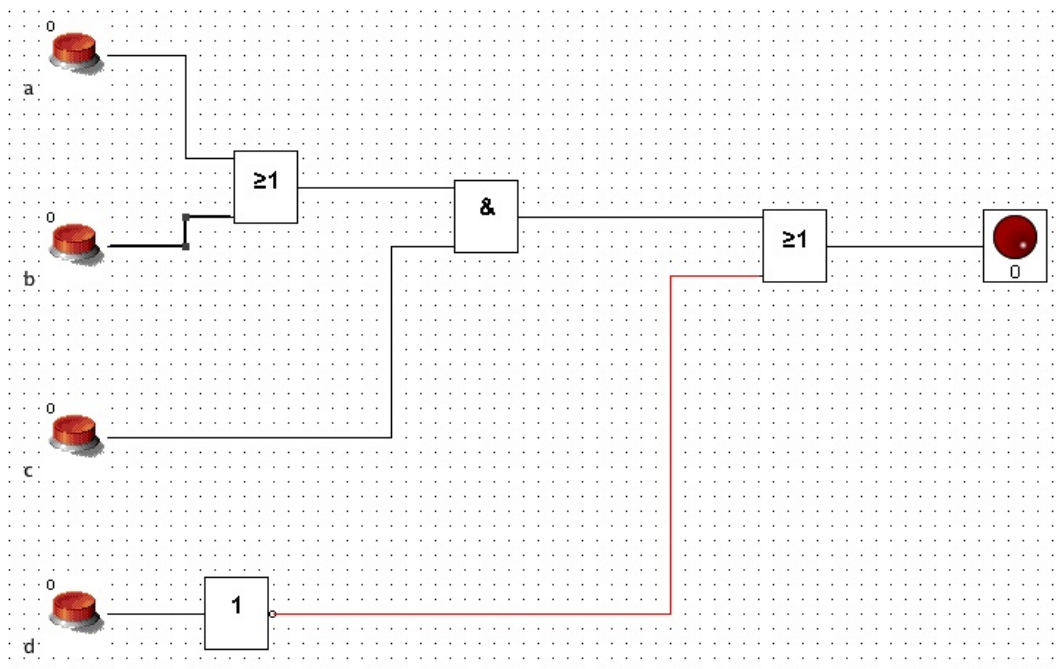
$x \wedge (y \vee z)$

Som svar på frågan godtas resultatet av något av de två sista stegen ovan.

5. a) $x = a \wedge (\neg b \vee \neg c)$

Se upp med beräkningsordningen och operationernas prioriteter!

b) Uttrycket kan t.ex. realiseras på följande sätt (ritat i LogicSim):



6. a) Uttrycken nedan är direkt härledda från tabellen och möjligheter till förenkling finns. Att finna förenklade uttryck lämnas dock som övning åt den intresserade studenten. Paranteser har satts ut för tydlighetens skull (även om de egentligen är onödiga).

$$o = (\neg i \wedge \neg s1 \wedge s2) \vee (i \wedge s1 \wedge \neg s2)$$

$$s1' = (\neg i \wedge s1 \wedge \neg s2) \vee (i \wedge \neg s1 \wedge s2)$$

$$s2' = (\neg i \wedge \neg s1 \wedge \neg s2) \vee (\neg i \wedge \neg s1 \wedge s2)$$

- b) Den färdiga maskinen kan t.ex. realiseras på nedanstående sätt (ritat i LogicSim). Mindre och enklare realiseringar är möjliga, men så här blir det om man direkt utgår från uttrycken ovan.

