

Repetition
TSIU05 Digitalteknik Di/EL

Michael Josefsson

Här kommer några frågeställningar och uppgifter du kan använda för att använda som egenkontroll på om du förstått huvudinnehållet i respektive föreläsning. Frågeställningarna är inte avsedda att vara speciellt mariga utan rimliga utifrån föreläsningens innehåll.

Användningen är helt frivillig. Inget facit finns eller kommer att finnas. Diskutera gärna din lösning och jämför med en kamrats lösning. Blev det samma? Varför? Varför inte? Kan det vara samma lösning bara att den är skriven på annat sätt? Diskutera!

- Efter föreläsning 1:
 1. Rita, så komplett du kan, elektriskt schema med strömbrytare för Ex 4. "Simulera" sedan genom att prova med några sensorers olika ut signaler (1/0) och följa dem till chaufförens lampa.
 2. Övertyga dig att varje rad i sanningstabellen för C_2 ger en *term* i det booleska uttrycket för C_2 .
 3. Använd sanningstabell för att visa att $\overline{B} \cdot \overline{C}$ inte är samma som \overline{BC}
 4. Rita exempel 2, 3 och 4 med schemasymboler.
 5. Kan du, utan större svårighet, gå från sanningstabell till booleskt uttryck och vice versa? Om inte, träna på det!

- Efter föreläsning 2:

1. Övertyga dig, med sanningstabell eller räknelagar, att $u(a, b, c) = ac + b\bar{c} = \overline{\overline{ac + b\bar{c}}} = \overline{\overline{ac}} \cdot \overline{\overline{b\bar{c}}}$
2. Rita sedan en krets för uttrycket u ovan med schemasymboler och formulera *i ord* vad det egentligen är den gör?
3. Träna räkneregler. Speciellt *absorption* och *consensus* brukar vara krångliga.
4. Träna i att formulera booleska uttryck så de enbart utförs med NAND- eller NOR-grindar. Denna sorts "nand-ifiering"/"nor-ifiering" behöver du inför laborationerna.

- Efter föreläsning 3:

1. Vad innebär att ett K -nät är *momentant*?
2. Vad är en *minterm* och en *normalform*?
3. Är alla SP -former också normalformer? (Om "nej", vad skiljer dem åt?)
4. Vad finns det för likheter mellan SP -form och tvånivåers "OCH-ELLER-nät"?
5. Vi förstår att minimeringen

$$f(x_1x_0) = \overline{x_1}x_0 + x_1x_0 = (\overline{x_1} + x_1)x_0 = 1 \cdot x_0$$

fungerar. Kan du hitta motsvarande operationer i Karnaughdiagrammet?
(Svara först "Ja" och hitta dem sedan!)

- Efter föreläsning 4:
 1. Skriv (och lär in) de decimala talen 0–15 som
 - a) Binära tal, hexadecimala tal, BCD-tal, graykodade tal.
 2. Varför heter det egentligen *binary coded decimal*?
 3. Med fyra bitar kan vi få fram 16 kombinationer (0000–1111). Hur många kombinationer ger 5, 6, 7 och 8 bitar? N bitar?
 4. Vi behöver 7 kombinationer. Hur många bitar behövs? Hur många bitar för N kombinationer?
 5. Omvandlar du enkelt mellan hexadecimala tal och binära tal? Det borde du!
 6. Vad är poängen med *Gray*-kod?
 7. Vad är poängen med *BCD*-kod?
 8. Det finns ett samband mellan strecken på denna nyckelhållare. Finn det!



- Efter föreläsning 5:
 1. Repetera *D-vippan*, *tillstånd*, *aktiv flank* och se till att du lätt kan skriva utsignalen i ett tidsdiagram om du har D- och CLK-signalerna givna.
 2. Tillståndstabeller och sanningstabeller ser ganska lika ut. På vilket sätt skiljer de sig?
 3. Tänk igenom vad som egentligen skiljer utsignalen från ett sekvensnät av MOORE-typ och ett av MEALY-typ.
 4. På föreläsningen konstruerade vi en autonom binärräknare (sekvens: {00, 01, 10, 11, 00, ...}). Övergången 01→10, borde vara "jobbig" eftersom två bitar ska ändras. Kräver en autonom grayräknare {00, 01, 11, 10, 00, ...} enklare logik? Konstruera en sådan! Är resultatet rimligt?
 5. Rita *tillståndsdigrammet* som hör till grayräknaren ovan.
 6. Tänk igenom (och helst genomför) de förändringar som en yttre START/STOPP-knapp skulle medföra i föreläsningens binärräknare. Hur ska tillståndstabellen utformas för att ta hänsyn till en yttre signal? Ledning: Rita tillståndsdigrammet och börja därifrån med att fundera på vad som skall styra nästa tillstånd q^+

- Efter föreläsning 6:
 1. Förstå MOORE-modellen och MEALY-modellen!
 2. Använd föreläsningens första exempel för att repetera de olika stegen i lösningen. Alla steg finns i den lösningen, med lite vana kommer du kunna slå ihop steg 3 och 4 till ett enda moment.
 3. Gå baklänges, dvs från schema till tillståndsgraf, i samma exempel.
 4. Vandra runt i tillstånden i den större 4-tillståndiga grafen. Vilka olika sekvenser leder till $u = 1$? Hur ska en insekvens se ut för att aldrig komma tillbaka till S0? För att så snabbt som möjligt komma till S0 eller S3? Fundera!
 5. Du gjorde 1 ovan va?

- Efter föreläsning 7:

1. Inse att ett givet sekvensnät kan realiseras med K -nät likaväl som minne. Använd labsatsens PROM (16 rader a 4 bitar) i några av bokens övningsuppgifter.
2. Använd komponentsatsens binärräknare (LS669). Lägg till logik för att få den att bli en dekadräknare och en modulo-13-räknare. Vilka räknarbitar måste *egentligen* avkodas för dessa räknare? (Alla fyra behöver inte avkodas!)
3. Skriv ut kompletta booleska uttryck för q_i^+ i exempel 31. Rita ett blockshema över den nya lösningen. Rita schemalösningen med grindar för K -nätet för q_i^+ .
4. I exempel 32 föreslogs att blocket "x1x2" kunde utföras med två-ingångars *muxar* och efterföljande avkodning för att till exempel få $6 \cdot 2 = 12 \rightarrow 1 + 2 = 3$. Realisera denna avkodning med grindar.
5. Fundera på laboration 2 (tärningen och bromsljuset).

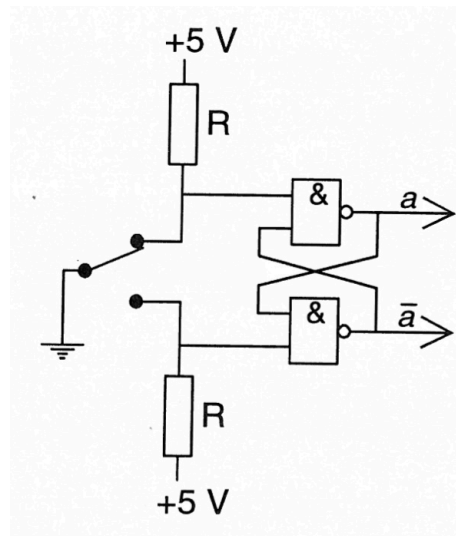
- Tärningen utförs som ett sekvensnät med utfallsrummet $\{1, 2, 3, 4, 5, 6\}$ och som klockas med hög frekvens. *Slumpen* utgörs av hur länge man håller nere tryckknappen. Sekvensnätet måste såklart ha detta utfallsrum, *dock inte nödvändigtvis i denna ordning.*

Kontrollera att din lösning går att genomföra med de typer och antal grindar som finns i labsatsen. (Gissningsvis behöver du prova flera olika kodningar innan du hittar en som är realiserbar). PROM eller färdiga räknare för inte användas i tärningen, den skall lösas med grindar och vippor.

- Bromsljuset är mer omfattande och här kan PROM ge en elegant lösning.

- Efter föreläsning 8:

1. Repetera SR-vippans härledning. Metoden med en hjälpvariabel är vägen fram.
2. En vanlig tillämpning av av NAND-grindar är *avstudsning* av insignaler från strömbrytare. Det visar sig att när en strömbrytare sluter/bryter en strömkrets uppstår under några millisekunder s.k. *kontaktstudsar*. Förklara hur följande koppling eliminerar eventuella kontaktstudsar vid slutning/brytning av kontakten. Tänk på att själva tungan i kontakten en kort stund inte har kontakt med någon av ingångarna.



3. I boken finns flera sorters vippor beskrivna. En lämplig övning är sätta sig i sidorna 256–257 där T-vippan beskrivs. Gör några tidigare uppgifter med T-vippor istället för D-vippor.
4. (Överkurs light) Anledningen till J/\overline{K} -vipporna i labsatsen är att de ofta leder till enklare logik för sin önskade utsignal. Man behöver inte färdigställa hela nästa tillstånd innan vippan utan det räcker att peta på J respektive K för att få rätt sak att hända. J/\overline{K} -vipporna svarar mot följande sanningstabell:

| J | K | Q^+ |
|-----|-----|----------------|
| 0 | – | Q |
| 1 | – | 1 |
| – | 0 | 0 |
| – | 1 | \overline{Q} |

Notera att tabellen gäller J/K inte J/\overline{K} . Enklast blir att räkna enligt sanningstabellen ovan och sedan låta signalen gå genom en inverterare för att få \overline{K} . För övrigt gäller allt vi vet sedan tidigare om sekvensnät som vanligt.