



NTNU

Det skapende universitet

Side 1 av 14

INSTITUTT FOR ELEKTRONIKK OG TELEKOMMUNIKASJON

EKSAMEN I FAG TFE4101 KRETS- OG DIGITALTEKNIKK, LF DIGITALTEKNIKKDELEN AV EKSAMEN (VERSJON 1)

Faglig kontakt: Ragnar Hergum (1–3.5) / Per Gunnar Kjeldsberg (3.6–4)

Tlf.: 920 87 172 / 934 59 550

Eksamensdato: Fredag 11. desember 2015

Eksamenstid (fra - til): 0900-1300

Hjelpemidler: D–Ingen trykte eller håndskrevne hjelpemidler er tillatt.
Bestemt, enkel kalkulator tillatt.

Annen informasjon: Maksimalt antall poeng for hver oppgave er gitt i parentes.
Maksimalt antall poeng oppnåelig totalt: 100.

Sensur faller innen 11. januar 2016.

Målform: Bokmål

Antall nummererte sider: 14

Antall unummererte sider i vedlegg: 0

Kontrollert av:

Dato

Sign

Tom side

Oppgave 1 (15%)

- a) (5%)
- b) (10%)

Oppgave 2 (15 %)

- a) (5%)
- b) (5%)
- c) (5%)

Oppgave 3 (40%)

- 3-1 A.
 B.
 C.

- 3-2 A.
 B.
 C.

- 3-3 A.
 B.
 C.

- 3-4 A.
 B.
 C.

- 3-5 A.
 B.
 C.

- 3-6 Hvilket av disse tre tallene har IKKE samme tallverdi som de to andre?
 A. $101011, 101_2$
 B. $A_3, 5_{16}$
 C. $22_3, 22_4$

LF:

Her kan vi benytte polynomform til å regne ut og sammenlikne tallverdiene på desimalformat. Vi kan imidlertid alternativt sjekke om binærformen i A svarer til hexadesimalformen i B ved å gruppere fire og fire bit, fra komme mot venstre og høyre. Vi ser da at hexadesimal versjon av A er $2B$, A_{16} (husk å legge til en ekstra 0 etter siste bit etter komma for å få en gruppe på fire). A og B er følgelig ikke like. Siden C er firetallsystem som også er en potens av to så kan vi her gruppere to og to bit i A fra komma mot venstre og høyre for å sammenlikne med C. Vi ser da at vi får $223,22_4$. A og C er derfor like mens B er forskjellig.

3-7 Hvilken av følgende påstander er IKKE korrekt?

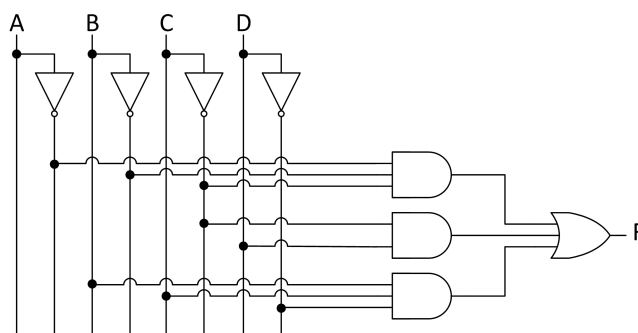
- A. Addisjon av tall på formatet fortegn-tallverdi er mer komplekst enn addisjon av tall på formatet 2s-komplement.
- B. Ved addisjon av tall på formatet 2s-komplement tar man ikke særskilt hensyn til fortegnsbiten, men behandler alle bit likt.
- C. Ved addisjon av tall paa formatet 2s-komplement saa kan man detektere overflyt ved aa se paa mente inn og ut av mest signifikante bit. Dersom disse er like har det oppstaatt overflyt og svaret er feil.

LF:

A og B er korrekte. Første del av C er også korrekt, men det er dersom mente inn og ut av fortegnsbiten er ulik at det har oppstått overflyt og at svaret er feil.

3-8 Gitt tre representasjoner av en Boolsk funksjon (kretsskjema, Karnaughdiagram og likning). Hvilken av representasjonene nedenfor svarer IKKE til samme Boolske funksjon som de to andre?

- A. Kretsskjema i figur 1.
- B. Karnaughdiagram i figur 2.
- C. Likning: $F = (\bar{B} + D)\bar{C} + B(C + D)$



FIGUR 1 – Boolsk funksjon representert ved kretsskjema

		CD			
		00	01	11	10
AB	00	111	111		
	01		111	11	111
	11		111	11	111
	10	X1	1X1		

FIGUR 2 – Karnaughdiagram med mintermer for alle funksjoner

LF:

Vi finner først likningen for kretsskjemaet i A. Den øverste OG-porten gir $\bar{A}\bar{B}\bar{C}$. Den midtre gir $\bar{C}D$ og den nedre gir $BC\bar{D}$. Den etterfølgende ELLER-porten gir totalt: $F = \bar{A}\bar{B}\bar{C} + \bar{C}D + BC\bar{D}$. Vi merker av de tilhørende mintermene i Karnaughdiagrammet fra B med røde 1-ere.

Likningen i C kan vi løse ut for enklere å finne mintermene. Det gir $F = \bar{B}\bar{C} + \bar{C}D + BC + BD$. Vi merker av de tilhørende mintermene i Karnaughdiagrammet fra B med blå 1-ere.

Som vi ser av figur 2 så mangler det to røde 1-ere i forhold til de to andre. A svarer dermed ikke til samme funksjon.

- 3-9 Hvilket alternativ lister 0-maxtermene til funksjonen $F(A, B, C, D) = \bar{A}\bar{B}\bar{C} + (A + BC)D + A\bar{B}C$ der A er mest signifikant.

- A. $\prod(0,3,4,8,10)$
 B. $\prod(0,2,7,10,12,13)$
 C. $\prod(2,3,4,5,6,8,12,14)$

LF:

Her setter vi opp sannhetstabellen i tabell 1 for funksjonen F. Radene med 0 i denne sannhetstabellen svarer til 0-maxtermene, det vil si svaralternativ C

Rad	A	B	C	D	F
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1

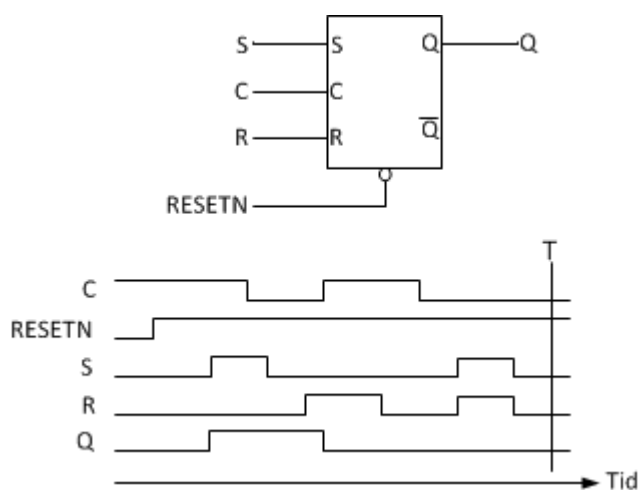
TABELL 1 – Sannhetstabell for funksjonen F i oppgave 3-9.

- 3-10 Gitt SR-låsen med påtrykt stimuli i figur 3. Hva er verdien på utgangen Q ved tidspunktet T?
- A. 0
 B. 1
 C. Ikke mulig å si.

LF:

I figur 3 er resulterende forløp på utgangen Q tegnet inn. Merk at dette er en lås slik at utgangen følger de påtrykte stimuli på inngangene når C er høy. Når C er lav husker låsen verdien som sto på Q da C gikk lav. Siden S og R er høye samtidig mens C er lav så blir dette påtrykket altså ignorert.

Merk ellers at vi i dette tilfellet har aktiv lav reset som setter Q til null når RESETN er lav. I figuren antas det en idealisert SR-lås uten forsinkelse.



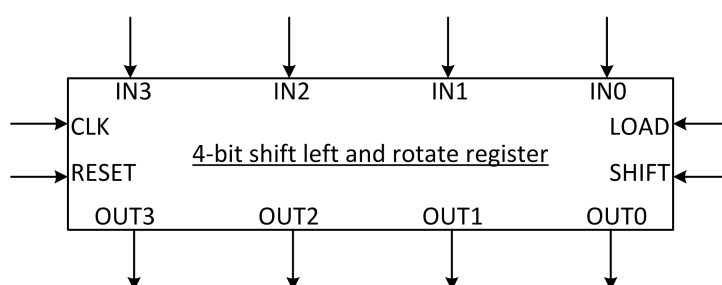
FIGUR 3 – SR-lås med tidsskjema for påtrykk av stimuli på inngangssignalene.

Oppgave 4 (30%)

- a) (11%) I forbindelse med et design av en større digital krets så trenger du et spesielt 4-bit skiftregister med følgende egenskaper:
- Mulighet for parallell innlasting av data styrt av aktivt høyt signal LOAD.
 - Fast parallell data ut.
 - Mulighet for skifting av data mot venstre (fra minst signifikant mot mest signifikant) styrt av aktivt høyt signal SHIFT.
 - Fast rotasjonsegenskap slik at data som skiftes ut fra mest signifikante bit lastes direkte inn på minst signifikante bit.
 - LOAD skal ha prioritet over SHIFT. Dersom begge er høye skal det lastes inn nye data og ikke skiftes.

Figur 4 viser blokkdiagram av det nye skiftregisteret. Lag en tabell tilsvarende tabell 2 der du angir hva som skjer på neste positive klokkeflanke for hver av de mulige kombinasjonene av verdier på LOAD og SHIFT.

Du skal så designe innmaten i skiftregisteret. Til hjelp har du to typer komponenter som angitt i figur 5 (D-vippe og MUX (Selector)). Tegn kretsskjema for innmaten i skiftregisteret ved hjelp av disse.



FIGUR 4 – Blokkdiagram for skiftregister.

LF:

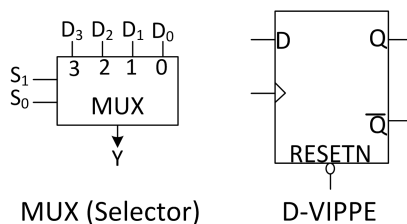
I tabell 2 er det fylt inn hva som skjer på neste positive klokkeflanke for hver av de mulige kombinasjonene av verdier på LOAD og SHIFT.

Figur 6 viser et eksempel på hvordan innmaten av skiftregisteret kan designes. En MUX benyttes til å velge kilde for data som lastes inn i hver D-vippe ved neste klokkeflanke i henhold til spesifikasjonen i tabell 2. Merk spesielt at data på Q-utgangen av vippen lengst til venstre (MSB) føres inn som et av valgene på MUXen lengst til høyre (LSB). Dette gir rotasjonsfunksjonen ved skifting.

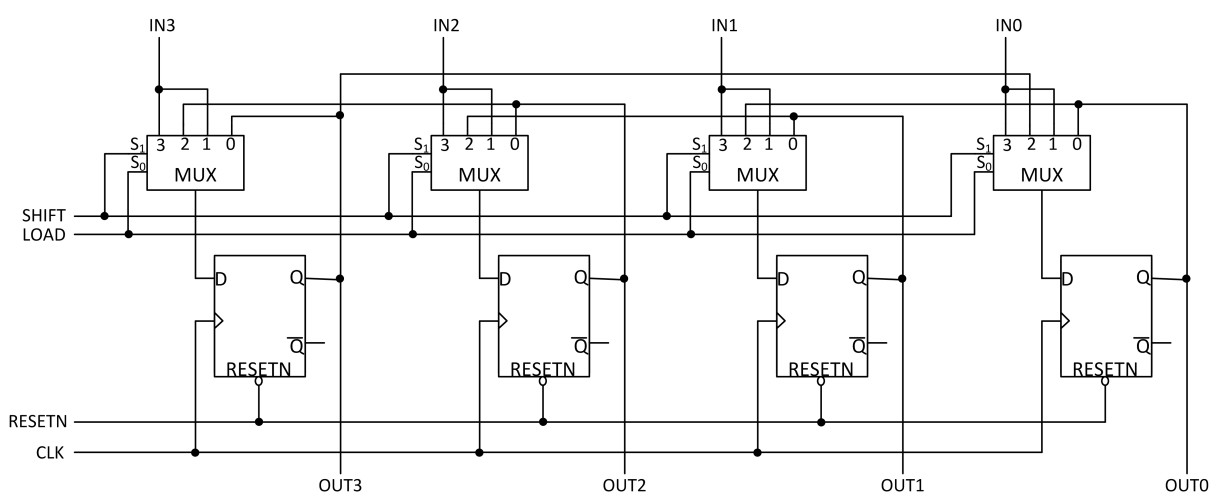
SHIFT	LOAD	Funksjon
0	0	Data beholdes uforandret
0	1	Nye data lastes parallelt inn i skiftregisteret
1	0	Data skiftes et steg mot venstre. MSB skiftes inn på LSB
1	1	Nye data lastes parallelt inn i skiftregisteret

TABELL 2 – Tabell for funksjonsbeskrivelse skiftregister.

- b) (4%) Anta nå at vi laster inn fire bit i skiftregisteret ved å sette $IN[3..0] = 1100$ og $LOAD = 1$ og holder disse verdiene i en klokkeperiode. Deretter setter vi $LOAD = 0$ og $SHIFT = 1$ og holder disse verdiene i en klokkeperiode før vi setter $LOAD = 0$ og $SHIFT = 0$. Anta at oppsett- og holdetider overholdes i alle tilfellene. Hva er nå verdien på utgangen $OUT[3..0]$?



FIGUR 5 – Komponenter for bruk i desing av skiftregister.



FIGUR 6 – Innmat i skiftregister.

LF:

Her laster vi først parallelt inn $IN[3..0] = 1100$ slik at vi etter første positive klokkeflanke får $OUT[3..0] = 1100$. Deretter skifter vi alle bit et steg mot venstre på neste positive klokkeflanke. Eneren som skiftes ut fra MSB ($OUT3$) lastes inn på LSB slik at vi får $OUT[3..0] = 1001$.

- c) (11%) Fra andre deler av det større kretsdesignet kommer det fire signaler A, B, C og D som skal benyttes til å bestemme hvilken funksjon skiftregisteret skal ha til enhver tid. Du skal nå designe en kombinatorisk krets (figur 7) som genererer signalene LOAD og SHIFT ut fra følgende spesifikasjon:

- LOAD skal være høy dersom A er lav og B er høy.
- LOAD skal være høy dersom B er høy og C er lav.
- LOAD skal være høy dersom B er høy og D er høy.
- LOAD skal være høy dersom C er lav og D er høy.
- SHIFT skal være høy dersom A er høy, C er høy og D er lav.
- LOAD og SHIFT skal ellers være lave.

Sett opp en sannhetstabell som svarer til denne spesifikasjonen. Design deretter kretsen basert på dette og optimaliser den slik at implementasjonen krever så få transistorer som mulig. Vis fremgangsmåten du bruker.

LF:

Sannhetstabellen i tabell 3 svarer til spesifikasjonen. Ut fra denne kan vi nå først lage et Karnaughdiagram for LOAD og danne primimplikanter basert på 1-mintermene. Dette er vist i figur 8 og gir funksjonen:

$$LOAD = \bar{A}B + \bar{C}D + B\bar{C} + BD$$

Alternativt kan vi danne primimplikanter basert på 0-maxtermene. Dette er vist i figur 9 og gir funksjonen:

$$LOAD = (B + D)(B + \bar{C})(\bar{A} + \bar{C} + D)$$

Denne siste har færre literaler og krever derfor tilsynelatende færrest transistorer. Dette kommer imidlertid også an på potensiale for omgjøring til ikke-standard form og eventuell gjenbruk mellom LOAD og SHIFT. La oss derfor først se på SHIFT. Karnaughdiagram for denne er vist i figur 10. Funksjonen for denne er:

$$SHIFT = AC\bar{D}$$

Den enslige primimplikanten i SHIFT er overlappende med en av primimplikantene i maxterm-versjonen av LOAD. Dette kan utnyttes i forbindelse med gjenbruk av ledd når vi nå ønsker å gå over til å bruke NAND og NOR for å spare ytterligere transistorer. Vi tar derfor utgangspunkt i LOAD basert på 0-maxtermer som vi først gjør om til ikke-standard form og så omformer ved hjelp av DeMorgan:

$$LOAD = (B + D)(B + \bar{C})(\bar{A} + \bar{C} + D) = (B + \bar{C}D)(\bar{A} + \bar{C} + D) = \overline{(B + \bar{C}D)(\bar{A} + \bar{C} + D)} = \overline{(B + \bar{C}D) + (\bar{A} + \bar{C} + D)}$$

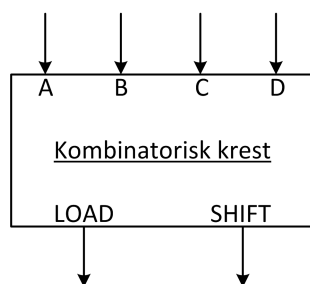
I forbindelse med omgjøring til ikke-standard form er det utnyttet at ELLER er distributiv over OG i den Boolsk algebra.

Til slutt tar vi DeMorgan av uttrykket for SHIFT og ser at vi får det samme som det siste leddet i LOAD:

$$SHIFT = AC\bar{D} = \overline{\overline{AC\bar{D}}} = \overline{(\bar{A} + \bar{C} + D)}$$

Likningene vi her har funnet for LOAD og SHIFT representerer designet når vi samtidig har gjort det klart at deler av LOAD kan gjenbrukes som SHIFT. For portimplementasjon av kretsen, se figur 11 i neste deloppgave.

Merk at vi i denne løsningen har funnet en rekke optimaliseringsmuligheter som ikke alle er opplagt enkle å se. Det er mulig å oppnå tilnærmet full uttelling på denne oppgaven med enklere løsninger. Viktige elementer som i så fall bør være med er sannhetstabell, bruk av Karnaugh eller liknende for forenkling, bruk av ikke-standard form, og bruk av DeMorgan.



FIGUR 7 – Kombinatorisk krets for generering av LOAD og SHIFT.

Rad	A	B	C	D	SHIFT	LOAD
0	0	0	0	0	0	0
1	0	0	0	1	0	1
2	0	0	1	0	0	0
3	0	0	1	1	0	0
4	0	1	0	0	0	1
5	0	1	0	1	0	1
6	0	1	1	0	0	1
7	0	1	1	1	0	1
8	1	0	0	0	0	0
9	1	0	0	1	0	1
10	1	0	1	0	1	0
11	1	0	1	1	0	0
12	1	1	0	0	0	1
13	1	1	0	1	0	1
14	1	1	1	0	1	0
15	1	1	1	1	0	1

TABELL 3 – Sannhetstabell for kombinatorisk krets i oppgave 4 c).

- d) (4%) Tegn kretsen på portnivå (dersom du ikke alt har gjort dette). Tegn også inn kretsens kritiske sti. Hva angir denne?

Dersom du ikke har kommet fram til noen krets i oppgaven over så kan du tegne en tilfeldig krets her som du kan bruke til å demonstrere kritisk sti.

LF:

Figur 11 viser kretsen fra oppgaven over på portnivå. Kritisk sti er inntegnet. Denne angir den veien gjennom kretsen som har lengst forsinkelse. Det vil altså si at det er den som bestemmer hvor lang tid det tar fra det skjer en endring på inngangene til vi er sikker på at utgangene er stabile.

		CD			
		00	01	11	10
AB	00		1		
	01	1	1	1	1
	11	1	1	1	
	10		1		

FIGUR 8 – Funksjonen LOAD representert ved Karnaughdiagram med 1-mintermer

		CD			
		00	01	11	10
AB	00	0		0	0
	01				
	11				0
	10	0		0	0

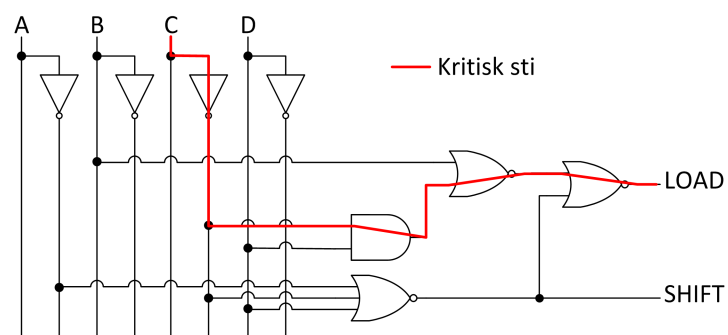
FIGUR 9 – Funksjonen LOAD representert ved Karnaughdiagram med 0-maxtermer

		CD			
		00	01	11	10
AB	00				
	01				
	11				1
	10				1

FIGUR 10 – Funksjonen SHIFT representert ved Karnaughdiagram med 1-mintermer

Oppgitt formel - kretsdel

$$x(t) = x(t_{slutt}) + [x(t_0) - x(t_{slutt})] e^{-(t-t_0)/\tau}; \quad t \geq t_0$$



FIGUR 11 – Kombinatorisk krets for generering av LOAD og SHIFT på portnivå med inn-tegnet kritisk sti.

Vedlegg 1
HUSK Å LEVERE
DETTE ARKET SOM EN
DEL AV BESVARELSEN

Kandidatnr: _____
Emnekode: _____
Side: _____ / _____

SPØRSMÅL NR.	A	B	C
3-1			
3-2			
3-3			
3-4			
3-5			
3-6			
3-7			
3-8			
3-9			
3-10			